

**MACHINE LEARNING TECHNIQUES FOR
IMMUNOTHERAPY DATASET CLASSIFICATION**

**A THESIS SUBMITTED TO THE GRADUATE
SCHOOL OF APPLIED SCIENCES**

**OF
NEAR EAST UNIVERSITY**

**By
EMMANUEL OFOR**

**In Partial Fulfillment of the Requirements for
the Degree of Master of Science**

**in
Information Systems Engineering**

NICOSIA, 2018

EMMANUEL OFOR

MACHINE LEARNING TECHNIQUES FOR

IMMUNOTHERAPY DATASET CLASSIFICATION

NEU

2018

**MACHINE LEARNING TECHNIQUES FOR
IMMUNOTHERAPY DATASET CLASSIFICATION**

**A THESIS SUBMITTED TO THE GRADUATE
SCHOOL OF APPLIED SCIENCES**

**OF
NEAR EAST UNIVERSITY**

**By
EMMANUEL OFOR**

**In Partial Fulfillment of the Requirements for
the Degree of Master of Science
in
Information Systems Engineering**

NICOSIA, 2018

**Emmanuel OFOR : MACHINE LEARNING TECHNIQUE FOR IMMUNOTHERAPY
DATASET CLASSIFICATION**

**Approval of Director of Graduate School of
Applied Sciences**

Prof. Dr. Nadire ÇAVUŞ

**We certify this thesis is satisfactory for the award of the degree of Masters of Science in
Information Systems Engineering**

Examining Committee in Charge:

Assoc.Prof.Dr. Kamil Dimililer

Department of Automotive Engineering, NEU

Assist.Prof.Dr. Boran ŞEKEROĞLU

Department of Information systems Engineering,
NEU

Assist.Prof.Dr . Yöney Kırsal EVER

Computer and Communications Engineering,
NEU

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all materials and results that are not original to this work.

Name, Surname: Emmanuel Ofor

Signature:

Date:

ACKNOWLEDGMENTS

I would like to sincerely thank my supervisor Assoc. Prof. Dr. Kamil DİMİLİLER for his understanding, patience, and guidance throughout my graduate studies at Near East University. His supervision was paramount in providing a well-rounded experience in projecting my long-term career goals. I graciously thank you for all you have done for me Assoc. Prof. Dr. Kamil DİMİLİLER.

Furthermore, I would like to thank my family for their love, prayers and their great confidence in me, also without forgetting my awesome friends; Jeffrey Idehenand Amadi Samuel for their genuine support.

ABSTRACT

This thesis presents classification of immunotherapy data using six machine learning techniques; Decision tree, Random forest, k-nearest neighbors (KNN), Support Vector Machine (SVM), Logistic Regression, Convolutional Neural network (CNN). The above models are the chosen for the classification task after a couple of thorough research.

Machine learning classification of immunotherapy dataset in medical applications is useful to radiologist/doctors in diagnostics and interpretation of immunotherapy data thus increasing treatment options. The classification was executed using all seven techniques by deploying a validated immunotherapy dataset from the UCI repository. Comparison between the techniques was performed, in order to discover accuracies as well as to detect the model that performed best in the classification task. The accuracy ranged between 81.11% and 85% in the training, K-Nearest Neighbor outperformed the other five techniques and CNN performed least.

Keywords: cancer; immunotherapy; Decision tree; Random forest; k-nearest neighbors; K-Fold; Support Vector Machine; Logistic Regression; Convolutional Neural network

ÖZET

Bu tez, altı makine öğrenme tekniği kullanılarak immünoterapi verilerinin sınıflandırılmasını sunmaktadır. ; Decision tree, Random forest, k-nearest neighbors (KNN), Support Vector Machine (SVM), Logistic Regression, Convolutional Neural network (CNN). Yukarıdaki modeller ayrıntılı araştırma bir çift sonra sınıflandırma görevi için seçilir.

İmmünoterapi veri setinin tıbbi uygulamalardaki sınıflama öğrenme sınıflandırması, radyolog / doktorların immünoterapi verilerinin tanısı ve yorumlanmasında ve dolayısıyla tedavi seçeneklerinin artırılmasında yararlıdır. Sınıflandırma, UCI deposundan valide edilmiş bir immünoterapi veri setinin dağıtılmasıyla yedi teknik kullanılarak gerçekleştirilmiştir. Teknikler arasındaki karşılaştırma, diğerlerinin yanı sıra, sınıflandırma görevinde en iyi performans gösteren modeli saptamanın yanı sıra doğrulukları da saptamak için gerçekleştirilmiştir. Eğitimde doğruluk% 81.11 ile% 85 arasındaydı. K-En Yakın Komşu ise diğer beş teknikten daha üstün performans gösterdi ve CNN en az performans gösterdi.

Anahtar Kelimeler: kanser; immünoterapi; Karar ağacı; Rastgele orman; k-en yakın komşular; K-kat; Destek Vektör Makinesi; Lojistik regresyon; Konvolüstal Sinir ağı

TABLE OF CONTENTS

ACKNOWLEDGMENTS	i
ABSTRACT	ii
ÖZET	iii
TABLE OF CONTENTS	iv
LIST OF TABLES	vii
LIST OF FIGURES	viii

CHAPTER 1:INTRODUCTION

1.1 Thesis Problem:	1
1.2 The Aim of the Thesis	2
1.3 The Importance of the Thesis	3
1.4 Limitations of the Study	3
1.5 Thesis Outline	3

CHAPTER 2:LITERATURE REVIEW

2.1 Cancer:	4
2.2 Immunotherapy:	5
2.2.1 Immunostimulants	7
2.2.2 Monoclonal Antibodies	7
2.2.3 Radioimmunotherapy	8
2.2.4 Immunotoxins.....	8
2.2.5 Antibody Directed Enzyme Prodrug Therapy (ADEPT)	9
2.2.6 Immunomodulators.....	9
2.2.7 Adoptive Cell Therapy (ACT).....	9
2.2.8 Cancer Vaccines	10
2.3 Machine Learning in Medical Applications:	14
2.4 Machine learning repositories.....	15

2.5	Over View of Related Works.....	16
-----	---------------------------------	----

CHAPTER 3:MACHINE LEARNING TECHNIQUES

3.1	Machine Learning	18
3.1.1	Supervised Learning	18
3.1.2	Unsupervised Learning.....	19
3.1.3	Reinforcement Machine Learning.....	20
3.2	Machine Learning Techniques.....	21
3.2.1	Decision Tree.....	21
3.2.2	Random forest	26
3.2.3	KNN	27
3.2.4	SVM	30
3.2.5	Logistic Regression	31
3.2.6	Neural Network	32

CHAPTER 4: SYSTEM RESULTS AND DISCUSSIONS

4.1	Methodology Introduction	41
4.2	Tools Used	41
4.2.1	Python:.....	41
4.2.2	Jupyter Notebook:	42
4.2.3	Computer:	42
4.2.4	Dataset:	42
4.3	Algorithms Overview	44
4.3.1	Step1 (Filtering).....	44
4.3.2	Step2 (Normalization)	44
4.3.3	Step 3 (Cross-Validation)	44
4.4	Step4 (Dataset Classification).....	45
4.5	Decision Tree Technique:.....	46
4.5.1	Advantages	46
4.5.2	Decision Tree Training.....	46

4.6	Random Forest Classification:	49
4.6.1	Random Forest Training	49
4.7	KNN Classification:	51
4.7.1	KNN Training	51
4.8	SVM Classification	52
4.8.1	SVM Training	52
4.9	Logistic Regression Classification and Training:	54
4.10	Convolutional Neural Networks Training:	55
 CHAPTER 5: CONCLUSION &RECOMMENDATION		56
5.1	CONCLUSION.....	56
 REFERENCES		58
APPENDIX		65

LIST OF TABLES

Table 2.1: General Information of Various Immunotherapy Attributes	6
Table 2.2: The Categories of Cancer Vaccines	11
Table 3.1: Comparison between Traditional Computing and ANN	35
Table 4.1: Decision Tree Training Parameters	48
Table 4.2: Random forest Tanning Parameters	50
Table 4.3: KNNTanning Parameters	51
Table 4.4: SVM RBF KernelTanning.....	53
Table 4.5: All SVM KernelTanning Results	53
Table 4.6: Logistic RegressionTanning Parameters	54
Table 4.7: Convolutional Neural Network Training Parameters.....	55
Table 5.1: All Techniques Result	56

LIST OF FIGURES

Figure 2.1: Typically shows how immunotherapy work.....	5
Figure 2.2: mAbs structure of activities	7
Figure 2.3: Immunotherapy monoclonal antibodies.....	8
Figure 2.4: Features of Adoptive Cell Therapy (ACT)	10
Figure 2.5: Machine learning pipeline.....	15
Figure 3.1: Supervised machine learning diagram	19
Figure 3.2: Unsupervised machine learning	20
Figure 3.3: Reinforcement machine learning	21
Figure 3.4: Decision tree	22
Figure 3.5: Decision tree nodes	23
Figure 3.6: Entropy Diagram.....	24
Figure 3.7: Random forest Diagram.....	26
Figure 3.8: KNN algorithm Diagram	27
Figure 3.9: Cross-Validation	29
Figure 3.10: Support Vector Machine	30
Figure 3.11: Logistic Regression.....	32
Figure 3.12: Artificial Neurons Structure.....	36
Figure 3.13: A Simple Network	37
Figure 4.1: Treatment Result.....	43
Figure 4.2: Age Range.....	43
Figure 4.3: Flow Chart of the Entire Classification Process	45

Figure 4.4: Generated Decision Tree.....	47
Figure 4.5: Decision Tree Results	48
Figure 4.6: Random forest Results	50
Figure 4.7: KNN Results	52
Figure 4.8: Results from all SVM Kernel	53
Figure 4.9: Logistic Regression Results	54
Figure 4.10: Convolutional Neural Network Results	55
Figure 5.1: Result from All Techniques	57

CHAPTER 1

INTRODUCTION

The popularity of Cancer has increased because of its lethal nature and so many deaths are attributed to it. Records from the US documented that 1 in 4 deaths was cancer related, In 2005 alone 7.6million deaths was attributed to the illness and then at 2030 it is estimated to elevate to about 11.4 million (GLOBOCAN, 2010). In 2014 the new cases in the US alone exceeded 1.6 million and over 500 hundred thousand fatalities. The tendency of having cancer is rapidly increasing (Siegel R, 2014).

Medicine has continually advanced to remedy the cancer situation, towards this effort a multitude of other practices like machine learning has joined in to provide a more reliable medical diagnosis. For possible treatment it follows same basic intuition like other illnesses treatment procedures of early detection and diagnosis is key, conjoined with treatment options which makes a cure is more likely. A possible solution of cancer is immunotherapy; it is a fairly new anti-cancer procedure that utilizes the victims own immune system to hunt and wipeout cancerous cells.

1.1 Thesis Problem:

The problem of cancer detection and treatment has made the researcher to explore various new technologies to address the problem. An objective of this thesis is to specify a technique that yields the highest accuracy. To attain this machine learning techniques was evaluated to determine the technique with the highest performance.

Machine learning techniques perform well in calcification tasks main reason is due to its ability of processing various datasets with complex features. This system very helpful in the health sector because it provides probabilistic range to a degree of accuracy.

Analysis was done using the UCI immunotherapy repository dataset, The UCI [90] dataset was newly uploaded 2018-01-04 consists of 90 instances, 8 attributes and other Characteristics with results of Treatment (UCI, 2018).

1.2 The Aim of the Thesis

The goal of this thesis work is to declare a technique that yields the highest accuracy to analyzing and classifying the immunotherapy dataset according to its attributes. Different supervised ML techniques have been achieved using Python ® and tested on real case immunotherapy treatment dataset from UCI repository.

The primary motivation for using ML is the flexibility of learning and vast attributes by minimizing a loss function that is directly optimizing classification tasks. This is an edge over the use of hand-crafted procedure. The motivation for evaluating using other ML techniques resulted in the need to even-up the training process knowing that large ML models can outfit training set. The fact that fully automated systems for immunotherapy dataset classification are rarely reported in literatures thus it's another motivating factor for using various techniques in this thesis.

The UCI dataset classification will be analyzed using six declared ML techniques and results will be compared to determine the one that yields the highest accuracy. They are:

- Decision tree
- Random forest
- KNN
- SVM
- Logistic Regression
- Convolutional Neural network.

1.3 The Importance of the Thesis

1. Shows classification of immunotherapy dataset using six machine learning techniques and the methodology, is explained in the third chapter.
2. Introduction of the classification technique with highest accuracy.

3. It provides in-depth details of ML techniques useful in immunotherapy classification and in other areas .In order to avoid wasting time and Fast tracks possible options as regards to cancer treatment.

4. Finally, a reliable classification system for immunotherapy dataset is presented as explained in Chapter 4 and 5. As a result, the best accuracy is 85%.

1.4 Limitations of the Study

Although this thesis reached its goals these are the limitations and shortcomings. The classification techniques and performance was processed using immunotherapy dataset gotten from UCI repository which contained 90 instances. It would be better if the researcher got a larger dataset.

1.5 Thesis Outline

The thesis is compiled and partition into five chapters.

Chapter 2: The following are explained: anatomy of the cancer, vaccines as well as immunotherapy, and the machine learning with relations to cancer. Also in this chapter some technical concepts are discussed.

Chapter 3: Explains the theory and application of the various machine learning techniques used in this thesis.

Chapter 4: Includes details on the classification process and experimental results.

Chapter 5: Presents conclusions, remarks, and observations including prospective future works.

CHAPTER 2

LITERATURE REVIEW

2.1 Cancer:

Cancer is a series of extremely complicated and lethal disease comprising of extrinsic and intrinsic molecular factors, thus it's a major problem in the world especially in advanced countries (W.H.O, 2018), medicine and science are yet fix the puzzle and finally defeat this disease. The popularity of Cancer has increased because its lethal nature and so many deaths are attributed to it. Records form the US documented that 1 in 4 deaths was cancer related, In 2005 alone 7.6million deaths was attributed to the illness and then at 2030 it is estimated to elevate to about 11.4 million (GLOBOCAN, 2010). In 2014 the new cases in the US alone exceeded 1.6 million and over 500 hundred thousand fatalities. The tendency of having cancer is rapidly increasing (Siegel R, 2014) and 2015 it was a cause of death for 8.8million people (W.H.O, 2018).

Treatment regarding it is complex but what makes it even harder is metastasis which is the spread or migration from a primary cancerous cell to another cell (secondary) this makes it much more difficult to curtail (Avi Schroeder, 2012). Regular cancer treatment is becoming more reliable and effective, although specific targeting is needed to curtail metastatic cancerous cells (Steeg P.S., 2006).

The wide nature of cancer has been made easy to understand by defining the nature and properties of cancer (DouglasHanahan, 2011); the above researchers pointed out that the obtained practical attributes of cancerous cells are viable through two characteristics; improvement regarding genomic instability that generates random mutations, then tumour-promoting inflammation among which, the infiltrating immune cells (mostly radical immune cells) hold paradoxical impact through instigating increase factors, progress factors, proangiogenic factors, extracellular matrix-modifying enzymes including inductive signals in accordance with the tumour microenvironment so that cancerous cells can benefit.

It is hopeful that declaring the various properties of cancer is a step in the right direction in the development of targeted and efficient anti-cancer agents. Cancer immunotherapy is the focal point of this study from the properties to its classification.

2.2 Immunotherapy:

The act of regulating a cancer patient's own immune system rather than using other therapies to spot and kill cancerous cells is cancer immunotherapy, the process is developing agents that kick starts or boosts the immune system for identification and the termination of cancerous cells (Mittendorf EA1, 2010). The roots of immunotherapy goes all the way to the seventeenth century 1774 to be precise, a physician of Persian Origin introduced "pus" to a cancer patient's leg and the effect was examined, turns out the tumor's growth was significantly suppressed (Davis, 2000) . Between 1893 to 1936 well above 800 soft tissue sarcoma patients was treated by an American surgeon (William Coley) who used a blend of toxins known as Coley's toxins (Streptococcus and Serratia) .This was the first official documented probable immunotherapy procedure, the toxins used to be the only recognized systemic cure for cancer at that era (Davis, 2000) (John M. Kirkwood, 2008) . Through in these early trials of cancer immunology it was noted that the immune system has the ability to the ability to suppress and regulate the advancement of cancer to a point (Gavin P. Dunn, 2002) (Sharma P., 2011).

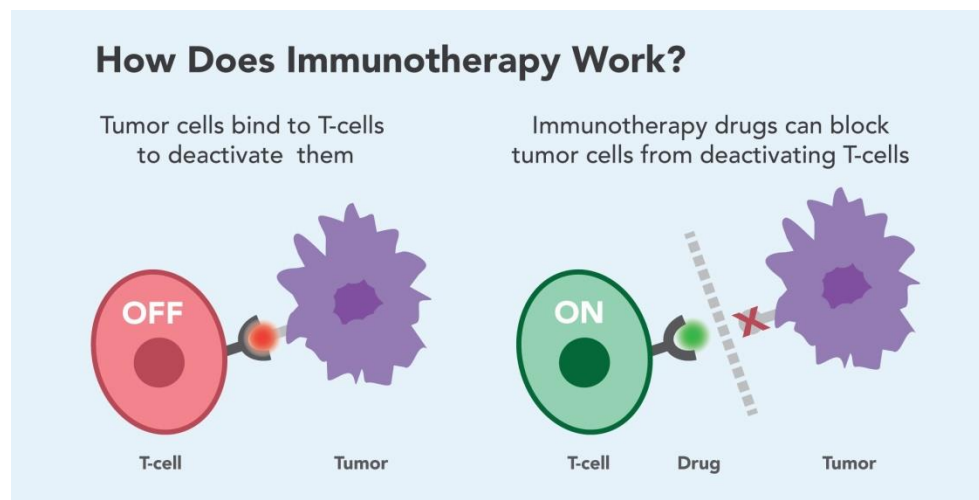


Figure 2.1: Typically shows how immunotherapy work (University, 2015)

Table 2.1: General Information of Various Immunotherapy Attributes

Type	Agents	Information
Cancer Vaccines	Vitespen	vaccine (Peptide-based) utilizing proteins from victim's cancer
	DCVax	Anti-idiotype vaccine that's locked on B cell lymphomas
	Sipuleucel-T	Bonding of cells that has cancer antigen and
	BiovaxID	
Immunomodulators	Ipilimumab	Antibody to PD-1
	Tremelimumab MDX-1106 PF-3512676	TLR-9 Agonist Antibody to CTLA-4 Antibody to CTLA-4
Lymphocytes Transfer	Cells modified (ACT) Adoptive Cell	Integration of ex vivo grown cancer infiltrating or peripheral lymphocytes
Autograph or Allograph		Genetic modification of the lymphocytes before infusion
Radioimmunotherapy	tositumomab	yttrium-90 CD-20 Antibody conjugated to radioactive isotope
	ibritumomab- tiuxetan	CD-20 Antibody conjugated to radioactive isotope
Immunostimulants	Interleukin-2 (IL-2)	A likely growth reason for T-cells triggers
	Alpha-Interferon (IFN- α)	B and T cells and has apoptotic, antiangiogenic and Alpha-Interferon (IFN- α) antiproliferative properties
Monoclonal Antibodies	Cetuximab	the VEGF Against the HER1/EGFR
	Bevacizumab	Against the CD-20 Against the HER-2
	Rituximab	Against
	Trastuzumab	Trastuzumab Bevacizumab Cetuximab

What is known regarding the cellular and molecular structures underlying the immune system's role regarding cancer has increased drastically nowadays, leading to the growth of various strategies (Table 2.1).

2.2.1 Immunostimulants

This is amongst the foremost methods in carrying out immunotherapy. It is a non-specific strategy that targets to raise the system; generally the actions of the lymphocytes that are already engaging the cancerous cells however they are inadequate or under powered to outturn a full-powered immune response. The patient's own immune system is a major determinant factor.

2.2.2 Monoclonal Antibodies

These days monoclonal antibodies is very crucial in immunotherapy due to their wide impact (Figure 2.2). mAbs is short for monoclonal antibodies, some it's activities are Direct toxicity that comprises of (ADCC) antibody-dependent cellular cytotoxicity with (CDC) complement dependent cytotoxicity, these contributes in increasing the actions of effector immune cells thus reducing the growth of tumor (Figure 2.2) (Bisht M., 2010).

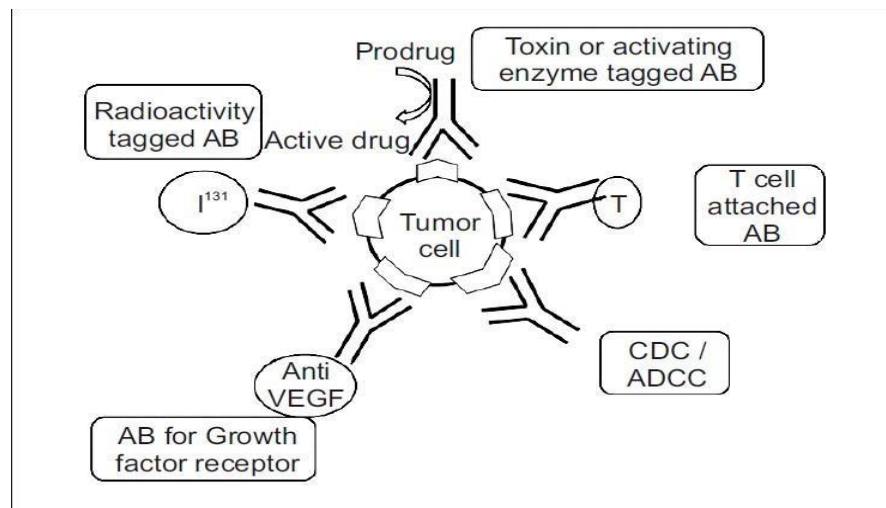


Figure 2. 2: mAbs structure of activities

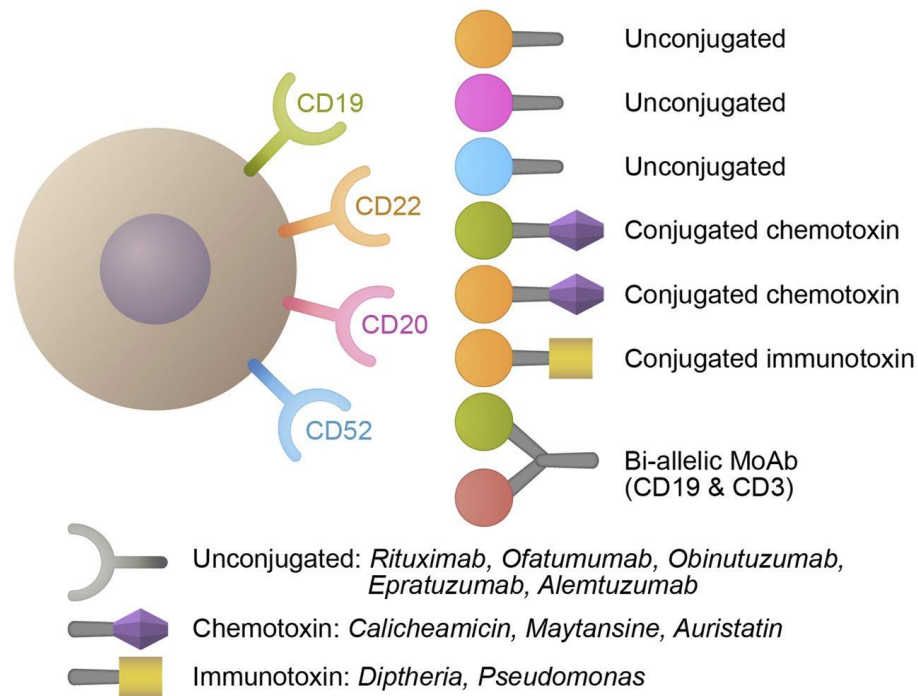


Figure 2. 3: Immunotherapy monoclonal antibodies (Elias Jabbour, 2014)

CDC contributes to Alemtuzumab and Rituximab, Another impact of CDC is that it makes ADCC more effective regardless of these it is still not acknowledged as an effective anti-cancer properties of mAbs (Gelderman, 2004) (Di Gaetano N., 2003).

2.2.3 Radioimmunotherapy

This is the act of deploying radioactive atom to monoclonal antibodies that is aiming at tumour specific antigen; the aim is to reduce the introduction of fatal radiation to the cancerous cells and to keep toxicity level low for strong cells (Peniel M. Dimberu R. M., 2011).

2.2.4 Immunotoxins

Immunotoxins are generated through the composition of plant (derived) or bacterial toxins that pin-points to a particular antigens on the outer layer of a cancerous cells. Early development of these toxins comprises of abrin, ricin, Pseudomonas, gelonin including Diphtheria toxin

(Coombes R.C., 1986). In addition based on series of problems of these methods like speedy evacuation from the blood stream and then immunogenicity has prompted further development (R.V.J, 2011) Some toxins are currently being examined in medical trials (Kreitman R.J., 2011).

2.2.5 Antibody Directed Enzyme Prodrug Therapy (ADEPT)

ADEPT is a different method in anti-cancer therapy where an antibody is deployed to transfer an enzyme that has the ability of activating a nontoxic drug "prodrug" to an agent (cytotoxic) for cancerous cells (R.F, 1996) . In it method, conjugate of an antibody-enzyme is injected in the cancerous cells relying on the uniqueness of the system's antibody. Then, also the prodrug is applied and will remain transformed to a cytotoxic agent solely inside the cancerous tissue where the activating enzyme inhabits (P.D., 2010).

2.2.6 Immunomodulators

These are various primary regulatory elements, that's known as immune checkpoints in our immune system, which regulates the stages of immune reply through a potential prohibition and down regulation to fix homeostasis. All these are essential factors that are of utmost importance because of the improvement on self-tolerance and also to stop autoimmunity, although, cancerous cells gain from this feature of our immune system which is helpful to evade its lethal tendencies (Peniel M. Dimberu R. M., 2011).

2.2.7 Adoptive Cell Therapy (ACT)

The is a recent development in immunotherapy that turns out to be incredibly efficient towards metastatic melanoma (Steven A. Rosenberg, 2009) In Adoptive Cell Therapy, the "T" cells of an affected person that portrays anti tumour activities are located, separated, grown outside the organism, stimulated farther through the tumour antigen supplying cells and introduced again to the affected person. Prior to the introduction of large quantities of tumour infiltrating lymphocytes (TILs), finally the victim may stay influenced among the system together with

increasing the usefulness of the assigned cells (Peniel M. Dimberu R. M., 2011) (Rosenberg S, 2009).

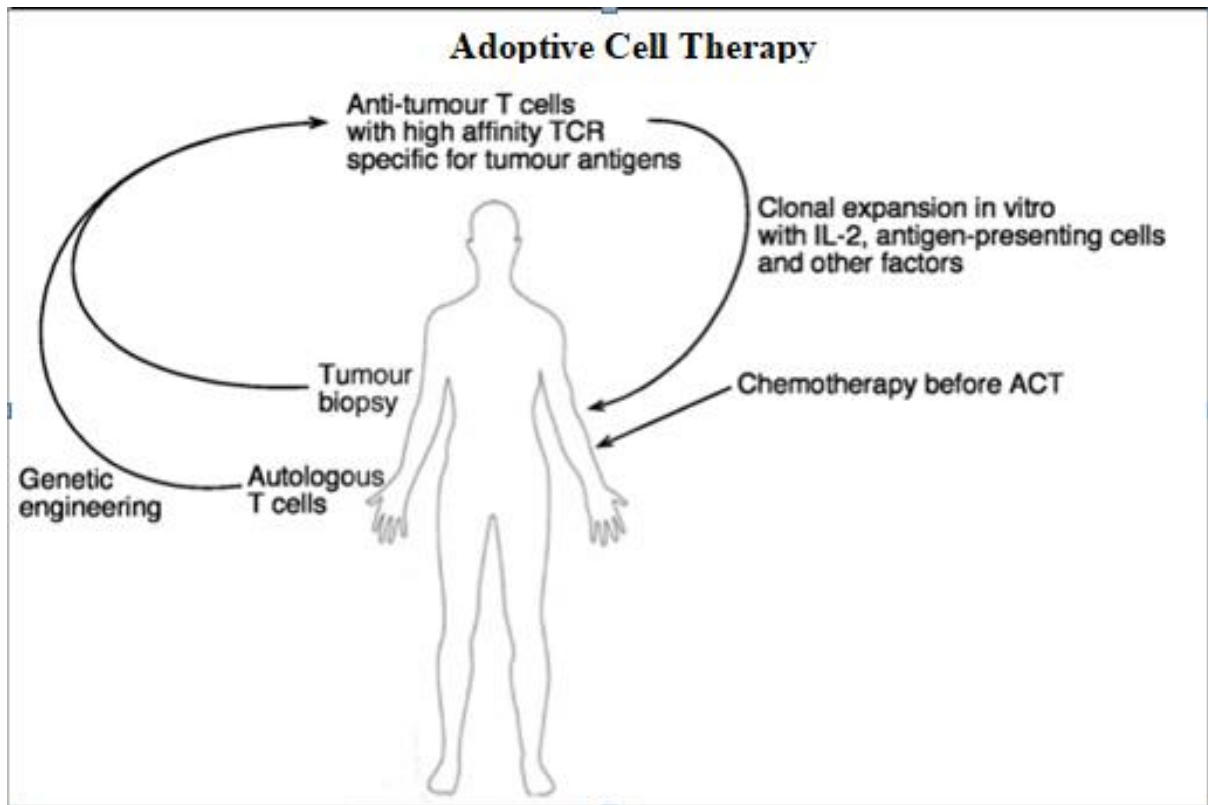


Figure 2.4: Features of ACT (Minda Asfaw Geresu, 2016).

2.2.8 Cancer Vaccines

Cancer vaccines have the most numerous classes within the immunotherapeutic strategies. The production of cancer vaccines is separated into 2 sections; therapeutic and preventative which is known as prophylactic. The two sections are further splitted into sub sections and samples regarding each are quickly mentioned in (Figure 2.5). There are basically two types of the vaccines.

Preventive vaccines, this measure is for healthy people to restrict the growth of cancer also called prophylactic vaccines. Treatment vaccines, this type is targeted towards to an already

existing cancerous cell by boosting the patient's immune system natural response, also known as therapeutic vaccines (Institute, 2015).

Table 2.2: The categories of cancer vaccines (Geresu, 2016)

Preventative	Therapeutic
Hepatitis B	Melanoma and locally renal Cell
Human Papilloma Virus vaccines	Carcinoma advance
Cervarix	Melanoma
Vitepen	Prostate Cancer
Gp100	ProstVac VF
GVAX	BioVaxID
SipuleucelT (Provenge)	Glioblastoma

2.2.8.1 Preventative/ pre-emptive Cancer Vaccines

The accomplishment of these vaccines are impressive, since when introduced over 30 years to avoid elevated risk of tumorigenesis triggered by numerous viral infections. The earliest of such preventative cancer vaccine was developed for hepatitis B of which, In 1981 the FDA approved it and afterwards, In children routine vaccinations it has been approved as an accepted antigen (Peniel M. Dimberu R. M., 2011).

Human papillomavirus is another form of the vaccine. It was tested in 1980s and Harald zur Hausen noted that the HPV forms already existed in a lot of cervical cancer biopsies and also available in cervical cancerous cells (Hausen, 2009).

Beyond being typical vaccines, another purpose is to boost the immune system response rate, in the case of an already diagnosed cancerous cell instead of attempting to stop the formation. This strategy was created based on the fact that victims still creates cytotoxic and also T cells,

matching the antigens that is existing in the cancerous cells (Boon T, 2006). Triggering or boosting the already-existing T cells feedback/responses in opposition to cancerous cells and their numerous unique strategies in production of these vaccines is the motive of Therapeutic cancer vaccines (Figure 2.5) (Mellman I, 2011).

2.2.8.2 Protein Based Vaccines (Peptide)

This kind of vaccines uses an entire short peptide or protein gotten out of cancerous cells. A Major purpose is immunization, based on antigens that are cell-specific. However, recent research showed that there is little improvement with cancer patients that newly got the illness and additionally the sub grouped evaluation shows that victims that has greater intake of Vitespen out lives the other ones with the decreased doses (Aimee E Hammerstrom, 2011) Circumventing these problems may certainly enhance the medical benefits that is made possible by using cancer vaccines.

2.2.8.3 Allogeneic Whole Tumour Cell (Autologous)

These are developed from any of these two cells the autologous, allogeneic cells. Even though the uses of autologous cells kills antigens, the issue of agent selection by making available the potential of locking on to the person's cancer related antigens, Allogeneic Whole Tumour method has been deserted based on the impression that this vaccines of this type generally does not trigger an efficient immune system response (anti-cancer) considering the fact that it wasn't existing (Aimee E Hammerstrom, 2011).

2.2.8.4 Gene Therapy Based Vaccines (GTBV)

These GTBV vaccines are referred to as viral vector or just vector vaccines, due to the fact that it uses viruses to introduce the vaccine (Aimee E Hammerstrom, 2011). In this method, the viral vectors vaccines are designed for the exact cancer antigens for simulation purposes and then improving the response rate of the immune system towards tumor cells containing specific antigens. However benefits that comes from using viruses as a medium of transport comprises of the effortless integration of genes, little amount and capability all of which can result in chronic response of the immune system, Due to the safe functions starting from the

1960s viruses that is linked to the poxvirus group can produce an appealing specimen for this therapy (Aimee E Hammerstrom, 2011).

2.2.8.5 Idiotypic Immunoglobulin Based Vaccines (IIBV)

These vaccines are organized by the help of blending the victim's own cancerous B lymphoma cell units with adding a myeloma cell that is outcome of heterohybridoma shows antibodies that is specific to the affected person's own particular antigens is known as idiotypes. The idiotypes are separated from the already developed antibodies, the heterohybridoma B cells are then purified before they are combined to the keyhole limpet hemocyanin that boosts their immunogenic attributes through supplying precise responses from the T-cells (Aimee E Hammerstrom, 2011) (Reinis M, 2008).

2.2.8.6 Dendritic Cell Based Vaccines (DCBV)

Considering the volume of data recently gathered of the worth of dendritic cells are now recognized because of the potential T-cell development also in conjunction with continued anticancer response. A type of DCBV is referred to as DCVAX Prostate it is an autologous DCBV on the other hand that does not makes use of a an entire protein i.e peptide vaccines and also it doesn't Granulocyte-macrophage colony-stimulating factor (GM-CSF) in the introduction. The production adheres to the affected person's dendritic cells incubation (Christopher J Wheeler, 2009) (Mellman I, 2011).

2.2.8.7 Sipuleucel T (Provenge)

This is used for enhanced metastatic prostate cancer and has been authorized in 2010 by the FDA; the production of therapeutic cancer vaccine was the first that was authorized by FDA. This is an autologous customized cancer vaccine that is organized from the victim's very own blood mononuclear cells. In the light of these trends it is encouraging to see that cancer vaccines are eventually rising as an efficient immunotherapy. However, there are various obstacles that are related with these developments (Aimee E Hammerstrom, 2011).

2.3 Machine Learning in Medical Applications:

Machine Learning (ML) presents methods, techniques, and also tools that are useful aid in solving diagnostic and prognostic problems in a range of defined domains. ML is being utilized for the evaluation and the importance on medical parameters and their combos for prognosis, e.g. calculation of sickness progression, extraction concerning clinical information for result research, therapy strategy and support, and for the typical affected person management. ML is also being utilized for information analysis, such as finding of regularities within the data through appropriately dealing with faulty data, interpretation of continuous information utilized in the Intensive Care Unit, alarming resulting in an efficient and effective monitoring. It is argued that upon the successful implementation of ML techniques can assist in the integration of computer-based infrastructures in the healthcare domain offering possibilities to facilitate and boost the labor of medical specialists and eventually to enhance the effectiveness and quality of clinical care. Some principal ML usages in medicine are stated below.

In a lot of cases, the improvement of these structures is regarded as an attempt to simulate the doctor's knowledge in the identification of malicious areas in minimally invasive imaging tactics (e.g., computed tomography, ultrasonography, endoscopy, confocal microscopy, computed radiography or magnetic resonance imaging). The goal is to extend the expert's capacity to discover malicious regions while reducing the necessity for intervention, and retaining the ability for correct diagnosis.

The usage of ML techniques can grant beneficial aids to assist the physician in a lot of cases, eliminates problems associated with human error and habituation, grant speedy identification of abnormalities and allow analysis in real time.

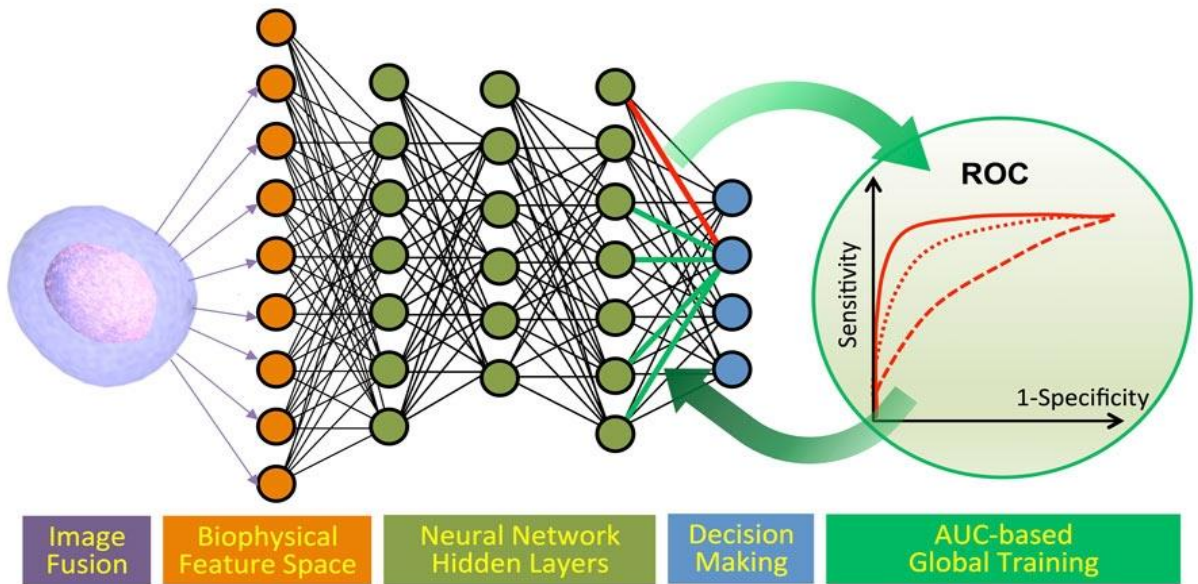


Figure 2.5: Machine learning pipeline (Claire Chen, 2016).

2.4 Machine learning repositories

The UCI Machine Learning Repositories grants access to a wide array of information sets for the experimental evaluation of multiple machine learning algorithms. Specialized ML repositories include quality-controlled data units from defined applications domains. These data sets enable performance evaluation of a technique in various conditions and evaluation of multiple algorithms across various data sets. Understanding the theoretical elements of ML algorithms is essential because of the development of ML capabilities however it is no longer adequate for the progress of these applications. Such services require specific tuning of ML algorithms to skip domain-specific peculiarities of data properties and models, but at the same of time limitations, exceptions, and sufficient level of complexity of the domain. An instance of standard ML repository is the UCI Machine Learning Repository. The UCI ML Repository is a series of databases, domain theories, or data generators mainly for the purpose of serving the ML group for the development, analysis, and evaluation of ML algorithms. It serves as a major supply of data sets for research, development and learning in the ML field (Frank A, 2010).

Data repositories assume a vital function - they supply benchmarking data sets for the improvement of ML services together with algorithms (search, classification, clustering, feature extraction, prediction, forecast), mathematical modeling, and quality evaluation metrics. Below are some of the characteristics

- Detailed: The datasets are well documented
- Comparability: Data gotten from various sources are put together for computation
- Consistency: Provides data with good quality.

2.5 Over View of Related Works

Knowledge of some research work related to this thesis is declared in this chapter.

ANN implementation for the early diagnoses and detection of illnesses has been the challenge of various researches in a while back. A neural network study was proposed in 2006 (Anita Thakur & Bansal, 2016). The study centered on the uses of artificial neural networks within the early detection and discovery of periodontal illnesses and gingivitis using their signs and risk factors. Collected information for 200 patients and these data had been used in the training of the ANN. Multilayer neural network used to be applied in LevenbergMarquardt training algorithm. In this research, sigmoid method ANN was applied with 1 hidden layer.

Dhungels report regarding applied machine learning in Detection, Segmentation and Classification of Mammograms Masses the use of entirely automated Computer Aided Diagnosis CAD structures that can be helpful in mammograms. Also proposed a technique primarily based on newly developed machine learning models to enhance a completely automatic CAD system for automatic detection, segmentation then classification on masses from mammograms. The researcher proposed answer to mass detection. (Dhungel, 2016)

Sally J. Deeb (Sally J. Deeb & Mann, 2015) Indicated that the researchers promoted a top notch quantitative mass spectrometric pipeline to represent formalin-fixed paraffin-embedded tissues of affected persons together with associated subtypes of diffuse massive B-cell lymphoma Using deep learning SVM to remove candidate proteins with the greatest segregating power.

In 2003 Monika shinde did a research (Shinde, 2003) that portrays an automatic mass classification system for Digital Mammogram applications, the result of the general classification outcome is as follow: Based on the 300 image dataset comprising of ninety seven deadly and 203 effective cases, a 63% sensitivity then 89% specificity was achieved. Although researcher indicated that the segmentation requires more investigation.

Mohammed.J provides an effective pc aided mass classification technique in digitized mammograms utilizing Artificial Neural Network (ANN), which performs benign-malignant classification of area of interest (ROI) that consists of mass. They proposed 3 layers artificial neural network (ANN) with seven applications was proposed for classifying the marked areas of benign and deadly and 90.91% sensitivity and 83.87% specificity is gotten as compared to the radiologist's sensitivity of 75% (Islam1, Ahmadi2, & Sid-Ahme, 2010).

CHAPTER 3

MACHINE LEARNING TECHNIQUES

3.1 Machine Learning

This is an area that offers machines (computer aided) the ability to study and learn with no alterations is acknowledged as machine learning. Earlier research of deep learning that has to do with pattern recognition, machine learning and A.I examines the mathematical development of learning to be able to carry out data predictions (K & P, 1998). Guidelines in programming of prediction driven or data (information) driven choices that accurately adhere to algorithmic input model for the datasets. Machine learning really shines in unique cases where executions and programming calculations are tough or not practical; examples of these functions are data mining, image processing, optical character recognition, advanced data encryption and many more. The machine learning process that is used to examine and activate profile behaviors for different components and then it is also utilized to locate tangible patterns without supervision is unsupervised.

Data analysis in the machine learning domain is used in advance systems and computation that are accessible for prediction. This is an analytical system that is predictive for commercial purposes and allows scientists, designers and information researchers to effect reliable, choices and make hidden information known by learning from patterns available in the information set. Machine learning has 3 main types as seen below.

- Supervised
- Reinforcement
- Unsupervised

3.1.1 Supervised Learning

This consists of two parts a tutor and the system, samples containing inputs and preferred outputs are supplied to the system, the aim is learning so that the system can analyze and map the input.

This type of machine learning consists of techniques depends on data that are labeled for its activities, these labeled data are responsible for recording information inputs and the preferred result (output) data. When these algorithms are given input and output, trigger then calculations to examine the outcome of the selected technique to the preferred result and then error checks, which is as a result of the system's parameters (Caruana & Niculescu, 2006). The application of supervised learning is wide. However, one disadvantage concerning supervised learning is that it is solely operational if the data are labeled; but in reality only a handful of data are labeled or a given labeled data might be costly.

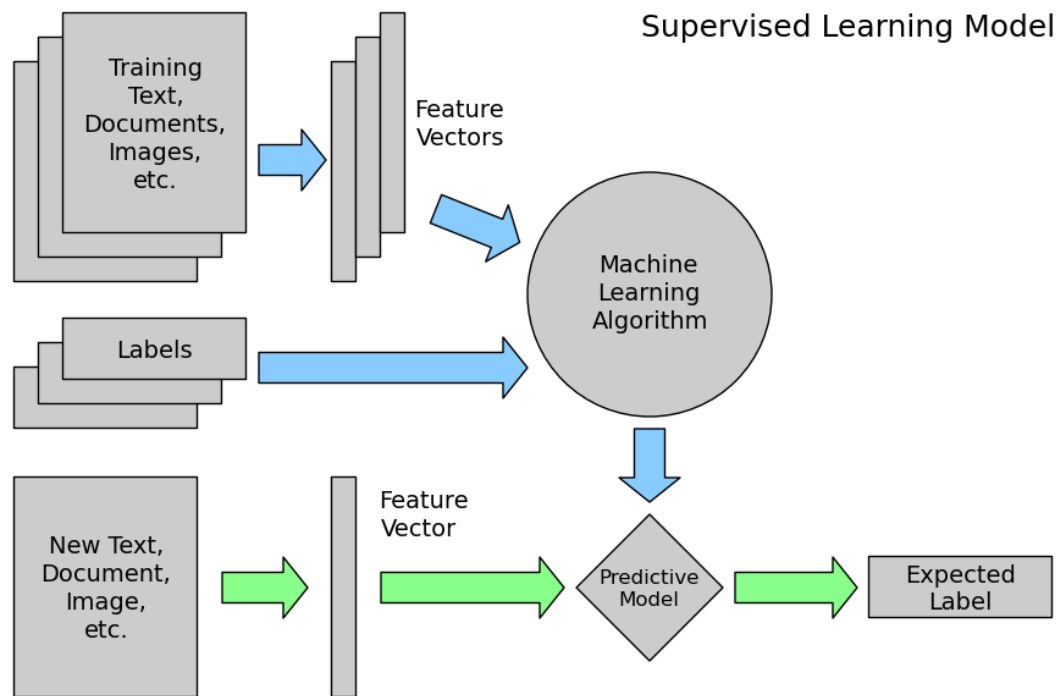


Figure 3. 1: Supervised machine learning diagram

3.1.2 Unsupervised Learning

Unsupervised machine learning model, this model is label-less, allowing the system to discover patterns without any help, only input is required. The purpose is learning/mapping of

featured, unseen or hidden patterns in a dataset. The only information it needs is just the input info/data. A preferred outcome is not provided unlike for supervised machine learning. The main purpose of unsupervised learning is to examine inputted information in search for other structured data and patterns that forms clusters primarily located on the dataset. The Figure: 3.2 depict the model for unsupervised learning.

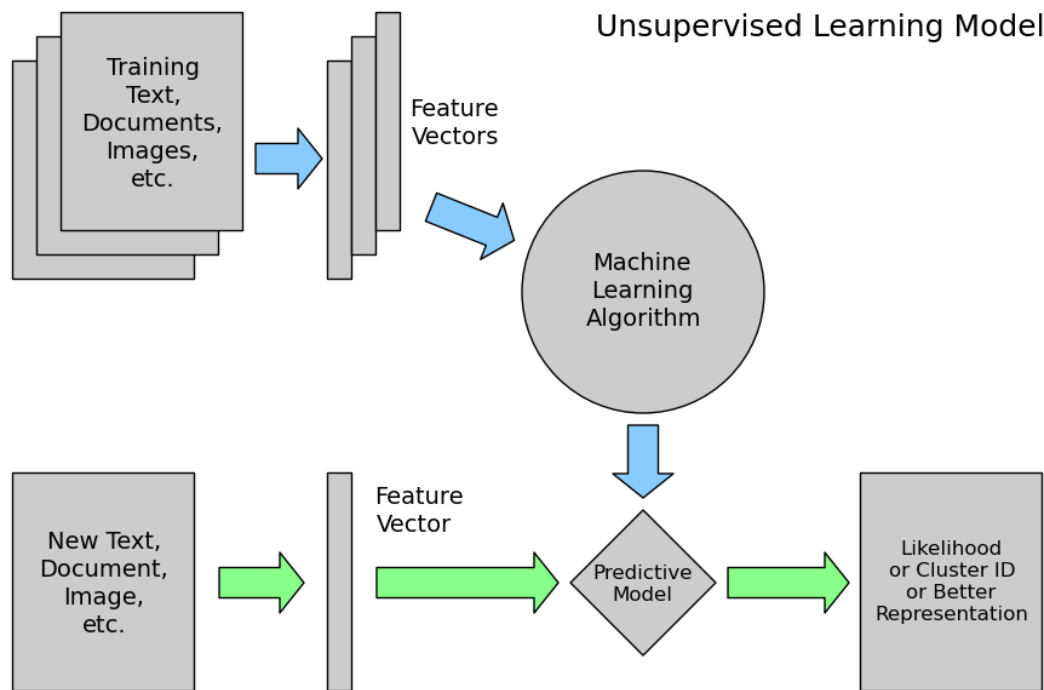


Figure 3. 2: Unsupervised machine learning

3.1.3 Reinforcement Machine Learning

In this type of machine learning is similar to unsupervised learning because the data are not labeled but in this case there is clearly a preferred results; this algorithm has both negative and positive indicators for rewards which is like a pointer for some the most beneficial outcomes. Figure 3.3 Indicates reinforcement machine learning model.

This algorithm maybe used to execute certain tasks such as using a gaming where the application triggers dynamic Interactions and feedbacks in a dynamic environment.

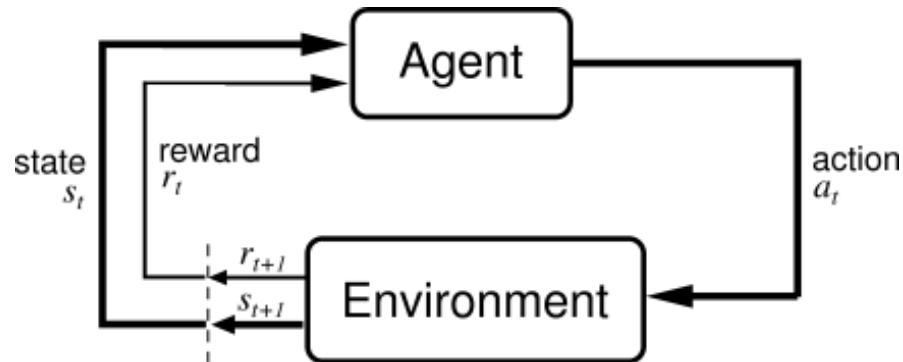


Figure 3. 3: Reinforcement machine learning (UCBWiki, 2016)

3.2 Machine Learning Techniques

In this thesis six supervised machine learning techniques was analyzed and deployed for the classification of immunotherapy dataset (UCI, 2018). The techniques include;

- Decision tree
- Random forest
- Logistic Regression
- SVM
- KNN
- Convolutional Neural Network.

3.2.1 Decision Tree

Tree based machine algorithms are regarded as one of the superior and generally used supervised learning methods. Tree based techniques empower predictive models together with

high accuracy and very simple interpretation compared to other algorithms, also used in the mapping of nonlinear relationships. Basic outcome is “YES” or “NO”.

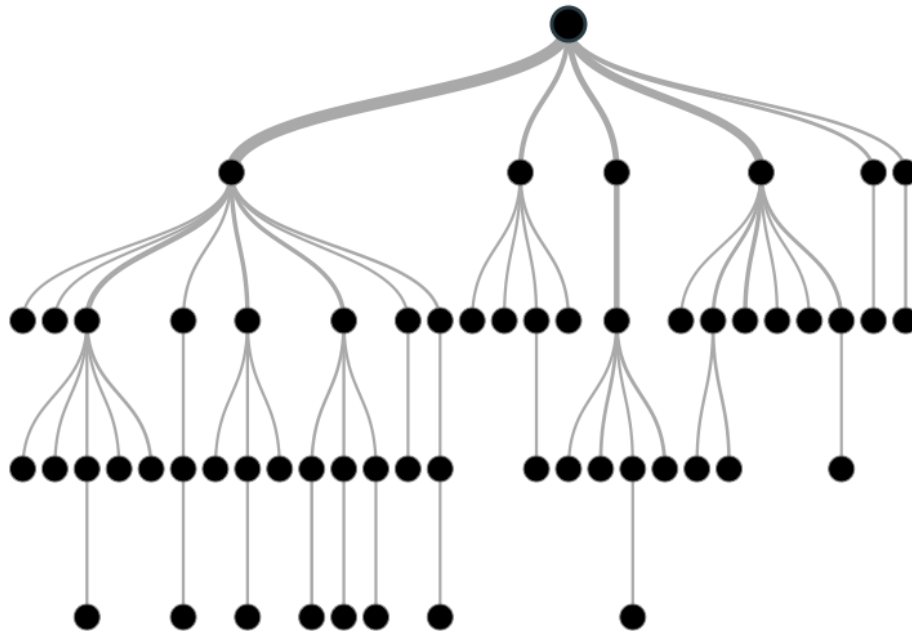


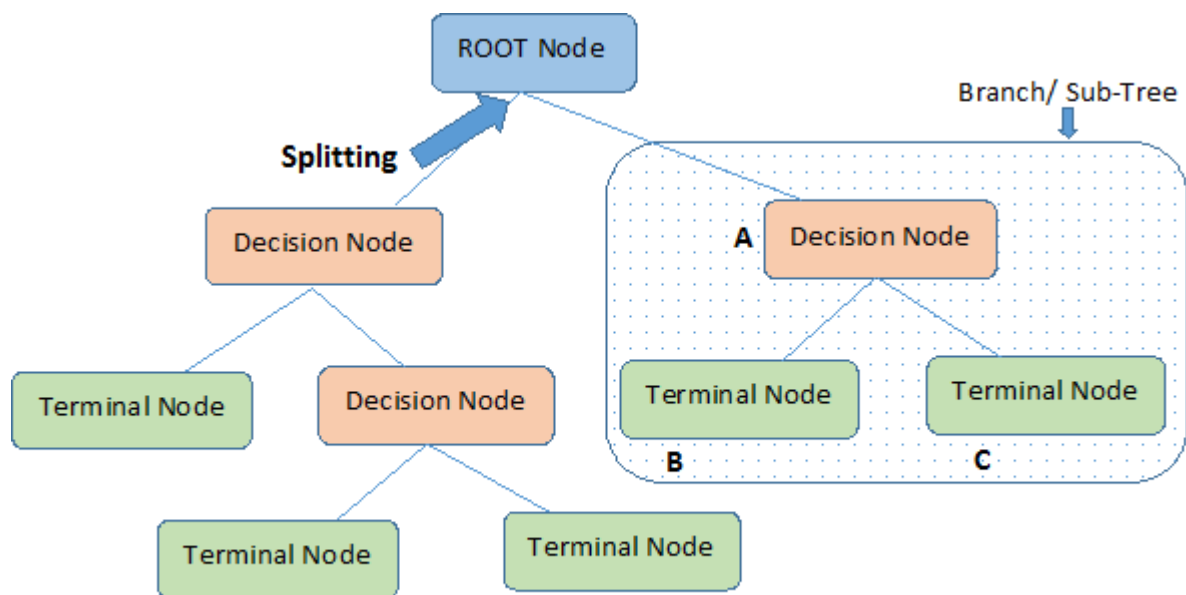
Figure 3. 4: Decision tree

Decision tree transforms classification techniques into the shape of a tree structure. It breaks and simplifies a datasets into smaller subsets then at the same process a related decision tree is incrementally developed proceeding from the root. The output is a tree that has leaf nodes and decision nodes. The decision node represents the branches which can be 2 or more then the Leaf node signifies a decision or classification. The base node in a tree called root node. An advantage is it can process numerical data in addition with categorical data. (Aedsayad)

3.2.1.1 Important Terminology related to Decision Trees

- **Root Node:** It represents the whole population or pattern and this further gets broken into two or more sets.
- **Splitting:** Is the method of breaking down a node into 2 or multiple sub nodes.

- **Decision Node:** This is the breaking down of a sub node and divided into even more into sub-nodes.
- **Leaf Node:** These nodes do not divide is referred to as Leaf or Terminal node.
- **Pruning:** When we subtract sub-nodes of a decision node, this method is known as pruning could further be said that it is the opposite process of splitting.
- **Branch / Sub-Tree:** A sub portion of complete tree is known as department and sub-tree.
- **Parent and Child Node:** A node, as is broken into sub-nodes and referred to as parent node of a sub-nodes while the sub-nodes are the child of a parent node.



Note:- A is parent node of B and C.

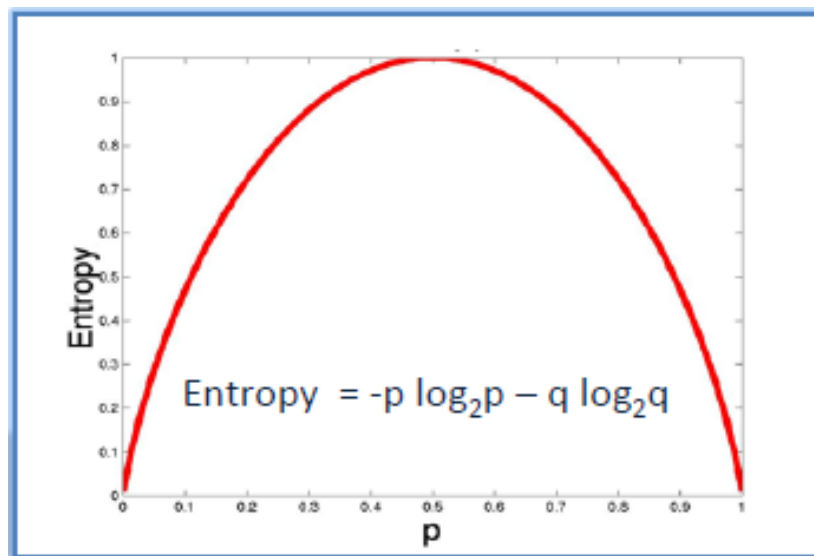
Figure 3. 5: Decision tree nodes

3.2.1.2 Algorithm

The core feature for setting up decision trees is known as ID3 by J. R. Quinlan who employs a top-down, inquiries through the areas of possible branches with no backtracking. ID3 utilizes Information Gain and Entropy to assemble a decision tree (Aedsayad).

3.2.1.3 Entropy

A decision tree starts from a root node and includes splitting the data within subsets that include instances with similar values. ID3 algorithm utilizes entropy to compute the homogeneity of a sample. If the pattern is absolutely homogeneous the entropy is zero and if the sample is an equally splitted it has entropy of one.



$$Entropy = -0.5 \log_2 0.5 - 0.5 \log_2 0.5 = 1$$

Figure 3. 6: Entropy Diagram

Entropy Frequency Calculation is as follows:

- a) Using Entropy frequency table of an attribute:

$$E(S) = \sum_{i=1}^c -P_i \log_2 P_i \quad (3.1)$$

- b) Using Entropy frequency table of 2 attributes:

$$E(T, X) = \sum_{c \in 1} -P(c)E(c) \quad (3.2)$$

3.2.1.4 The information gain

After the splitting of a dataset feature, the reduction entropy is called entropy Information Gain. Setting up a decision tree is entirely about finding features that returns the best possible information gain. It measures the relative change in entropy together with respect to the independent variables (Aedsayad).

$$Gain(T, X) = Entropy - Entropy(T, X) \quad (3.3)$$

3.2.1.5 Advantages of Decision Trees

- Decision trees are convenient to interpret.
- To setup decision trees requires minimal data preparation from the user- there is no reason to normalize data.

3.2.1.6 Disadvantages of Decision Trees

- Decision trees are probable to overfit noisy data. The chance of over fitting on noise raises as a tree expands deeper.

3.2.2 Random forest

They are famous approaches that are utilized to build predictive classification models problems. Random forest techniques create a complete forest that has random and uncorrelated decisions trees to get to the best possible result.

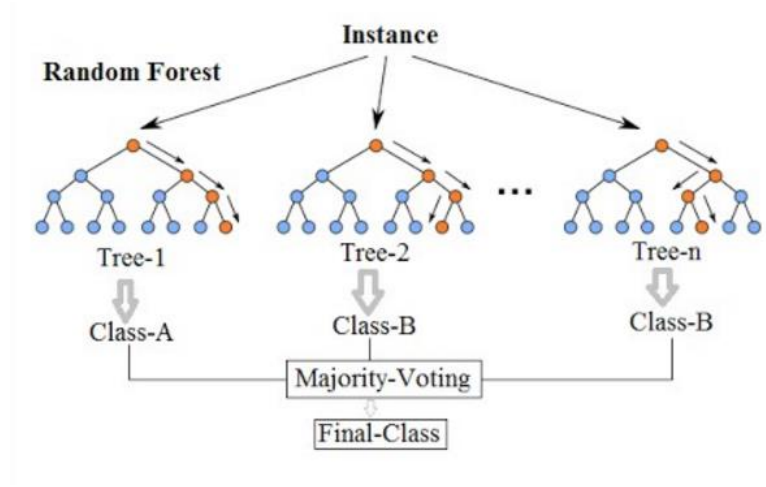


Figure 3. 7: Random forest Diagram

In the case of regression problems, Random Forests are created by growing trees that has the ability of producing numerical attributes, the mean square error formula for Random Forest is stated below.

$$\text{mean error} = (\text{observed} - \text{tree response})^2 \quad (3.4)$$

The total predictions of the Random Forest are summed and regarded as the average tress predictions.

$$\text{Random Forest Predictions} = \frac{1}{K} \sum_{k=1}^K K^{th} \text{ Tree Response} \quad (3.5)$$

Random Forest Predictions: The k index goes over each distinct tree in the forest.

3.2.3 KNN

This algorithm is robust, known for simplicity and it is a very capable classifier widely used as a template for more advanced classifiers like SVM and ANN. On a lot of cases K-nearest neighbor K-nearest neighbor (KNN) can perform better than some advanced classifiers in a wide array of applications (Zakka, 2016).

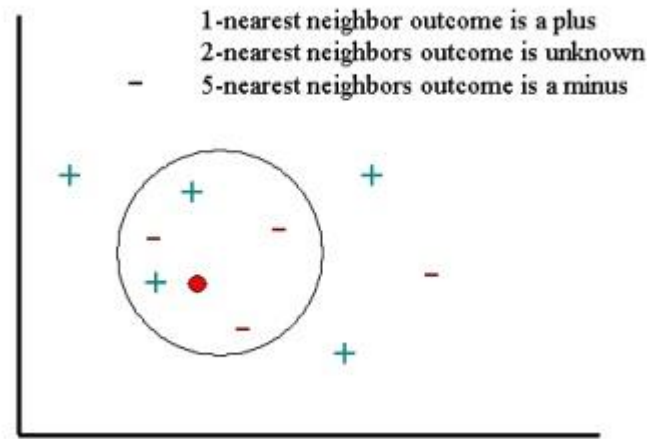


Figure 3. 8: KNN algorithm Diagram (Zakka, 2016).

In the classification settings, the KNN technique basically focuses on building most of the votes between the K most similar instances to a declared “unseen” observation. Similarity is defined in accordance with an association metric among a pair of 2 data points. A popular option is Euclidean distance formula is below (Zakka, 2016).

$$d(x, x^1) = \sqrt{(x_1 - x_1^1)^2 + (x_2 - x_1^1)^2 + \dots + (x_n - x_n^1)^2} \quad (3.6)$$

Though a different method may be required for other applications that may require Chebyshev and Hamming distance. KNN classifier steps are as follows:

- It runs over the complete dataset computing d within x and each training observation. We'll name the K factors in the training records that are closest to x the set A . Note that K is typically different to stop tie situations.
- It afterwards estimates the likelihood for each class fraction on the factors of A .

Given the label class. (The Indicator function that classifies 1 when factor x is true is $I(x)$)

$$P(y = j|X = x) = \frac{1}{K} \sum_{K \in 1} I(y^{(i)} = i) \quad (3.7)$$

Then finally, it will be assigned to a class that has the highest expectation.

3.2.3.1 Cross-Validation

Cross-validation is a solid method that can be utilized to acquire estimates of model parameters that are unknown. The common concept of this approach is to partition the information sample into a wide variety of v folds.

A choice and smarter method includes estimating the test error degree by holding out a subset of the training set from the fitting process. This subset, is known as the validation set, can be utilized to get the best possible degree of flexibility of our algorithm! There are various validation strategies that are utilized in practice, and we will keep exploring one of the more famous ones known as k -fold cross validation (Zakka, 2016).

Cross-validation: How it works?

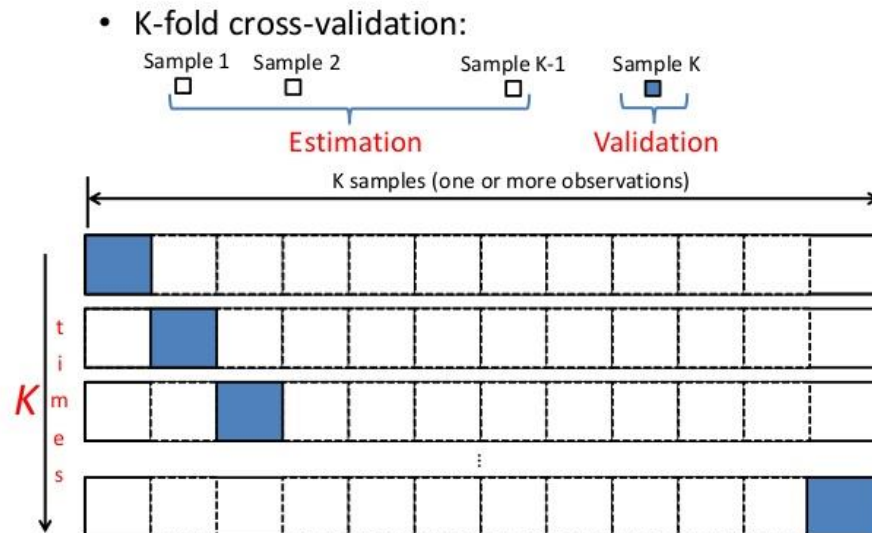


Figure 3. 9: Cross-Validation (Zakka, 2016)

As viewed in the data, k-fold cross validation (the k is absolutely unrelated to K) includes randomly splitting the training set into k groups, or folds, of roughly same size. The initial fold is considered as a validation set, and the technique is fit on the residual k-1 folds. The misclassification rate is afterwards computed on the observations in the held-out fold. This procedure is redone k times; every time, a unique group of observations is handled as a validation set. This system outcome in k estimates of the test error which are then averaged out.

KNN algorithm is among the easiest classification algorithms. Even including such simplicity, has the ability to assign incredibly competitive results. KNN algorithm can additionally be utilized for regression problems.

3.2.4 SVM

SVM is short for Support Vector Machine; this supervised machine learning method can be deployed for regression and classification tasks. SVMs are primarily based on discovering a hyperplane that best split a dataset in two classes, as displayed below (KDnuggets, 2016).

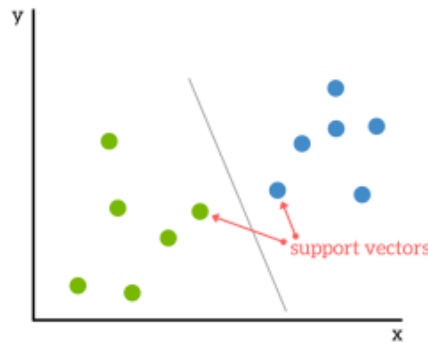


Figure 3.10: Support Vector Machine (KDnuggets, 2016)

3.2.4.1 Support Vectors

These are the information points that are nearest to the hyperplane, if the point of a dataset is removed, it would change the role of the splitting hyperplane. Based on this, it is a necessary factor of a data set.

3.2.4.2 What is a hyperplane?

Hyperplane separates and classifies a group of data in a linear method. Intuitively, the farther from the hyperplane the information points lie, the more assured that it has been effectively classified. Therefore the data points are needed to be distant from the hyperplane as much possible, while it is still on the mathematic aspect of it. So when new sample information is added, no matter the region of the hyperplane it lands will determine the type that we entrust to it (KDnuggets, 2016).

3.2.4.2.1 Pros

- Highly compactable on smaller cleaner datasets, Accuracy.

- Efficient in high dimensional regions.

3.2.4.2.2 Cons

- Low performance on larger datasets as the time needed for training might be high.
- Low efficiency on noisier datasets.

3.2.5 Logistic Regression

Logistic regression is a statistical technique for examining a dataset in as there are one or greater independent variables that decides an outcome. The result is calculated with a dichotomous variable that produces two viable outcomes. It is utilized to forecast a binary result on a group of independent variables. To signify binary / grouped outcome sample variables is used. In additionally logistic regression can be thought of as as a unique case of linear regression when the result variable is categorical, where we are utilizing log of odds as dependent variable. In easy terms, it predicts the likelihood of an incidence event by fitting information to a logit function (Chandrayan, 2017).

3.2.5.1 Derivation of Logistic Regression Equation

Logistic Regression is an element of a bigger class of algorithms acknowledged as Generalized Linear Model. In 1972, Nelder and Wedderburn proposed this technique along with an endeavor to supply a means of using linear regression to solve issues which does not suit the linear regression application. Infact, they proposed a category of unique models which are Poisson Regression (Analytics_Vidhya_Content_Team, 2015).

The vital equation of generalized linear technique is:

$$g(E(y)) = \alpha + \beta x_1 + \gamma x_2 \quad (3.8)$$

Whenever the log regarding the odd ratio is discovered to be positive, the chances of success are usually higher than 50%. A traditional logistic algorithm plot is as shown below. You be able see probability in no way goes beneath 0 and above 1.

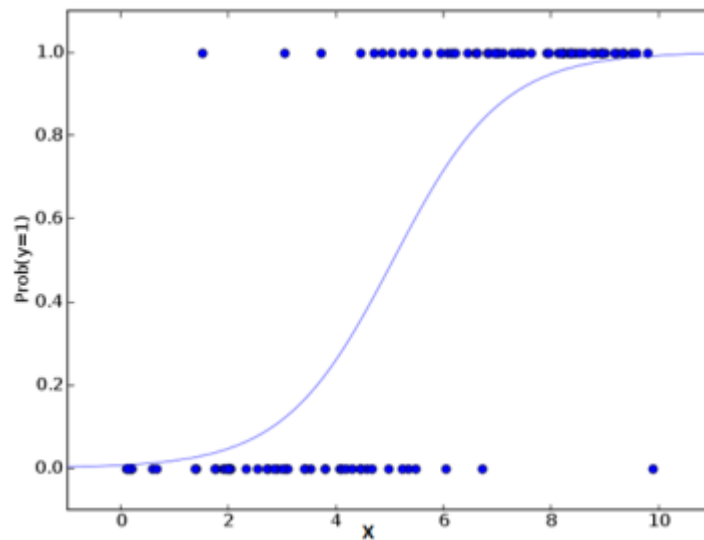


Figure 3.11: Logistic Regression (Analytics_Vidhya_Content_Team, 2015)

3.2.6 Neural Network

3.2.6.1 Overview of Artificial Neural Networks

The origin of neural network was from the previous century. Scientists of that era were attempting to discover an artificial resemblance of the organic brain that is capable of carrying out complicated tasks like recognition/realization and learning. Actually, these thoughts were not recent ideas but they have been mentioned in the writing of great thinkers like Aristotle, Plato and others. The first publication claiming the concept of neural networks in recent studies was posted in the Nineteen Forties and written by McCulloch and Pitts. It used to be an easy neuron application that can output binary signal. Just in the subsequent years later, the concept attracted lots of researchers to start commence work on neural networks. One scientist known as Hebb has mentioned a revolution learning algorithm that aided the foundation of the neural networks algorithms. This algorithm was clearly mentioned in 1949 (Hebb, 1949) . In

1954, Farley and Clark were the initial people to do use computational machineries, to simulation a Hebbian network in the Massachusetts Institute of Technology (MIT), these machines were referred to as calculators. Rochester, Duda, Habit and Holland built other neural networks calculation machines in the year 1956. By the 1962, Rosenblatt succeeded to set up a learning algorithm that converges usually to minimal error. The loop continues updating the weights until a proper group of the required outputs is generated. However, the computer systems of that generation were not appropriate to compute enormous calculations required by such algorithms. This reality really affected the growth of neural networks of that time. The implementation of singular layer networks was ineffective in developing options for various scientific problems. The multilayer networks were believed to be the answer for complicated issues and in a position to offer great performance. Unfortunately, there were no efficient learning algorithms that are able to supply convergence while training these multiple layered networks.

In 1975, the cognition system was setup by Fukushima and it is the first sample of multilayer neural network that has an effective training algorithm. The formation of the system weights are modified between a pattern of the neural and another. Each new system's energy points has pros and cons. Some networks has the ability to broaden the data in one path while other structures push information forward and backward firing various activation functions. Hopfield's networks invented in 1982 and had the capacity to produce data in bidirectional mode (Minsky & Papert, 1969). The implementation of the back propagation ANN was typically the primary inspiration for the integration of the artificial neural networks in 1986. That algorithm has proposed the generation of an error signal via the distinctive layers of the neural network. The propagated failure was then utilized to recalculate the new weight factor values in an ANN. A stochastic gradient descent algorithm was used to commence training in the neural structure (Anderson & McNeill, Artificial Neural Networks Technology, 2010). The back propagation algorithm attracted lots of interest as there was tons of argument on the feasibility of implementing such an algorithm in an artificial intelligence or not. This debate

was generally a result of the ambiguous notion of the training of the network at that time. The concept of utilizing a target signal in a training procedure was a bit complicated and not clear. However, in the previous decade various unsupervised learning techniques were learnt and then proposed for individual or multiple layer neural networks. Such strategies play a major role to discover transitional variations even when the preferred signal is not present (Rumelhart & McClelland, 1986) .

3.2.6.2 Principles of ANN

Flow of communication in biological neurons is regarded as one of the most complicated procedures that occur in the synapses of biological cells. Chemical elements are being discharged from the sender cell through synapses to the recipient cell. The recipient cell responds by developing an electric powered potential whose energy is an element of the chemical substance. This electric potential progresses into the body of the cell. The neuron activates if its potential is high and reaches a threshold value. The active neurons generate another signal and forward it to the subsequent neuron resulting in its activation. This easy idea is applied in developing artificial models of the ANN structures.

illustration of the notion of using artificial neural networks. The inputs denoted ($x_1, x_2, x_3 \dots$) are acquired through the neurons denoted ($\omega_1, \omega_2, \dots$). These alerts are accumulated and weighted by the neurons. The neurons then put up all the accumulated inputs to a summing junction whose characteristic is to determine activation is possible or not. In ANNs, the summing junction includes several applications that are utilized on the acquired signals (Hebb, 1949).

3.2.6.3 ANN

ANNs are pc applications that are cued biologically to emulate some fundamental duties of the human intelligence by using training algorithms that are able to understand from experience. They are systems made of integrated flexible easy processing factors (known as artificial

neurons or nodes) that are have the capacity of executing vast comparison computations for executing data and knowledge representation. ANNs gain knowledge through experience with the good training examples as people do and not from traditional pc programs. ANNs have data processing features of biological brains like nonlinearity, learning, errors and tolerance of fault, robustness, high comparison and capability to generalize. Therefore, ANNs are used in fixing complicated real life issues like pattern classification, approximation and optimization (Sun Y1, 2003).

Table 3.1: Comparison between Traditional Computing and ANN

Features	Typical Computing	Artificial Neural Networks
Learning Rule	Rules	Experience
Functions	Logically	Perceptual pattern
Method of processing	Sequential	Parallel

Various ANN systems were created for several unique applications. ANN techniques can be supervised or unsupervised based about the learning (training) algorithm. The input or output information sets are made available to the ANN techniques for supervised learning while only the input information is provided to the ANN technique in unsupervised learning which learns to understand the pattern/shape in the data. ANN can additionally be categorized based on topology as feed forward and feedback. The link of neurons does not create circles in feed forward architecture. On this technique there is no link from the output to the input neurons and therefore the file of preceding result values are not available. In Feedback ANN system coalition between nodes consists of circles. The output of one layer routes returned to the input of identical layer or preceding layer. Feedback models are typically very tough to train than the feed forward (Sun Y1, 2003).

3.2.6.4 Artificial Neuron

This is the essential part of the artificial neural network that is engineered to mimic/reproduce the applications of natural neuron. Inputs signal instances the connection measure are first combined (summed) and afterwards transfer the features to the preferred output of that specific neuron. The activation feature is the weighted aggregate of neuron's inputs and sigmoid feature is usually used (Kustrin & Beresford, 2000). Artificial neurons or nodes are the solid base of ANN that processes data based on weighted inputs using transfer features and outputs. Adjacent layers neurons are fully or partly connected with weighted links. Net input in a neuron is seen below.

$$\text{Net Input to a neuron} = \sum_1 x_1 w_1 \quad (3.9)$$

where w_1 = the weight factor

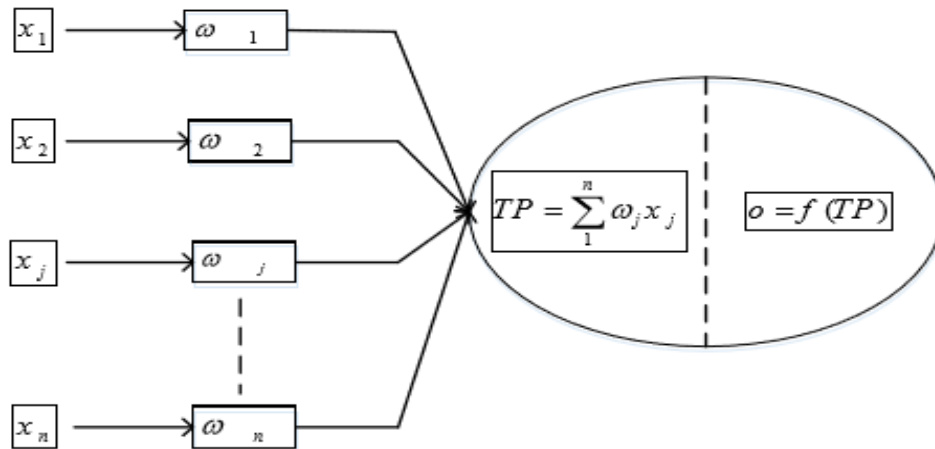


Figure 3.12: Artificial Neurons Structure (Kustrin & Beresford, 2000)

The artificial neuron illustrated in the image above includes a fair variety of inputs. Each 1 of these inputs is related to a dynamic weight. The weights are utilized to analyze the inputs and regulate their strengths in the development of the neural reaction. These weights are used to analyze the result of the neuron. The formula of the result is described.

3.2.6.5 Components of Artificial Neuron (Bias)

A bias upgrades the neural network performance. It serves as a form of weight on a link from a group that usually has activation feature of 1. The same method to initialization of weights, bias should be initialized to either 0 or any precise value primarily based on neural net. If bias is present the net input is attached as:

$$\text{Net} = b + \sum x_i w_i \quad (3.10)$$

Here: Net=net input, b=bias, x_i =input from neuron i and w_i =weight of neuron i to the output neuron.

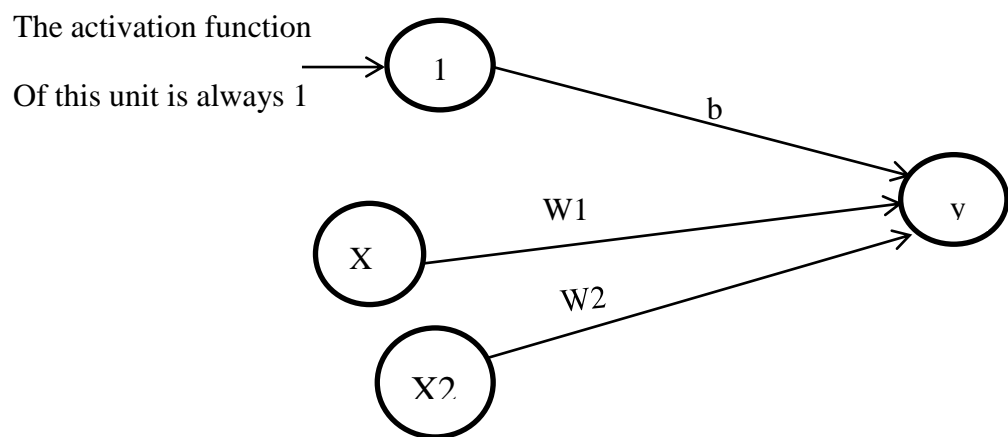


Figure 3.4: A Simple Network (bias)

3.2.6.6 Weighting factors

Artificial neuron normally receives many input variables at same time. A Particular input possesses its own weight that gives it the impact it requires on the summation function. Some inputs are designed to be more essential than others so as to have high impact on the neuron as they join together to give a neural output. The weights used on the different layers exert more influence in the function of neural network. Steps below are taken when choosing the weights:

- Run the network with one set of weights
- Run the network again with new sets of weights after modifying some or all the weights
- The process is repeated until some predetermined goal is achieved.

3.2.6.7 Summation Function

The preliminary step of neural network system is computing the weighted amount of all inputs after neuron. Mathematically, the inputs information and the subsequent weights are like vectors that can be expressed as $(I_1, I_2 \dots I_N)$ and $(W_1, W_2 \dots W_N)$ respectively. Each element of I vector is increased by the respective factor of W vector and then summing up all the entities to find the summation function.

Example

$$\text{Input1} = I_1 \times W_1$$

Input2 = $I_2 \times W_2$ etc are added as

$$\text{Input1} + \text{Input2} + \dots + \text{InputN}$$

Single number not multi-element vector is the result

3.2.6.8 Transfer function

Every neuron is allocated a transfer characteristic which determines the result values. Summation feature result worth is transformed to working result using a logarithmic method known as the transfer function. The summation amount can be in contrast to locate the neural output. Some transfer capabilities used in ANN are like PURELIN , TANSIG and LOGSIG functions. LOGSIG transfer feature is broadly used for non-linear applications to enter and result values. The LOGSIG is expressed as:

3.2.6.9 Output function

Each neuron usually has some result signal that it can transfer forward to a lot of other neurons which is similar to organic neuron in which there are numerous inputs however only one output. The result value is same as the result of transfer function.

$$y^{\text{Sigmoid}} = \frac{1}{1 + e^{-x}} \quad (3.11)$$

3.2.6.10 Error function and back propagated value

Variations of expected and predicted values are analyzed in lots learning architectures. This virtue is changed by goal failure function to be a reproduction of a specific architecture. This failure is used directly by a lot of networks but some figure it, others cube it while the raw failure output is modified by other paradigms operated on their reasons (Anderson & McNeill, 1992).

3.2.6.11 Keras

Keras is a solid Python ML library that is used for the evaluation and development of ML algorithm. It is fully featured with effective numerical calculation libraries CNTK, Theano and TensorFlow. The benefit of this is usually that one can begin with neural networks in an easy way.

Also keras is a simple methon to utilize neural network library that promotes an easy and intuitive syntax. It is additionally geared towards neural network systems users as opposed to

producers, at least to some degree. The block among these 2 organizations is not precisely well-defined; what, exactly, constitutes the layout and implementation of a machine learning algorithm against enhancing an already-prepared one? This is egg versus chicken stuff, and not actually an excellent discussion subject for here. However, I will perform the assertion that, possibly more than other established, mainstream neural network library, Keras is ideally suited to the work of information science. In fact, I would say to that amount Keras is a must have tool into the toolbox of all data scientist working with neural networks.

CHAPTER 4

SYSTEM RESULTS AND DISCUSSIONS

4.1 Methodology Introduction

This chapter is designed to discuss about the applied technologies, application process and the results obtained during this study. A major function of this thesis was to compare the efficiency of various machine learning techniques with relations to immunotherapy dataset classification. Six variants of ML techniques are to be deployed separately on the processed dataset to determine the performance of the different techniques.

4.2 Tools Used

4.2.1 Python:

This programming language is object-oriented very robust, with thanks to Guido van Rossum. It has an extensive range of purposes from Web, Computer GUI, to mathematical and scientific computing applications. It's a lot simpler to study and compose Python applications in contrast to other top languages like: C#, java, C++. Python makes programming enjoyable and permits users to be more productive (Programiz).

4.2.1.1 Python Libraries.

Below are the Python Machine Learning Libraries used in this study;

- Pandas
- Keras
- Statistics
- Pydotplus
- GraphViz
- Scikit_learn
- Num_Py
- TensorFlow
- Pandoc

4.2.2 Jupyter Notebook:

This is a unique kind of web application that permits creation and modification of files that include visualizations, live code and equations. The uses include: numerical simulation, information visualization, machine learning, statistical modeling, information transformation and tons more, this Notebook is open source (DADOUCHE, 2018).

The Jupyter Notebook documents are easily readable files created through the Jupyter Notebook Application which includes both rich text properties code python codes and results.

4.2.3 Computer:

These are the specifications of the computer used in this research; HP Pavilion 8GB Ram, Core i5 Quad Core Processor, 2GB NVidia GeForce Graphics Card.

4.2.4 Dataset:

The immunotherapy dataset was gotten from the UCI dataset repository as discussed in Chapter 2. The UCI dataset consists of 90 instances, 8 attributes and other Characteristics.

In the quest to get high accuracy, the dataset were normalized/rescaled to the range of [0, 1]. The bullets below summarized the immunotherapy dataset attributes used for the classification task.

- Sex
- Age
- Time
- Number_of_Warts
- Type
- Area,
- induration_diameter
- Result_of_Treatment

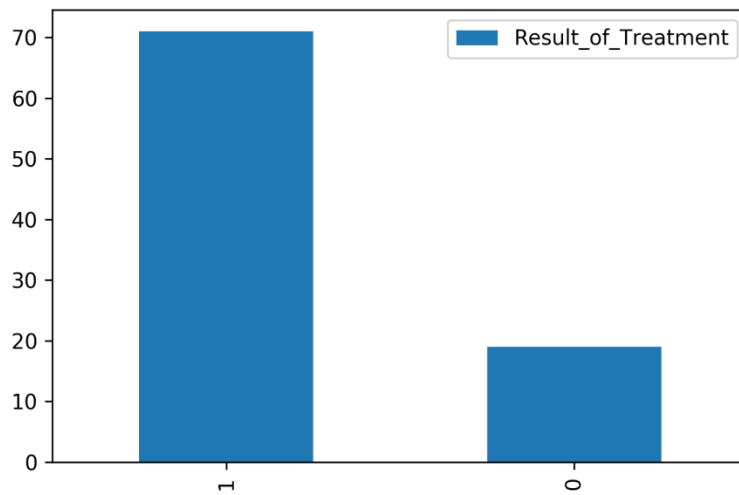


Figure 4.1: Treatment Result

The picture above shows the result of cancer patient's immunotherapy treatment with 70% success rate as retrieved from the dataset.

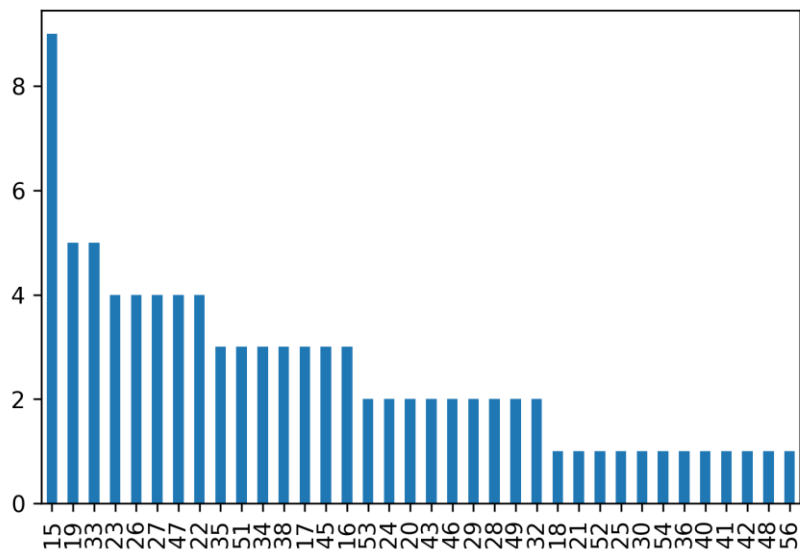


Figure 4.2: Age Range

Figure 4.2 depicts the age range of patients compile with the occurrence form the statistics age 15 occurred the most at 9 times.

4.3 Algorithms Overview

These are the fundamental elements of smart system applications. It analyzes and gathers intelligence from a given data. The steps below show the whole training process from start to finish.

4.3.1 Step1 (Filtering)

The first step starts with data preprocessing which is a vital step in data processing, it includes: missing data checks and filtering which makes the chunk of data prepared and flawless. When a dataset has not been filtered it may lead to lots of faulty models or even wrong outputs. Hence, the representation of data is an important step in vetting the quality of data before moving to the next step.

4.3.2 Step2 (Normalization)

Data normalization: The sklearn.preprocessing bundle offers numerous common application services and transformer classes to change raw feature vectors into a representation that is more appropriate for the downstream estimators/techniques. If a fraction of outliers are available in the set, powerful scalars or transformers are far more appropriate.

4.3.3 Step 3 (Cross-Validation)

For the general purpose of classification that is applicable to all techniques, the dataset from UCI repository was splitted into two parts the training set, at which 75% of the datasets was taken as training set and then the testing set, which the remaining 25% of the set is taken as the testing set, thus a globally cross validated data was archived which is to be used in all techniques. In this method prior to the commencement of the dataset training, a part of the data is removed then if the training was successfully completed, the initially removed part of the dataset may be used to test techniques on the newly modified datasets. An array of techniques used for assessment of models that used these same basic thoughts is called cross validation.

The approximator feature foretells the target value of the dataset that is still at the testing phase.

4.4 Step4 (Dataset Classification)

This is the fourth step, it is meant to analyze the already processed data using; Decision tree, Random forest, KNN, SVM, Logistic Regression, and CNN techniques for the classification of immunotherapy dataset also the average of multiple iterations was taken due to the fact that result is different on every run. The outcome of the above techniques indicates that the KNN technique outperformed other techniques at 85%. The flow chart below shows the whole activities.

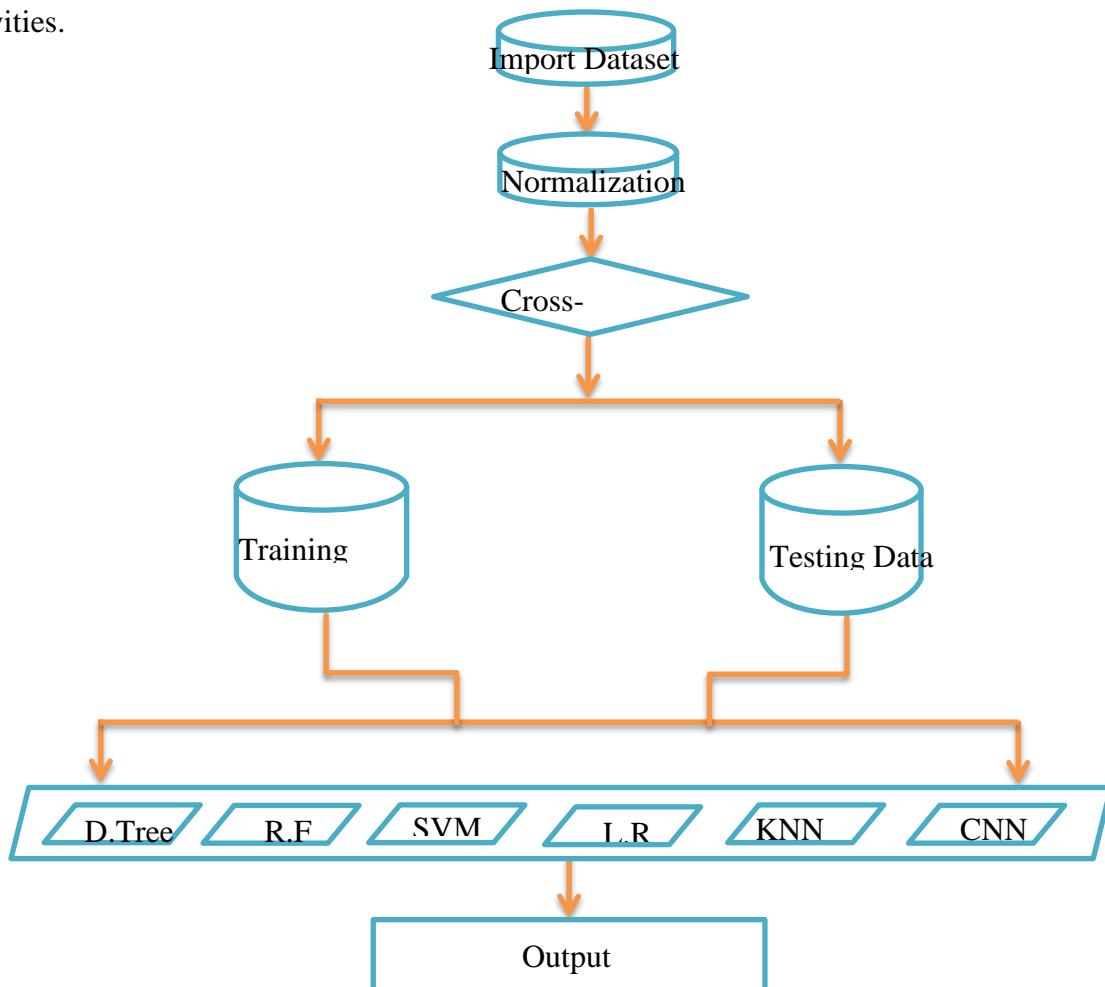


Figure 4.3: Flow Chart of the Entire Classification Process.

4.5 Decision Tree Technique:

Decision tree is a graphical drawing (boxes) of an algorithm, illustrating all feasible decisions, costs, utility, and consequences of an issue, and also permits for comparison of such choices in one pane (Nayab & Scheid, 2011).

4.5.1 Advantages

- **Flexibility:** It can handle items that have mixed data.
- **Specificity:** Allows specific assignments of values.
- **Resilience:** efficient with little degree of errors.
- **Transparency** : it has a transparent nature
- **Ease to Use:** simplifies data output in an uncomplicated pictorial format.
- **Comprehensive:** Allows complete evaluation of the consequences of every possible decision, additionally approves for partitioning data in a much deeper degree.

4.5.2 Decision Tree Training

This classification system was trained with 90 instances that's 100% of the dataset as gotten from our globally processed cross validated data, the training was ran 10 times and the mean score was taken(81.31%) also the tree was generated as seen in figure 4.4.

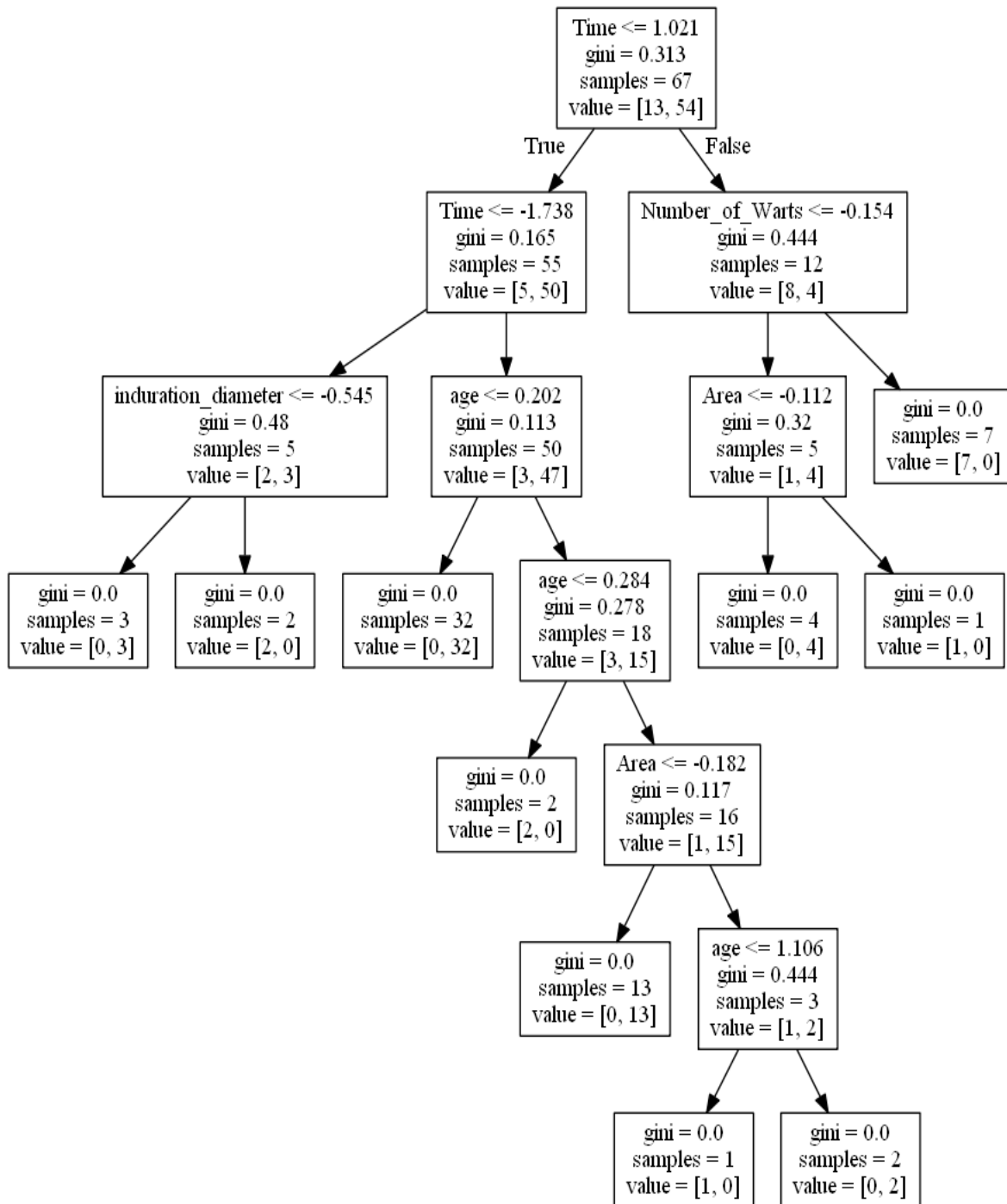


Figure 4.4: Generated Decision Tree

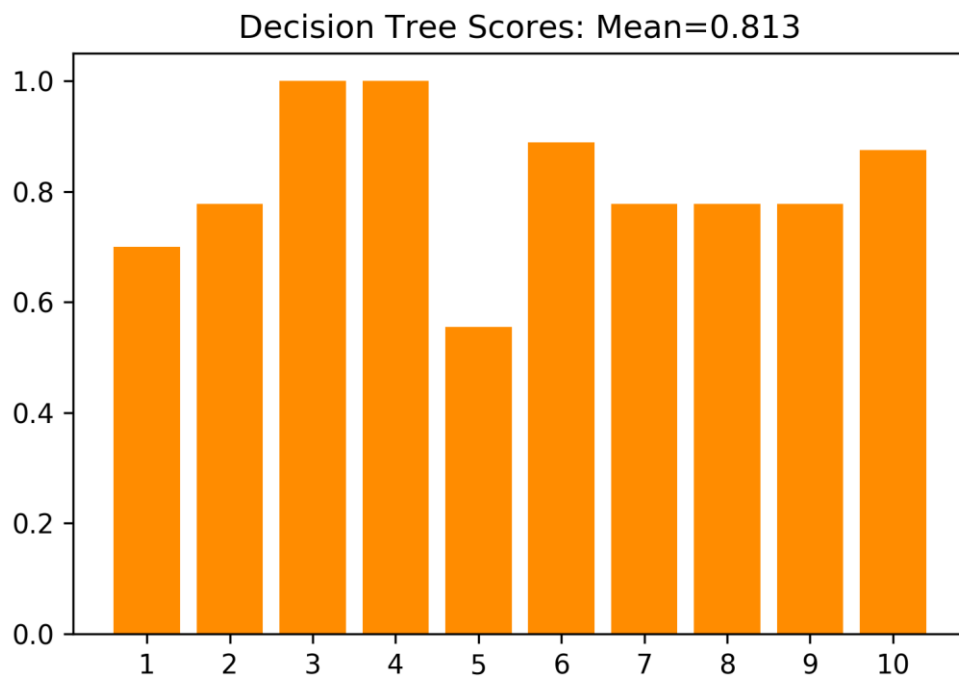


Figure 4.5: Decision Tree Results

Table 4.1: Decision Tree Training Parameters

Parameters	Value
Training Samples	90
Iterations	10
Score	81.31%
Mean Square Error	0.174
Root Mean Square Error	0.417
Duration	1.7sec

4.6 Random Forest Classification:

This is an ensemble of classifiers that comprises of dense selection of trees. Advantages of Random Forests are as follows (Gupta, 20013).

- Runs effectively on huge information bases
- Accuracy
- Gives estimates of what variables are essential in the classification
- Prototypes are computed that deliver data about the relation between the variables and the classification.
- Computes proximities into pairs of instances that can be utilized in clustering, finding outliers, then (by scaling) assign interesting views of the data
- Handles lots of input variables besides variable deletion
- Provides techniques for balancing error in classification load unequal data sets
- Generated forests can be protected for future usage on other data
- Generates an internal impartial calculation of the generalization error as the forest developing progresses
- Provides efficient strategies for estimating displaced data
- Maintains precision when a massive percentage of the information are missing

4.6.1 Random Forest Training

The developed random forest classification system used the same globally processed data, after series of training, a total of 13 forests yielded the best results on 10 iterations and the mean score was documented (83.56%).

Table 4.2: Random forest Tanning Parameters

Parameters	Value
Training Samples	90
Forests	13
Iterations	10
Score	83.56%
Mean Square Error	0.173
Root Mean Square Error	0.416
Duration	2.17sec

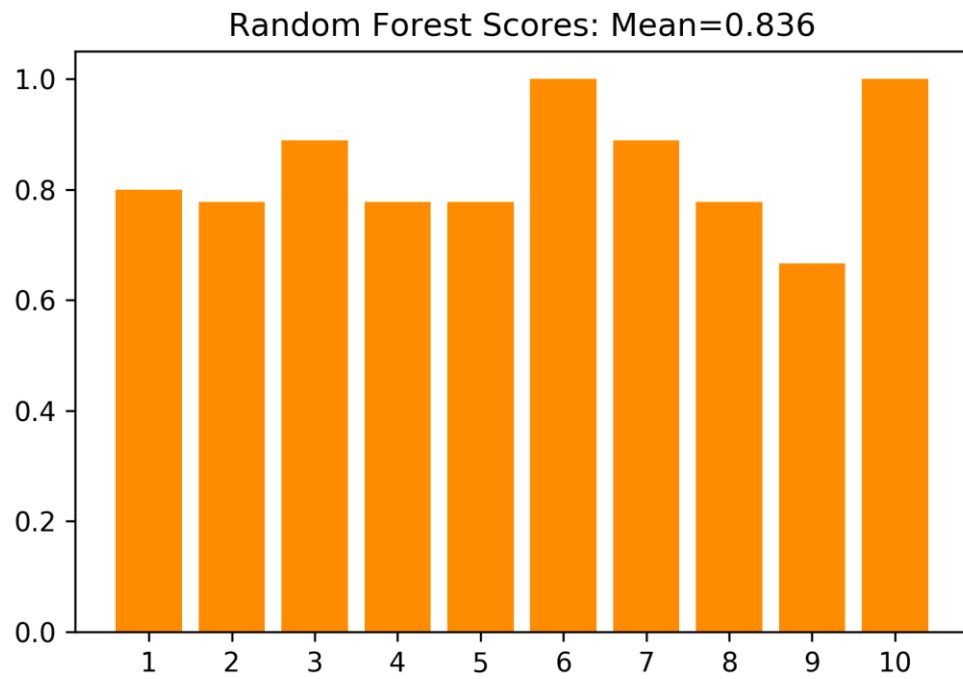


Figure 4.6: Random forest Results

4.7 KNN Classification:

The KNN Classifier is a simple classifier that performs well on easy recognition problems; then again it can be gradual for real-time calculation if there are a lot of training examples and is not powerful the incase of noisy data (Nickgillian, 2014).

- Well optimized for training data
- Effective if the training information is large

4.7.1 KNN Training

During the training the globally processed data was utilized but for specific method 8 n_neighbors feature attribute was used alongside 10 iterations as it appears to yield highest accuracy. The mean score is 84.67% that's approximately 85% which happens to be the best performing technique.

Table 4.3: KNN Tanning Parameters

Parameters	Value
Training Samples	90
Neighbors	8
Iterations	10
Score	84.67%
Mean Square Error	0.261
Root Mean Square Error	0.51
Duration	0.89sec

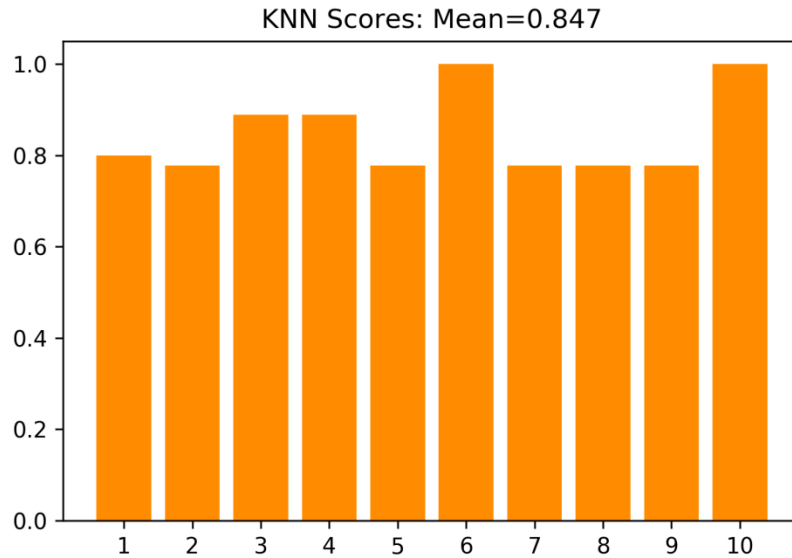


Figure 4.7: KNN Results

4.8 SVM Classification

SVMs are more frequently deployed in classification tasks that are based on the concept of discovering a hyperplane that excellently divides a dataset into two classes (Bambrick, 2016),

- SVM's are absolutely excellent when data elements are unknown.
- Efficient together with semi-structured and also unstructured data, example text, Images and trees.
- It measures relatively properly to excessive dimensional data.
- SVM techniques have generalization in practice; the danger of over fitting is less with SVM.

4.8.1 SVM Training

The SVM Kernel comprises of 4 types Linear, Rbf, Sigmoid and Poly, The classification was fully carried out on all 4 kernels, Rbf Kernel was higher at 81.33% as shown on Table 4.4 and Figure 4.8, for this reason the Rbf Kernel was selected to represent SVM at the final comparison with other ML techniques.

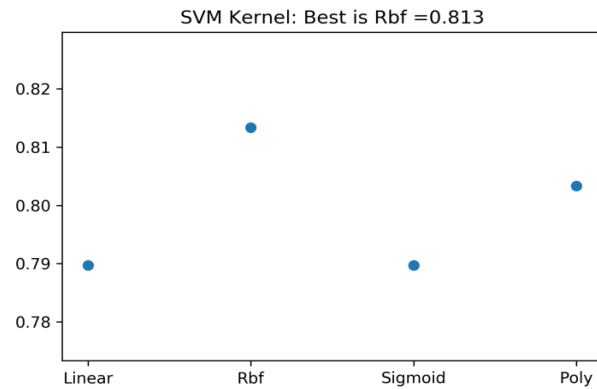


Figure 4.8: Results from all SVM Kernel

Table 4.4: All SVM Kernel Tanning Result

Support Vector Machine	Score
RBF Kernel	81.33%
Poly Kernel	80.33%
Linear Kernel	78.97%
Sigmoid Kernel	78.97%

Table 4.5 SVM RBF Kernel Tanning

Parameters	Value
Training Samples	90
Iterations	10
Score	81.33%
Mean Square Error	0.26
Root Mean Square Error	0.51
Duration	1.11sec

4.9 Logistic Regression Classification and Training:

Logistic regression utilizes regression to forecast the output result of a categorically based variable. The probable results of an individual trial are modeled as a feature of the explanatory variable that is utilizing a logistic function. Logistic modeling is done on categorical information which may be of different types including binary and nominal (Researchoptimus), best results was achieved at 15 iterations 81.56%.

Table 4.6 Logistic Regression Tanning Parameters

Parameters	Value
Training Samples	90
Iterations	15
Score	81.56%
Mean Square Error	0.217
Root Mean Square Error	0.466
Duration	1.11sec

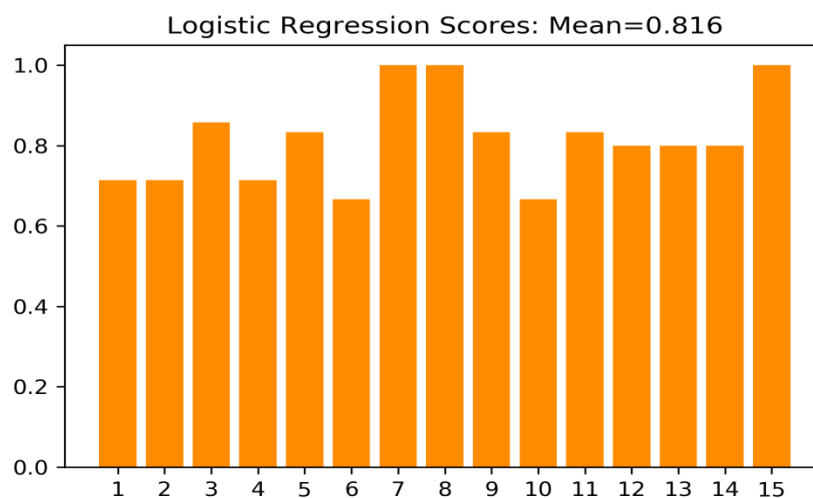


Figure 4.9: Logistic Regression Results

4.10 Convolutional Neural Networks Training:

CNN comprises of neurons that are clever to learn biases and weights. Every neuron gets tangible inputs, executes a dot product and conditionally chooses to accompany it with a non-linearity (Cs231n). During training process 1100 neurons was introduced and 10000 epoch which did take some time ,the result after 10 iterations is 81.11%.

Table 4.7: Convolutional Neural Network Training Parameters

Parameters	Value
Training Samples	90
Iterations	10
Neurons	1100
Epoch	10,000
Activation	Relu
Score	81.11%
Mean Square Error	0.260
Root Mean Square Error	0.51
Duration	24.88sec

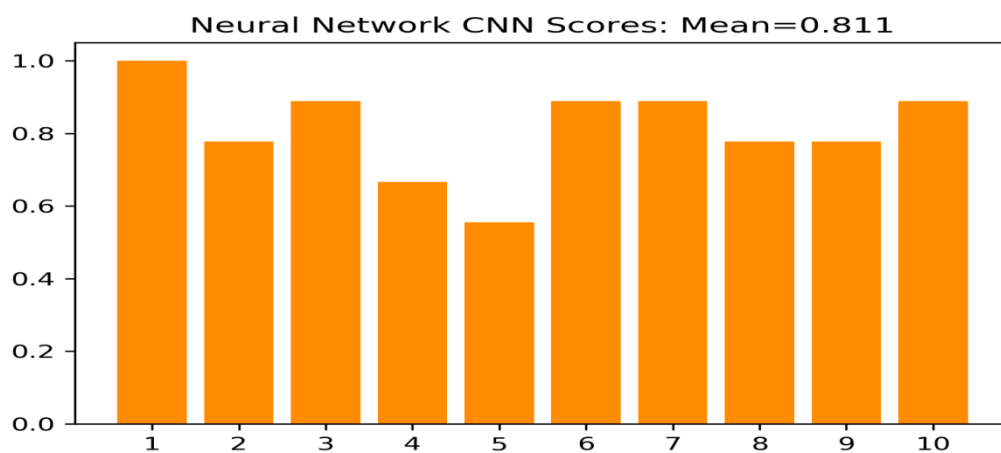


Figure 4.10: Convolutional Neural Network Results

CHAPTER 5

CONCLUSION

Data validation was applied to verify the structural integrity of the data and then Normalization, generally learning algorithms benefit from standardization of the dataset, for the purpose of a global classification data, the dataset from UCI repository was splitted into two parts, the training set, at which 75% of the datasets was taken as training set and then the testing set, which the remaining 25% of the set is taken as the testing set, thus a globally cross validated data was archived which is to be used in all techniques. The classification was executed using all six techniques by deploying the validated data. Comparison between the techniques was performed, in other to discover accuracies as well as to detect the model that performed best in the classification task.

The results acquired from the system were satisfactory, different output values of the accuracy was obtained by varying parameters. The accuracy ranged between 81.11% and 85% in the training, the obtained result illustrates effects of the system's performance. The Table 5.1 shows lists of all the techniques, sorted by “highest first” together with their result values.

Table 5.1: All Techniques Result (Table)

Technique	Value
KNN	84.65%
Random Forest	83.56%
Logistic Regression	81.56%
SVM RBF	81.33%
Decision Tree	81.31%
CNN	81.11%

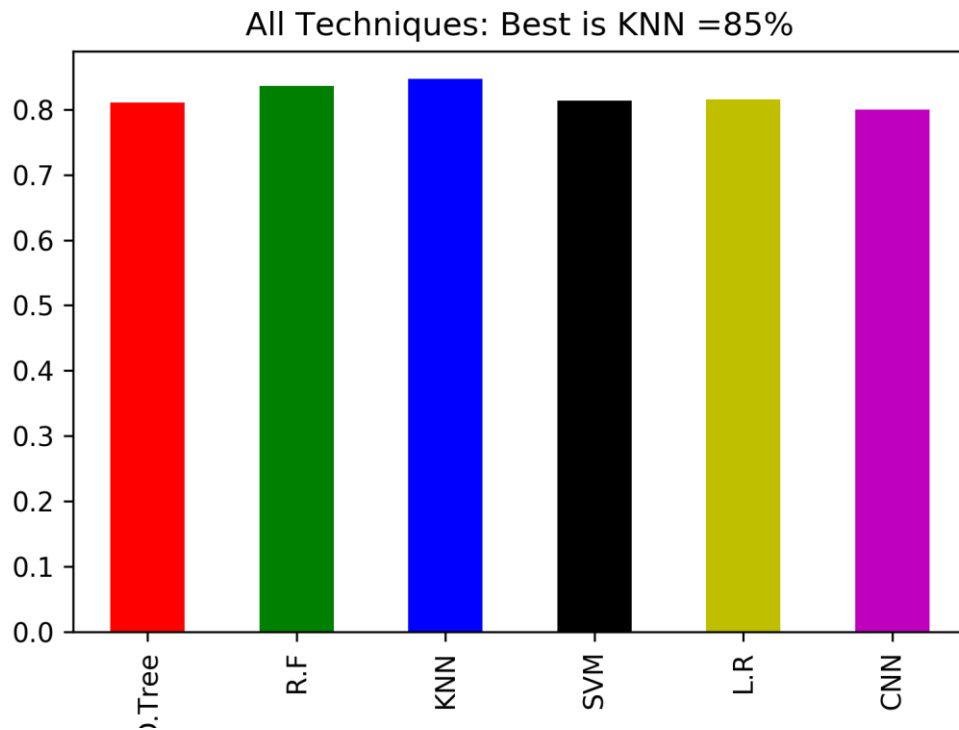


Figure 5.1: Result from All Techniques (Barchart)

Finally, a brief overview of the entire classification tasks shows that of the six selected Machine Learning techniques, K-Nearest Neighbor outperformed the other five techniques and CNN performed least as seen Table 5.1 and Figure 5.1.

Future contributions to this work would feature possible repetition of the classification tasks using other machine learning techniques such as Naive Bayes (NB), extreme learning machines (ELMs), co-adaptive neuro-fuzzy inference system (CANFIS), Convolutional Neural Network (CNN) with more Neurons and Radial basis function network (RBFN) for more efficient results.

REFERENCES

- Aedsayad. (n.d.). *Decision Tree - Classification*. Retrieved April 22, 2018, from Aedsayad: http://www.saedsayad.com/decision_tree.htm
- Aimee E Hammerstrom, D. H. (2011). Cancer Immunotherapy: Sipuleucel-T and Beyond. *researchgate*, 813–828.
- Analytics_Vidhya_Content_Team. (2015, November 1). *Simple Guide to Logistic Regression in R*. Retrieved April 2018, 25, from Analytics Vidhya: <https://www.analyticsvidhya.com/blog/2015/11/beginners-guide-on-logistic-regression-in-r/>
- Anderson, & McNeill. (1992). Artificial Neural Networks Technology. Bataineh, M. (2012). Artificial neural network for studying human performance. *University of Iowa*.
- Anderson, & McNeill. (2010). Artificial Neural Networks Technology. *Rome Laboratory*.
- Anita Thakur, P. G., & Bansal, N. (2016). Symptom & risk factor based diagnosis of Gum diseases using neural network. *IEEE*.
- Avi Schroeder, D. A. (2012). Treating metastatic cancer with nanotechnology. *nature.com*, 39–50.
- Bambrick, N. (2016, June 24). *Support Vector Machines for dummies; A Simple Explanation*. Retrieved April 28, 2018, from AYLIEN | Text Analysis API | Natural Language Processing: <http://blog.aylien.com/support-vector-machines-for-dummies-a-simple/>
- Bisht M., B. S. (2010). Current use and future prospects in cancer therapy. *Indian journal of cancer*, 443-451.
- Boon T, C. P. (2006). Human T cell responses against melanoma. *Annu Rev Immunol*, 175-208.

- Caruana, & Niculescu. (2006). An empirical comparison of supervised learning algorithms. *acm*, 161-168 .
- Chandrayan, P. (2017, August 26). *Machine Learning Part 3 : Logistic Regression*. Retrieved April 2018, 2018, from Towards Data Science: <https://towardsdatascience.com/machine-learning-part-3-logistics-regression-9d890928680f>
- Christopher J Wheeler, K. L. (2009). DCVax-Brain and DC vaccines in the treatment of GBM. *researchgate*, 509-19.
- Claire Chen, A. M.-C. (2016). Deep Learning in Label-free Cell Classification. *Researchgate*.
- Coombes R.C., B. R. (1986). In vitro and in vivo effects of a monoclonal antibody-toxin conjugate for use in autologous bone marrow transplantation for patients with breast cancer. *Cancer Res*, 4217–20.
- Cs231n. (n.d.). *Convolutional Neural Networks (CNNs / ConvNets)*. Retrieved April 28, 2018, from CS231n Convolutional Neural Networks for Visual Recognition: <http://cs231n.github.io/convolutional-networks/>
- DADOUCHE, A. (2018, March 27). *Machine Learning in a Box (week 7) : Jupyter Notebook*. Retrieved April 28, 2018, from SAP Software Solutions | Business Applications and Technology: <https://blogs.sap.com/2018/03/27/machine-learning-in-a-box-week-7-jupyter-notebook/>
- Davis, I. D. (2000). An overview of cancer immunotherapy. *Heidelberg*, 179–195.
- Dhungel, N. (2016, October). AutomatedDetection,Segmentationand ClassificationofMassesfromMammograms usingDeepLearning. *AutomatedDetection,Segmentationand ClassificationofMassesfromMammograms usingDeepLearning*. adelaide.

- Di Gaetano N., C. E. (2003). Complement activation determines the therapeutic activity of rituximab in vivo. *J. Immuno*, 1581–1587.
- Douglas Hanahan, R. A. (2011, March 04). Hallmarks of Cancer: The Next Generation. *sciencedirect*, 144(5), 646-674.
- Elias Jabbour, S. O. (2014). Monoclonal antibodies in acute lymphoblastic leukemia. *bloodjournal*, 125(26), 4010-4016.
- Frank A, A. (2010). UCI Machine Learning Repository.
- Gavin P. Dunn, A. T. (2002). Cancer immunoediting: from immunosurveillance to tumor escape. *Nat Immunol*, 991–998.
- Gelderman, K. T. (2004). A. Complement function in mAbmediated cancer immunotherapy. *Trends Immunol*, 158–164.
- Geresu, M. A. (2016). A Review on Immunotherapy against Cancer. *researchgate*.
- GLOBOCAN. (2010, May 05). *GLOBOCAN 2008: Cancer Incidence and Mortality Worldwide*. Retrieved January 2018, from GLOBOCAN: <https://www.iarc.fr/en/media-centre/iarcnews/2010/globocan2008.php>
- Gupta, M. (2013, September 13). *Random Forests Algorithm*. Retrieved April 28, 2018, from Data Science Central: <https://www.datasciencecentral.com/profiles/blogs/random-forests-algorithm>
- Hausen, Z. (2009). Papillomaviruses in the causation of human cancers - a brief historical account. *Virology*, 206-5.
- Hebb, D. O. (1949). The Organization of Behavior: A Neuropsychological Theory. *W. Press, Ed*, 378.

- Institute, N. C. (2015, December 18). *Cancer Vaccines*. Retrieved 04 19, 2018, from National Cancer Institute: <https://www.cancer.gov/about-cancer/causes-prevention/vaccines-fact-sheet>
- Islam¹, M. J., Ahmadi², M., & Sid-Ahme, M. A. (2010). AN EFFICIENT AUTOMATIC MASS CLASSIFICATION METHOD IN DIGITIZED MAMMOGRAMS USING ARTIFICIAL NEURAL NETWORK. *International Journal of Artificial Intelligence & Applications*.
- John M. Kirkwood, A. A. (2008). Next generation of immunotherapy for melanoma. *Journal Of Oncology*, 3445-55.
- K, R., & P, F. (1998). Glossary of terms. *Machine Learning* . 271-274.
- KDnuggets. (2016, july). *Support Vector Machines: A Simple Explanation*. Retrieved April 25, 2018, from Machine Learning, Data Science, Data Mining, Big Data, Analytics: <https://www.kdnuggets.com/2016/07/support-vector-machines-simple-explanation.html>
- Kreitman R.J., P. I. (2011). Antibody-fusion proteins: anti-CD22 recombinant immunotoxin: moxetumomab pasudotox. *Clin Cancer Res* , 6398–405.
- Kustrin, & Beresford. (2000). Basic concepts of artificial neural network (ANN) modeling and its application in pharmaceutical research. *Journal of Pharmaceutical and Biomedical Analysis*, 717-727.
- Mellman I, C. G. (2011). Cancer immunotherapy comes of age. *Nature*, 480-9.
- Minda Asfaw Geresu, A. F. (2016). mmunotherapy against cancer: A comprehensive review. *cademicjournals*.
- Minsky, & Papert. (1969). Perceptrons. MIT Press, Cambridge. *MIT Press*.

- Mittendorf EA1, S. P. (2010). Mechanisms of T-cell inhibition: implications for cancer immunotherapy. *tandfonline*, 89–10.
- Nayab, & Scheid, J. (2011, September 2). *Advantages of Decision Tree Analysis*. Retrieved April 28, 2018, from Bright Hub Project Management: <https://www.brighthubpm.com/project-planning/106000-advantages-of-decision-tree-analysis/>
- Nickgillian. (2014, April 17). *KNN* . Retrieved April 28, 2018, from NickGillianWiki: <http://www.nickgillian.com/wiki/pmwiki.php/GRT/KNN>
- P.D., S. (2010). Activation of prodrugs by antibody-enzyme conjugates: a new approach to cancer therapy. *FASEB*, 188–93.
- Peniel M. Dimberu, R. M. (2011). Cancer Immunotherapy Takes a Multi-Faceted Approach to Kick the immune System into Gear. *Yale Journal of Biology and Medicine*, 371-380.
- Peniel M. Dimberu, R. M. (2011). Cancer Immunotherapy Takes a Multi-Faceted Approach to Kick the immune System into Gear. *Yale Journal of Biology and Medicine*, 371-380.
- Programiz. (n.d.). *Learn Python Programming The Definitive Guide* . Retrieved April 28, 2018, from Learn Programming: Tutorials and Examples from Programiz: <https://cdn.programiz.com/python-programming>
- R.F, M. R. (1996). Antibody-Enzyme Conjugates for Cancer Therapy. *J Natl Cancer Ins*, 153-65.
- R.V.J, T. B. (2011). Antibody Conjugate Therapeutics: Challenges and Potential. *Clin Cancer Res* , 6389-6397.
- Reinis M, B. (2008). a personalized therapeutic vaccine against B-cell lymphomas. *Curr Opin Mol*, 526–34.

- Researchoptimus. (n.d.). *What is Logistic Regression?* Retrieved April 28, 2018, from Researchoptimus: <https://www.researchoptimus.com/article/what-is-logistic-regression.php>
- Rosenberg S, D. M. (2009). Adoptive cell therapy for the treatment of patients with metastatic melanoma. *Current Opinion in Immunology*, 233–240.
- Rumelhart, D. E., & McClelland, J. L. (1986). *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*. New York. *MIT Press*.
- Sally J. Deeb, S. T.-S., & Mann, M. (2015). Machine Learning-based Classification of Diffuse Large B-cell Lymphoma Patients by Their Protein Expression Profiles. *mcponline*, 2947-2960.
- Sharma P., W. K. (2011). cancer immunotherapy agents with survival benefit: recent successes and next steps. *Nature reviews*, 805-12.
- Shinde, M. (2003). Computer Aided Diagnosis In Digital Mammography. *Computer Aided Diagnosis In Digital Mammography*: . South Florida, USA: University of South Florida .
- Siegel R, D. C. (2014, August 14). Cancer statistics. *cacancerjournal*, 64(5), 21.
- Steeg, P. S. (2006). Tumor metastasis: mechanistic insights and clinical challenges. *Nature Medicine* , 895 – 904.
- Steven A. Rosenberg, M. E. (2009). Adoptive cell therapy for the treatment of patients with metastatic melanoma. *Current Opinion in Immunology*, 233–240.
- Sun Y1, P. Y. (2003). Application of artificial neural networks in the design of controlled release drug delivery systems. *sciencedirect*, 1201-1215.

- UCBWiki. (2016, February 11). *Reinforcement Learning with Function approximation*. Retrieved April 22, 2018, from UCB Wiki: <http://wiki.ubc.ca/images/b/bd/Rldiagram1.png>
- UCI. (2018, January 04). *Immunotherapy Dataset Data Set*. Retrieved January 20, 2018, from UCI Machine Learning Repository: <http://archive.ics.uci.edu/ml/datasets/Immunotherapy+Dataset>
- University, C. (2015, March 16). *Immunotherapy: New Hope for Patients With Advanced Lung Cancer*. Retrieved April 15, 2018, from Bioscience Technology: https://abm-website-assets.s3.amazonaws.com/biosciencetechnology.com/s3fs-public/bt1503_columbia_immunotherapy.jpg
- W.H.O. (2018, February). *WHO Cancer*. Retrieved 04 14, 2018, from World Health Organization: <http://www.who.int/mediacentre/factsheets/fs297/en/>
- Zakka, K. (2016, July 13). *A Complete Guide to K-Nearest-Neighbors with Applications in Python and R*. Retrieved April 24, 2018, from Kevin Zakka's Blog: <https://kevinzakka.github.io/2016/07/13/k-nearest-neighbor/>

APPENDIX

SOURCE CODES

```
##### Immunotherapy Dataset Thesis
```

```
# Dataset will be processed by several different supervised machine learning techniques and results will be compared to determine the one that yields the highest accuracy via python. . ##### Dataset Preparation
```

```
# * Decision tree, Random forest, KNN, Naive Bayes, SVM, Logistic Regression. Convolutional
```

```
.....
```

Near East University

Information Systems Engineering

Ofor Emmmanuel 20164898

Machine Learning: Immunotherapy Classification

Master Thesis.

```
.....
```

```
##Plot Services
```

```
import matplotlib.pyplot as plt
```

```
#Dataset Importation and Preparation
```

```
#Dataset Importing Pandas Library
```

```
import pandas as pandas_dataframe
```

```
import numpy as np
```

```
from sklearn.metrics import precision_recall_fscore_support
```

```
from datetime import datetime
```

```
from sklearn.metrics import precision_recall_curve
```

```
import matplotlib.pyplot as plt
```

```
#Reading Immunotherapy Datase File
```

```
imported_data = pandas_dataframe.read_csv('immunotherapy.csv')
```

```
#Dataset Importation and Preparation
```

```
#Display Imported Data
```

```
imported_data.head(90)
```

```

#Display Full Details of Dataset and stastical calculations
imported_data.describe()

##Columns Names
imported_data.columns

##Result_of_Treatment Counts
## Where 1 indicates successful
treatment_counts = imported_data['Result_of_Treatment'].value_counts()

treatment_counts

##Age Plot barChart
treatment_counts.T.plot(kind='bar',legend=True,title='')
plt.savefig('treatment_counts',dpi=300)
plt.show()

##Age Counts
age_counts = imported_data['age'].value_counts()

age_counts

##Age Plot barChart
age_counts.plot(kind='bar')

##age_counts.plot(kind='bar', legend=True)
plt.savefig('age_counts',dpi=300)

# fig.savefig("myplot.png")
# fig.savefig('foo.png', bbox_inches='tight')
# fig.savefig('output.png')

plt.show()

##Sex Counts
sex_counts = imported_data['sex'].value_counts()

sex_counts

#processing dataset to check for missing data
#step ignored cause there was no missing data in the dataset

```

```

#imported_data.dropna(inplace=True)

##Finally Describe Filtered Values

imported_data.describe()

#####Converting Pandas dataframes into numpy arrays

all_features = imported_data[['sex', 'age', 'Time', 'Number_of_Warts', 'Type', 'Area', 'induration_diameter']].values

#####Array COncersion

all_classes = imported_data['Result_of_Treatment'].values ##Target Column

#####Global usage columns names in Arrays

feature_names = ['sex', 'age', 'Time', 'Number_of_Warts', 'Type', 'Area', 'induration_diameter']

## Indicates Features of Each Column in Arrays

all_features

#Some of our models require the input data to be normalized,

##Hint: using preprocessing.StandardScaler().

## SVM and Neural Network Techniques Greatly Depends on Data Normalization.

from sklearn import preprocessing

scaler = preprocessing.StandardScaler()

all_features_scaled = scaler.fit_transform(all_features)

all_features_scaled

#Decision Trees

# import numpy

# from sklearn.model_selection import train_test_split

# numpy.random.seed(0)

# (training_inputs,

# testing_inputs,

# training_classes,

# testing_classes) = train_test_split(all_features, all_classes, train_size=0.75, random_state=1)

# ## Global

# Creating Globally train/test split of our data. Set aside 75% for training, and 25% for testing.

```



```

#Decision Trees

#mapping classes accross scaled features 75% = 0.75

#single Train/Tet Split Data.

import numpy

# from sklearn import preprocessing

from sklearn.model_selection import train_test_split

# numpy.random.seed(0)#If the random seed is not reset, different numbers appear with every invocation
(duplication/Debug and testing reasons)

(training_inputs,testing_inputs,training_classes,testing_classes) = train_test_split(all_features_scaled,all_classes,
random_state=1)

#testing_classes) = train_test_split(all_features_scaled, all_classes, random_state=1)

# training_inputs  === all_features_scaled

# testing_inputs  === all_classes

# training_classes === train_size 0.75 ~ 75%

# testing_classes  === random_state

#testing_classes) = train_test_split(all_features_scaled, all_classes, train_size=0.75, random_state=1)

number_list = list(range(1, 20))

run      = " Run"

number_range = [1,2,3,4,5,6,7,8,9,10]

print(number_list[:10])

# print (number_range)

# ## Decision Trees

#Decision Tree Result and test data.

# decision_tree_final_score = classifier.score(testing_inputs, testing_classes)

import statistics as s

from sklearn import metrics

#Display the resulting decision tree.

from IPython.display import Image

from sklearn.externals.six import StringIO

```

```

from sklearn import tree

from pydotplus import graph_from_dot_data

import pydotplus

from sklearn.metrics import classification_report, confusion_matrix

from sklearn.metrics import precision_recall_fscore_support as score

#Now creating a DecisionTreeClassifier that fits into training data.

from sklearn.tree import DecisionTreeClassifier

start_time = datetime.now()

classifier = DecisionTreeClassifier()

# classifier= DecisionTreeClassifier(random_state=1)

# Train the classifier on the training set

classifier.fit(training_inputs, training_classes)

y_predict = classifier.predict(testing_inputs)

# print(confusion_matrix(testing_classes, y_predict))

# print(classification_report(testing_classes, y_predict))

dot_data = StringIO()

tree.export_graphviz(classifier, out_file=dot_data, feature_names=feature_names)

graph = graph_from_dot_data(dot_data.getvalue())

Image(graph.create_png())

##writes DTree to Image file

graph = pydotplus.graph_from_dot_data(dot_data.getvalue())

# graph.write_pdf("DecisionTree_Result.png")

graph.write_png("DecisionTree_Result.png")

# tree.export_graphviz(classifier, out_file='tree.dot')

# tree.export_graphviz(classifier, out_file=dot_data)

# graph=pydotplus.graph_from_dot_data(dot_data.getvalue())

# graph.write_png("dtree.png")

# graph.write_pdf("dtree.pdf")

```

```

decision_tree_final_score = cross_val_score(classifier, all_features_scaled, all_classes, cv=10)
decision_tree_final_score_plot = decision_tree_final_score

decision_tree_final_score = s.mean(decision_tree_final_score)

# precision, recall, fscore, support = score(testing_classes, y_predict )
support = score(testing_classes, y_predict )

recall = cross_val_score(classifier, training_inputs, training_classes, cv=10, scoring='recall')
precision = cross_val_score(classifier, training_inputs, training_classes, cv=10, scoring='precision')
fscore = cross_val_score(classifier, training_inputs, training_classes, cv=10, scoring='f1')
accuracy = cross_val_score(classifier, training_inputs, training_classes, cv=10, scoring='accuracy')

print('-----')
print("Decision Tree Score:",decision_tree_final_score)
print("Percentage:", "{0:.2f}%".format(decision_tree_final_score*100))
print('-----')
print('precision : ',precision.mean()," ")
print('recall   : ',recall.mean()," ")
print('fscore    : ',fscore.mean()," ")
# print('support : ',support.mean()," ")
# print('A.Accuracy: ', "{0:.2f}%".format(accuracy.mean()*100))
print('-----')
print('Mean Absolute Error:', metrics.mean_absolute_error(testing_classes, y_predict))
print('Mean Squared Error:', metrics.mean_squared_error(testing_classes, y_predict))
print('Root Mean Squared Error:', np.sqrt(metrics.mean_squared_error(testing_classes, y_predict)))
# Accuracy_Score = accuracy_score(testing_classes, y_predict)
# # print ('ACC:',Accuracy_Score)
# print('Single Accuracy:', "{0:.2f}%".format(Accuracy_Score*100))
#####plot#####
print('-----')
x = np.arange(10)

```

```

plt.title("Decision Tree Scores: Mean={0:.3f}" .format(decision_tree_final_score))

plt.bar(x, decision_tree_final_score_plot,color='darkorange')

plt.xticks(x, number_list[:10])

plt.savefig('Decision Tree Mullti',dpi=300)

plt.show()

#####plot#####

print('-----')

# do your work here

end_time = datetime.now()

print('Duration: {}'.format(end_time - start_time))

print('-----')

#####plot#####

print('-----')

# do your work here

end_time = datetime.now()

print('Duration: {}'.format(end_time - start_time))

print('-----')

my_colors = 'rgbkymc' #red, green, blue, black, etc.

plt.title("SVM Kernel: Best is Rbf={0:.3f}" .format(svm_rbf_final_score))

plt.savefig('all_svm_scores',dpi=300)

plt.show()

plt.plot([svm_linear_final_score,svm_rbf_final_score,svm_sigmoid_final_score,logistic_regression_final_score])

# plt.ylabel('some numbers')

# plt.show()

# plt.savefig('age_counts')

all_svm_scores_sorted = sorted(all_svm_scores)

all_svm_scores_sorted

### All Techniques Values

```

```

all_techniques_scores = [decision_tree_final_score,
                           k_fold_cross_validation_final_score,
                           random_forest_final_score,
                           knn_final_score,
                           svm_rbf_final_score,
                           logistic_regression_final_score,
                           convolutional_ann_final_score
                           ]

```

```

all_techniques_names = [ "D.Tree",
                           "K-Fold",
                           "R.F",
                           "KNN",
                           "SVM",
                           "L.R"
                           "CNN"
                           ]

```

```

# all_techniques_names = ["Decision Tree",
#                           "K-Fold Cross",
#                           "Random Forest",
#                           "KNN",
#                           "Naive Bayes",
#                           "SVM",
#                           "Logistic Regression",
#                           "Convolutional"
#                           ]

```

```

s = pd.Series(
    all_techniques_scores,
    index = all_techniques_names
)

```

```

)

my_colors = 'rbgkymc' #red, green, blue, black, etc.

xx = np.arange(7)

# plt.bar(xx, all_techniques_scores)

plt.title("All Techniques: Best is KNN = {0:.0f}%".format(knn_final_score*100))

plt.xticks(xx, all_techniques_names)

s.plot(
    kind='bar',
    color=my_colors,
)

plt.savefig('techniques_scores',dpi=300)

plt.show()

#sort

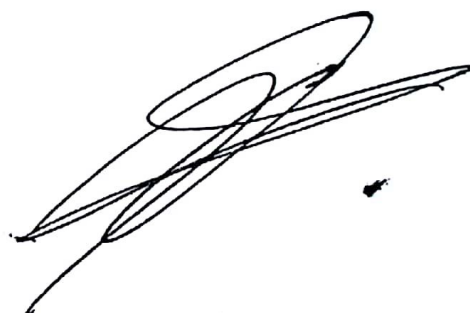
all_techniques_scores_sorted = sorted(all_techniques_scores)

all_techniques_scores_sorted

```

SIMILARITY REPORT AFTER JURY FOR
EMMANUEL OFOR - 20164898

Chapters	Percentages
Abstract.doc/docx	0%
Chapter 1.doc/docx	0%
Chapter 2.doc/docx	13%
Chapter 3.doc/docx	8%
Chapter 4.doc/docx Results.doc /docx	8%
Chapter 5.doc/docx Conclusion.doc/docx	0%
*All.doc/docx	14%



Assoc.. Prof. Dr. Kamil Dimililer

MSc. Supervisor