**A PROSTATE CANCER DETECTION MODEL USING SUPPORT VECTOR MACHINE APPROACH**

**CHUKWUDI CHUKWUNWEMMERI CHUDI-IWUEZE**

**CLU180203-181**

**AUGUST, 2021**

A PROSTATE CANCER DETECTION MODEL USING SUPPORT VECTOR MACHINE APPROACH

CHUKWUDI CHUKWUNWENMERI CHUDI – IWUEZE

CLU180203 – 181

A PROJECT IN THE DEPARTMENT OF COMPUTER SCIENCE SUBMITTED TO THE COLLEGE OF NATURAL AND APPLIED SCIENCE IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF BACHELOR OF SCIENCE (B.Sc.) OF CHRISLAND UNIVERSITY, ABEOKUTA

AUGUST, 2021

**DECLARATION**

I, CHUKWUDI CHUKWUNWENMERI CHUDI – IWUEZE, do hereby declare that this project work is entirely my work and composition. The work embodied in this project has not been submitted in candidature for any degree and is not concurrently being submitted for any degree. All references made to works of other persons have been duly acknowledged.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Chukwudi Chudi-Iwueze .C.

Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**CERTIFICATION**

This is to certify that this research work was carried out by Chukwudi Chudi-Iwueze in the department of Computer Science, College of Natural and Applied Sciences, Chrisland University, Abeokuta. The research work is considered adequate in partial fulfillment of the requirement for the award of B.Sc. in Computer Science.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

Prof. O. Folorunso Date

(Major Supervisor)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

Dr. Z. Oni Omogbadegun Date

(Head of Department)

**DEDICATION**

This paper is dedicated to the Almighty God, without whose grace and love this paper would not have been written.

**ACKNOWLEDGEMENT**

My endless gratitude goes to the Almighty God, for His grace and kindness that preserved me and saw me through, making this endeavor productive. I would not be productive without this grace.

To Lady. Bridget Nwankwo, Pharm. Chukwudi Iwueze, Mrs. Ihuoma Iwueze, for their support, love, care, and prayers I say a big thank you. It is your prayer and strength that have brought me this far.

To Onyinye Iwueze, you have been a friend, mother, and sister. I owe you everything. Your belief in and support of me is beyond what I deserve and I am eternally grateful.

I would like to appreciate the Vice Chancellor Prof. Chinedum Babalola, and the entire management of Chrisland University, for their effort and continued support. Also, for the hard work of instilling excellence and character in her students.

To all the lecturers and staff of the Department of Computer Science, Chrisland University, particularly, the Head of Department Dr. Zaccheus OniOmogbadegun, your commitment to excellence and consistent support for us students can only be repaid by God Almighty.

I would like to express immense gratitude to my Supervisor Prof. Folorunso for his invaluable and constructive contributions during the planning and development of this project. Your patience and guidance through the writing of this project have been invaluable.

To my friends and everyone who has encouraged me through the tedious task of writing this project, Priscilla, Bio, Bright, Barry, Emenike, Demilade, Olamide, Tomiwa, Akah, Whalid, Ruth, Brother Victor, Blessing, Emy, Martha, Precious, I say a big thank you. Your encouragement helped make the difference.

**ABSTRACT**

The American Cancer Society estimates that 1 in every 9 men 60 and above will be diagnosed with this disease. A major challenge with the diagnosis of this disease is that it can go a long time without detection and when detected, tests are difficult and usually time consuming. This project tends to proffer a solution to this challenge of testing and diagnosis. This project outlines the development of a decision support system for people who are at the risk of being diagnosed with prostate cancer. Approaching the problem as a classification problem, the major aim of the project is to use the Support Vector Machine model in the design and development of an Application Programming Interface to assist the diagnosis of the disease. I performed a comparative analysis of the Support Vector Machine to determine the effectiveness of the model in the analysis of the test cases. This is to effectively predict prostate cancer in test samples. I compared the model with the Naïve Bayes model and the Logistic regression. It was found to have an approximate accuracy of 90% above the 83% and the 80% of the other two models, respectively. The model was used in the creation of an Application Programming Interface and was found to be an effective solution given its high accuracy and its effective management of large datasets. In conclusion, the use of the Support Vector Machine was found to be an effective model in the classification, analysis and prediction of prostate cancer samples for effective analysis and diagnosis. However, recommendations suggest that the model be tested with wider range data to improve system optimization. This would further ensure accurate diagnosis of the disease and establish optimal accuracy levels for the decision support system.

*Key words: prostate cancer,diagnosis, tests, Machine learning techniques, Support Vector Machine, Logistic Regression, Naive Bayes.*

*Word count: 301.*

TABLE OF CONTENT

Content Page

Title Page ………………………………………………………………………… I

Declaration …………………………………………………………………….... II

Certification …………………………………………………………………..… III

Dedication ………………………………………………………………………. IV

Acknowledgement …………………………………………………………...….. V

Abstract …………………………………………………………………………. VI

List of tables …………………………………………………………………….. X

List of figures …………………………………………………………………… XI

1. Chapter One: Introduction …………………………………………………... 1
   1. Background to the study ……………………………………………….. 1
   2. Problem Statement ……………………………………………………… 2
   3. Objective of the study …………………………………………………... 2
   4. Significance of the study ………………………………………………... 3
   5. Scope of the study ……………………………………………………......4
   6. Definition of terms ……………………………………………………….4
   7. Organization of the work …………………………………………………5
2. Chapter Two: Literature Review …………………………………………....... 6
   1. Introduction …………………………………………………………........ 6
   2. Scholarly definition of terms ………………………………………...…... 6
   3. Machine learning in health care …………………………………………. 7
      1. Introduction ………………………………………………………. 7
      2. Application of Machine learning in health care ………………….. 8
      3. Advantages…………………………………………………….… 12
      4. Research gaps …………………………………………………… 13
3. Chapter Three: Design Methodology ………………………………………... 14
   1. Introduction …………………………………………………………….. 14
   2. Dataset ……………………………………………………………………14
   3. Support Vector Machine Algorithm ……………………………………... 14
   4. System Design ……………………………………………………….…... 16
      1. Structure Chart ………………………………………………….... 16
      2. Machine learning …………………………………………………. 17
         1. Flowchart for model learning …………………..…….. 18
      3. Application programming Interface ……………………………... 20
         1. Flowchart for Application Programming Interface …. 20

1. Chapter Four: Implementation, Result and Discussion ………………………. 23
   1. Introduction …………………………………………………………...… 23
   2. System Specification……………………………………………………... 23
      1. Hardware Requirements …………………………………………. 23
      2. Software Requirements ………………………………………….. 23
   3. Data retrieval and Feature Extraction ………………………………...…. 24
      1. Data retrieval ………………………………………………….… 24
      2. Feature extraction …………………………………………….…. 24
   4. Training the model …………………………………………………..…. 25
   5. Evaluation of model ………………………………………………..…… 25
   6. Comparison of Support Vector Machine with other Models …………... 26
   7. Application Programming Interface ………………………………….... 28
   8. Discussion ………………………………………………………….…... 30
      1. Comparison of Support Vector Machine with other models ……. 30
      2. Application of Programming result ……………………………… 33
2. Chapter Five: Summary and Conclusion
   1. Introduction ……………………………………………………………… 35
   2. Summary ………………………………………………………………… 35
   3. Conclusion ………………………………………………………………. 35
   4. Future works …………………………………………………………….. 35
   5. Recommendation ……………………………………………………….. 36

References ……………………………………………………………………….. 37

Appendices ………………………………………………………………………. 40

LIST OF TABLES

Table Pages

Table 4.1. Box scores for Naïve Bayes Model …………………………… 30

Table 4.2. Box scores for Logistic Regression …………………………… 31

Table 4.3. Box scores for Support Vector Machine ……………………… 32

Table 5.1. Summary of F1-score of each model …………………………35

**List of figures**

Figure Pages

Figure 3.1. Simple representation of hyperplanes ……………………… 12

Figure 3.2. Structure chart for diagnosis system ……………………….. 14

Figure 3.3. System architecture for diagnosis system ………………….. 16

Figure 3.4. Flowchart for Model building ……………………………… 18

Figure 3.5. Flowchart for API building ………………………………… 19

Figure 3.6. Cross section of dataset …………………………………….. 19

Figure 4.1. Code snippet for column reduction ………………………… 21

Figure 4.2. Code snippet changing text data to numeric ……………….. 21

Figure 4.3. Code snippet for function …………………………………... 22

Figure 4.4. Code snippet calling the train and fit function ……………... 22

Figure 4.5. Confusion Matrix for SVM ………………………………… 23

Figure 4.6. Confusion Matrix for Logistic Regression …………………. 24

Figure 4.7. Confusion Matrix for Naïve Bayes …………………………. 25

Figure 4.8. Code snippet loading the model into the API ………………. 26

Figure 4.9. Code snippet creating the .mdl file …………………………. 26

Figure 4.10. Code snippet creating the .pkl file ………………………… 26

Figure 4.11. Code snippet showing the function that call the model …… 27

Figure 4.12. The format for the test data saved in JSON format.……….. 28

Figure 4.14. Screenshot of the API being tested ………………………... 33

**CHAPTER ONE**

**Introduction**

**1.1. Background to the study**

Cancer is widely considered to be one of the world’s most dangerous diseases. It is recorded that cancer is responsible for about 13% of all deaths in the world (Oyewo and Boyinbode, 2020). According to The World Health Organization (W.H.O.), in 2020 cancer accounts for nearly 10 million deaths. Information from *Medicinet.com* explained that cancer occurs when there are more abnormal cells than the immune system can handle. Abnormal cells continually multiply and then pile up into tumors – or cysts. The aggressive growth invades surrounding tissues. Cancer cells are named after the organs they attack, and there are about 200 types of cancers.

Prostate cancer is the most common cancer in American men. The American Cancer Society projects that in 2021, about 248,530 men would be diagnosed with the disease, and about 34,130 men would die from the disease, and that 1 man in every 8 men 65 years of age and above would be diagnosed with the disease, with an estimated death of 1 in every 41 case. Prostate cancer is the most rampant cancer death among men in the world. (Oyewo and Boyinbode, 2020). The risk factor of prostate cancer is much higher in men who are 65 or older. The estimated average age of men diagnosed with prostate cancer is at 66 years.(The American Cancer Society, 2021) In the United States of America, the diagnosed cases of prostate cancer rank first among all malignant tumors in men. (Bhattacharjee et al, 2019)

Prostate cancer is marked by an aggressive (malignant) growth of cells in the prostate gland. The prostate is the walnut-sized gland in men, found just below the bladder and in front of the rectum, surrounding the urethra. A major function of the prostate is to produce and store the fluid that helps to make semen.

Even though prostate cancer is so commonly suffered by men, it often begins without symptoms. The growth of prostate cancer can be gradual, such that a lot of men die of other ailments before prostate cancer causes any substantial damage. However, many prostate cancers are more belligerent and often spread beyond the boundary of the prostate gland, and they become malignant – critical. The prostate cancer survival rate is greatly improved with early detection and personalized treatment. (UCLA Health, 2020).

Detection, diagnosis, and prognosis of prostate cancer is a meticulous process that involves a series of tests. These tests start orally, where the patient speaks with his doctor, who tries to determine the risk factors, like the family history and identify possible physical symptoms.

Then a Digital Rectal Examination is performed based on the recommendation of the doctor. After which a PSA (Prostate-Specific Antigen) Blood Test is performed. The essence of this test is to determine the level of PSA in the blood. The PSA blood level is a strong indicator of the presence of this disease. The higher the PSA in the blood, the more likely the chances of prostate cancer.

Following this test is a Biopsy examination. In this procedure core samples of the prostate are removed and examined under a microscope. The results of the biopsy determine whether or not the patient has this disease. However, the biopsy might be inconclusive if the result might simply be suspicious – some abnormalities may or may not be prostate cancer. Usually, the remedy for this situation is the recommendation of more tests. (American Cancer Society, 2021).

The entire process produces specific data which can be utilized in the creation of models and results that expedite examinations and analysis. For instance images from the biopsy can be utilized in a neural network model to create a diagnosis system that can be trained progressively. Also, numeric data can be utilized by other models for the same purpose of expediting result analysis.

This thesis attempts the latter instance, to utilize numeric data in the creation of a machine learning model, which analyzes data samples to detect the presence or absence of malignant prostate cancer tumors.

**1.2. Problem Statement**

With the advent of machine learning and the application of data science models in various fields of learning and domains of knowledge, one particular field where attempts have been made to improve the outcome of research and often in diagnosis is in Medicine. Artificial Intelligence and machine learning are capable of stimulating change in health systems, improving efficiency and effectiveness, creating headroom for universal health coverage and positively affecting outcomes. (Panch, Szolovits, Atun. 2018)

Before this study, a lot of research has gone into applying machine learning techniques to diagnosing prostate cancer. This study adds to the wealth of knowledge and employs the use of a machine learning technique - support vector machines - to diagnose prostate cancer. The technique used in this study, following literature studies, is found to produced high levels of accuracy and is precise in the classification of data to determine the presence of defined diseases.

**1.3. Aim and Objective of the study**

According to a paper released by *UCLAhealth.org*, there is no defined cause for prostate cancer, however, studies and research have earmarked certain risk factors that increase the chances of a person getting cancer. These risk factors can be used to determine the likelihood of a person developing this disease.

This can be defined early; close observation with proper precautions can be taken to ensure that the person gets the help required to manage the situation and find a cure. And also to provide proper information for this disease, to help better educate potential patients on this condition.

This study aims to

1. Develop a system that could be used as a first opinion for people with a high risk of being diagnosed with this disease. The researcher understands that if the disease is spotted earlier there is a higher chance of providing a solution for it and changing the lifestyle of the individual to ensure they lead better lives.
2. Review existing literature on the prediction of Prostate Cancer using computationally intelligent techniques.
3. Design a prostate cancer detection model using Support Vector Machines.
4. Implement and evaluate the model built to detect Prostate cancer.
5. Apply the developed model in an Application Programming Interface.

**1.4. Significance of the study**

This project will be very beneficial to individuals who are at risk of this disease. It provides a diagnosis system built on tested and proven techniques that are accurate and precise in data segregation and classification of cases.

It should also be beneficial to the practitioner, as it aims to accelerate the testing of samples and diagnosis of the disease.

The use of machine learning techniques as done in this paper contributes to the available work being done to improve the field of medicine by utilizing data science and artificial intelligence techniques.

**1.5 Scope of the Study**

This project also seeks to take this approach a step further by using the Machine learning model as part of an Application Programming Interface (A.P.I.) for a web-based decision support system.

The inclusion of the API will allow the use of the model developed in the background in a front-end facing way. This can potentially allow users to input details and get an initial opinion and aid better lifestyle choices and decision-making.

**1.6. Definitions of terms**

1. Artificial intelligence:

Artificial intelligence (A.I.) is a branch of computer science that designs and develops intelligent systems capable of performing operations like human beings.

1. Machine Learning:

Machine learning is a branch of artificial intelligence that focuses on applying data-intensive algorithms in creating intelligent systems capable of making decisions based on inferences from data.

1. Diagnosis:

Diagnosis is the identification of diseases or problems by analyzing the symptoms related to that problem or disease.

1. Prostate Cancer:

Cancer of the prostate – the organ responsible for producing seminal fluids.

1. Support Vector Machines:

A machine learning algorithm that depends on a defined hyperplane. It attempts to fit linear boundaries between data samples to solve the problem of multidimensionality in data.

**1.7. Organization of Work**

The rest of the work is organized as follows: Chapter two is a review of related works, which explores the application of machine learning techniques in medical diagnosis. The use of support vector machines and other machine learning techniques in the diagnosis of other diseases along with prostate cancer.

Chapter three introduces the design of the project and explains the algorithm applied in this work. It explains the design of the decision support system.

Chapter four follows chapter three, it describes the dataset, the implementation of the algorithm applied in the chapter, the datasets, and the result and analysis of the result.

Chapter five, concludes the work while stating the recommendation for future works and the general contribution to knowledge.

Also, a list of references is included.

**CHAPTER 2**

**Literature Review**

**2.0. Overview**

This chapter is a review of previous work relating to the study being carried out. It analyses the influence of machine learning in health care and disease diagnosis, it reviews the success and the weakness of the proposed system.

**2.1. Scholarly definition of terms**

Artificial intelligence (AI) is a broad discipline that aims to understand and design systems that display intelligent properties - demonstrated in its ability to learn: derive knowledge from data. (Panch et al, 2018). Artificial intelligence is also the study of how to make computers do things at which, at the moment, people are better. (Rich and Knight, 1991).

Machine learning is a sub-discipline of Artificial intelligence (Panch et al, 2018). “Machine learning is a branch of artificial learning and computer science that focuses on the use of data and algorithms to imitate the ways humans learn, gradually improving accuracy”.( IBM Cloud Education, 2020). Machine learning refers to a large family of mathematical and statistical methods that are applied in predictions and model simulation. (Gadde and Kalli, 2020)

Deep learning: this is the basis of machine learning. It allows a system to be fed with a large amount of raw data and to discover the representations necessary for detection or classification. (Panch et al, 2018). This subset of machine learning describes techniques and methods usually in inference to neural networks used in big data processing - raw datasets, large numbers of unrelated and/or unstructured datasets, or high-speed low latency of data creation. (Badawi et al, 2014)

Diagnosis: Diagnosis is defined as the investigation or analysis of the cause or nature of a condition, situation, or problem. It is also defined as the act or art of identifying a disease from its signs and symptoms. (Merriam Webster, 2021).

Support Vector Machines (SVM) is a learning technique designed to fit a linear boundary between samples of a binary problem ensuring the maximum robustness in terms of tolerance to isotropic - referring in some way to multidimensionality; exhibiting properties when measured along axes in all directions. (Igual, Segui. 2017)

Application Programming Interface (A.P.I) is a software intermediary that allows two applications to communicate with each other. (MuleSoft, 2021)

These concepts defined above are umbrella terms that describe the techniques applied in this paper. This chapter analyses similar applications of the same concepts to either solve the same problem or similar problems.

**2.2. Machine Learning in Healthcare**

The provision of healthcare itself involves 2 core information processing tasks; first, the diagnosis - the classification of cases based on history, examination, and investigations. (Panch et al, 2018). This phase of health care involves the careful observation of cases and very often symptoms, which serve as the basis for postulated hypotheses, applied in the next phase. The second task is the treatment and monitoring, which involve the planning, implementation, and monitoring of a multistep process to deliver a future outcome. (Panch et al, 2018). This stage involves the test of theories from the diagnosis phase, initial prognosis, and postulation of possible remedies for the case.

These cases involve the postulation of and testing of hypotheses. It also involves drawing inferences from available data. Machine learning expands on existing statistical techniques that are not dependent on any a priori assumptions about the distribution of data and can find patterns that can, in turn, be used to formulate hypotheses and test the hypothesis. Machine learning has the potential to mitigate hypothesis generation and hypothesis testing and tasks within a health system by revealing hidden trends in data.

Historically, doctors took a history, performed or recommended physical examinations, and made a diagnosis based on what he or she observed. With progress in technology and medical theory, domains such as image processing and laboratory modalities aid in the mitigation of chances in the diagnosis of diseases. (Badawi et al, 2014)

* + 1. **Applications of Machine learning in healthcare (Model-wise review)**

**Diagnostic Image Analysis using Neural Networks.**

Image analysis is the extraction of meaningful and concise features from digital images through computationally intelligent techniques / analytical and image processing techniques.

Mukherjee, 2017, in his article, *A.I. versus M.D: what happens when diagnosis is automated,* highlights the research and experiment carried out in January 2015 by Sebastian Thrun, where he challenges the rule-based algorithm applied in first-generation diagnostic devices and applies a more learning-based one. Thrun applies the “Neural Network” technique in his research because of its semblance to the function of a human brain, and its ability to learn progressively, similar to how a human works. Thrun begins his experiment with skin cancer (Keratinocyte carcinoma and melanoma). The experiment attempts to train the system to distinguish skin cancer from benign skin conditions - acne, rash, mole - using photographs. His initial test consisting of a validation set ( made up of 1400 images that had been diagnosed by dermatologists), showed an accuracy of about seventy-two percent compared to the sixty-six percent accuracy by two board-certified dermatologists who tested alongside the system.

The above experiment highlights the application of image processing (deep learning technique) in medical diagnosis. It also shows the increased rate of success between the system and human analysis. Although it does not exclude the human varieties that follow diagnosis, observations show that machine learning applied in disease detection improves accuracy compared to human diagnosis.

However, we must also recognize that while improvements are made to this field, these methods augment human effort in medical diagnosis. (Mukherjee and Thrun, 2017).

**Disease prediction:**

**Ensemble of Machine Learning Models**

The idea of an ensemble model is built on selecting several or a group of hypotheses from a given hypothesis space and combining their predictions, for improved optimization. (Russell and Norvig, 2010).

Ensemble Method is a machine learning technique that combines the predictive prowess of multiple base models to produce one single optimal predictive model. (Lutins, 2017).

Oyewo and Boyinbode in their paper “*Prediction of Prostate cancer using an ensemble of Machine Learning Techniques”* applied an ensemble of machine learning techniques - they combined the Multilayer perceptron, the decision tree, and the Support Vector Machine (SVM) to make predictions based on a defined dataset for the presence of malignant prostate cancer or benign prostate cancer tumors.

The dataset used in this instance consisted of about 641 data instances, with about 209 actual prostate cancer data and 432 non-prostate cancer. Their model correctly identified 205 instances of prostate cancer and 430 instances of non-prostate instances. They achieved a total accuracy of about 99 percent with the ensemble model. However, each model was found to produce different accuracy levels. The Multilayer perceptron produced an accuracy level of about 97.6599 %, with the Support Vector Model falling far behind with an accuracy of about 67.39% and the Decision tree at an accuracy of 92.9797% produced the next best individual result.

Observations from this paper show that the multilayer perceptron, although much more complex than the other two models, was much more valuable in solving this problem than the other two models.

**Support Vector Machines.**

In 2019, Bhattacharjee et al attempted a machine learning classification method to classify Gleason grade groups of prostate cancer.

The Gleason grade system is used to evaluate the severity of cancer cells. Higher grades denote a more bellicose cancer that propagates rapidly. (Bhattacharjee et al, 2019) This system was developed by Dr. Donald F Gleason, a Pathologist in Minnesota, and a member of the Veterans Administration Cooperative Urological Research Group (VACURG). This system has been proven on a large number of patients, with accompanying long-term follow-ups, and is considered an outstanding success and an adequate standard for prostate cancer diagnosis. (Gleason, 1992)

The Gleason grade system is an evaluation system for grading prostate cancer. The grading system combines two numbers on a scale from 2 for nonaggressive cancer to 10 for very aggressive cancer. (Singh, 2020).

The Gleason score is obtained by adding the primary (most common) and secondary (second most common) scores from H&E stained tissue microscopic images. A major challenge is the evaluation of the scores. Each score is given a certain grade based on the total of the primary and secondary scores. Scores less than or equal to 6 are in grade 1 and are not as lethal, a score combination of 3 for the primary and 4 for the secondary to make 7 would be a grade 2 score, while a score combination of 4 for the primary and 3 for the secondary score would be a grade 3 score. Grade 3 scores though benign would have to be under very close observation. Score combinations that are equal to 8 are rated as grade 4, higher scores are in grade 5. Grade 4 and 5 scores are malignant grades and imply that the patients require critical medical attention. (Bhattacharjee et al, 2019).

The research by Bhattacharjee et al, 2019, used the Support Vector Machine model as a result of improved accuracy compared to other models. The model used a linear and Gaussian kernel to classify samples as benign and malignant and discriminate between grade 3 and grade 4+5, and grade 4 and grade 5 of the Gleason grade groups.

They used a dataset of 400 images. 240 of the images were used for training and 160 were used for testing. 100 images were considered for each grade group, and the images were classified as malignant versus benign.

The binary classification methods - Support Vector Machine - produced an accuracy level of 92.5%, 90.0%, 90.0%, and 95.0%, for the grade 3, grade 4 and grade 5 respectively compared to 60%, 55%, 85%, and 50% for the same value set using a multi-class or one-shot classifiers.

In 2019, Yu et al in their paper "Application of Support Vector Machine modeling for prediction of common disease: the Case of diabetes and pre-diabetes" demonstrated the effectiveness of the machine learning model – Support vector machine – in the detection of common diseases – in this case, diabetes. They applied the model to a dataset retrieved from the National Health and Nutrition Examination Survey (NHANES). They selected 14 variables associated with the risk of diabetes – family history, age, gender, race and ethnicity, weight, height, waist circumference, BMI, hypertension, physical activity, smoking, alcohol use, education, and household income.

Following the selection of the optimal variables, LibSVM – a freely available SVM software library – was used to generate the SVM model.

There were two classification schemes; classification scheme I and scheme II. Classification scheme I compared diagnosed or undiagnosed diabetes with no diabetes or pre-diabetes. 8 variables – family history, age, ethnicity, weight, height, waist circumference – were used in these schemes to produce the best performance. Classification scheme II compared undiagnosed diabetes or pre-diabetes with no diabetes. 10 variables (family history, age, ethnicity, weight, height, waist circumference, BMI, hypertension, sex, and physical activity) were used in the scheme to obtain the optimal performance for the model.

The RBF (Radial Basis function) kernel was found to produce the best result in scheme I, while the linear kernel performed best in scheme II.

The researchers found that the overall discriminative ability of the classification scheme represented by the AUC values were at 83.47% and 73.18%. Comparison between this model and the traditional Multiple Logistic Regression used in solving this problem, showed no significant statistical difference in discriminative ability, implying that applying the Support Vector Machine Model to the diagnosis of diabetes would produce results that are similar in output to traditional models used in solving the same problem.

* 1. **ADVANTAGES OF MACHINE LEARNING IN HEALTH CARE**

A publication in 2011 by Chuang et al, where they applied the Support Vector Machine model to Single Nucleotide Polymorphism to predict for Oral cancer to aid DNA repair as a solution to oral cancer, postulates that the application of analytical models makes it practical to systematically explore genome-wide interactions. Such solutions to critical problems like these could change the way we practice medicine as they improve the effectiveness of current practice.

Machine learning potentially can improve clinical practice. This is evident in the identification of drug side effects. Models such as Naïve Bayes, linear and logistic regression, and similar models can be applied using combined and comprehensive variables to summarize robust variations without increasing the complication. (Badawi et al, 2014).

Integration of computers in medicine provides a quintessential combination, which improves the capacity of human beings to meet expectations and monitor AI systems to apply its abilities in the evaluation of vast quantities of data, further improving the predictive power. (Gadde and Kalli, 2020).

Furthermore, deep learning can be applied to solving fundamental biological problems, due to its ability to adequately leverage large amounts of data derived from studies. Neural networks have proven useful in the prediction of gene targets of microRNAs and the prediction of protein residue. (Ching et al, 2018).

In summary, computational intelligence techniques demonstrate superior learning abilities and are effective in handling complex and often ill-defined problems.

**2.4 RESEARCH GAP**

After a careful review of the literature, the researcher found that a lot of the models designed to test and predict the presence of certain diseases are not made available for public use. They are designed as experimental approaches to disease prediction. It is this niche that this research intends to fill.

The model tested in this project is applied in the creation of an API, which can be used by future software developers in the implementation of a frontend design for the prediction of this disease.

**CHAPTER 3**

**Design**

**3.0. Overview**

This section explains the methodology, including the phases used to develop the decision support system for prostate cancer. The research methodology utilizes the Support Vector Machine (SVM) in building the Machine Learning Model. SVM is a supervised learning technique that can be applied to both classification and regression problems. (Bhattacharjee et al, 2019). The system proposed in this research seeks to take previous research a step further by using the developed machine learning as part of an Application Programming Interface (API).

**3.1. Support Vector Machine Algorithm**

SVM is a binary classifier – it generally refers to classifiers that have 2 class labels (a normal state and an abnormal state). (Brownlee, 2021). It works by abstracting a decision boundary in multi-dimensional spaces using an appropriate subset of the training set of vectors. The elements of the subset are the support vector. (Sharma, 2016). SVM generates an optimal hyperplane in an iterative manner that maximizes the margin, where the margin is the smallest distance between the decision boundary and any of the samples. (Bishop, 2006) The closest point of the classes to the boundary is referred to as the Support Vector. Any change in the support vector means that the margin (the boundary) would change. (Igual, Segui. 2017).

The Support vector machine algorithm is built on the concept of linear separability. A dataset is linearly separable if there we can find a plane to separate the sets of data. (Add the name, 2020)

This concept feeds into the concepts of hyperplanes.









Figure 3.1. Simple representation of hyperplane margin, showing the distance from the data line. The leftmost line separates the positive values from the margin, with the closest data point being on the line, and the right-most line separates the opposite data classification. This is a generic diagrammatic description of a perfectly linear hyperplane.

Source: Bishop 2006.

The margin is defined as the perpendicular distance between the decision boundary and the closest of the data points. The location of this boundary is determined by a subset of the data points, known as support vectors. (Bishop, 2006).

Hyperplanes are used to separate data of more than 2 dimensions.

Definition 1: A hyperplane in Rd (a subspace) is defined as a connection combination of the variables.

(Igual, Segui. 2017)

Definition 2: The distance of a point x Rd to the hyperplane is

*D (x*,) =

(Igual, Segui. 2017)

According to Russell and Norvig, in their textbook; *Artificial Intelligence: A Modern Approach*, it is understood that for SVMs, class labels are traditionally either +1 or -1. SVMs do not have weights as other models do. The intercepts are kept as separate parameters.

Definition 3: The separator is defined as the set of points

{X: W.X+B=0}

We could search the space W and X with gradient boosting – an algorithm to iteratively determine what values appropriately suit the problem, it usually starts as some small value and rises iteratively to determine what values best suit the margin placement – to find the parameters that maximize the margin while correctly classifying all the examples. (Russell and Norvig, 2010).

**3.2 Logistic Regression**

Logistic regression is a type of model for probabilistic statistical classification. It is utilized as a binary model in the prediction of binary responses, the outcome of a categorical dependent variable, based on some variables. (Igual and Segui, 2017) The logistic regression model seeks to fit the weights of a linear model in order to minimize the loss on a data set (Russell and Norvig, 2010).

Definition 4: the form of a logistic regression is givens as

Where w is the weight and x is the bias.

The logistic regression function can take in input values ranging from negative infinity to positive infinity and produce an output restricted to values between zero and one.

**3.3. Naïve Bayes Model**

The Naïve Bayes Model is a set of classification models based on the Bayes theorem. (Gandhi, 2018). It is a learning algorithm that employs the Bayes theorem while assuming that the attributes are provisionally independent, given the class.

Definition 5: the Bayes Theorem is shown as this

This theorem is a deterministic algorithm as the probability of an event is determined by the occurrence of a prior event. A and B are events, B is the evidence and A is a hypothesis.

The model utilizes a conditional independence between all independent variable features, to ensure that the effects a feature has on the outcome would in no way interact with the way another feature affects the outcome. (Rice, 2013)

A major setback of this model is that it works best with relatively small datasets showing high levels of effectiveness with small datasets and falling short in larger datasets.

**3.4. System design**

In the design of this system, a structure chart was used to describe at a very high level the flow of data from the model to the backend of the system. A flow chart is used to add detail to each component described by the structure chart.

**3.4.1. System Architecture.**

Structure charts are modular structures that represent procedures by rectangles and procedure dependencies by directed arrows. (Brookshear, 2013, p 323)

Figure 3.2 is a structure chart representing the different modules of these and their respective dependencies









**Figure 3.2.** The figure above highlights a general module, “the diagnosis system” which is the umbrella term for the entire system. The next module highlighted is the Machine Learning model that is built using the Support Vector Machine and finally, the Application Programming Interface designed using the python framework Flask.

The system architecture outlines the flow of data through the system. It highlights the organization of the system and explicitly shows how the system manages data to produce the result expected. It starts at the retrieval of data and outlines the flow through the system to the API and subsequently the evaluation of the system using input data at the endpoint of the developed API, while highlighting the models used in the comparative analysis of the Support Vector Machine with other models.

**3.4.2 Machine Learning Model**

This subsection of the chapter describes at a very high level the processes involved in training a machine learning model.

There are generally five processes involved in training and evaluating a machine learning model.

1. Data collection and preparation

Our system uses secondary data from Kaggle.com. The data used in the system is already preprocessed and is ready for use. However, basic preprocessing is to be carried out on the data to reduce the number of columns in the data set and also change text data to numeric data for seamless training and evaluation.

1. Feature selection and feature engineering.

This step involves all the changes performed on the data set to improve it. This includes all column reductions and changes of data type.

1. Choosing the machine learning algorithm and training the first instance of the model:

This system employs the support vector machine model, due to its usefulness in solving classification problems.

1. Evaluating the model:

After the initial training and of the model, test data is used to evaluate the system for accuracy. The best accuracy margins are selection and there is no necessity for the system to be 100% accurate.

1. Model tweaking, regularization, and hyper parameter tuning:

Where the evaluation has failed to reach some defined margin, certain parameters can be tweaked and tuned to improve the chances of achieving the desired output. These Parameters include the cost function, the interception values, or the weights used by the model in training.

3.4.2.1.Flowchart for model training

Flowcharts are a diagrammatic representation of the stepwise nature of an algorithm.

Flowcharts are used to depict processes, systems, or computer algorithms. Various shapes are used to represent the steps. Such steps include rectangles, ovals, diamonds, and several other shapes used in depicting data flow and data sequence. (Lucid charts. 2021)

This flow show depicts the processes in line with the Machine Learning Model methodology explained above. It begins with the data retrieval and ends with the evaluation and testing of the system.

















































**Figure 3.4 Flowchart for the model building and comparative analysis.**

**3.4.2.2.** Application Programming Interface

Application Programming Interfaces (APIs) are interfaces/intermediaries that streamline software development and innovation by simplifying and securing the exchange of data and functionality between applications. (IBM cloud education, 2020).

This API outlines how data flows from the prediction model through the API to produce output that can be used in a frontend system.









**Figure 3.5. Flowchart for the Application Programming Interface.**

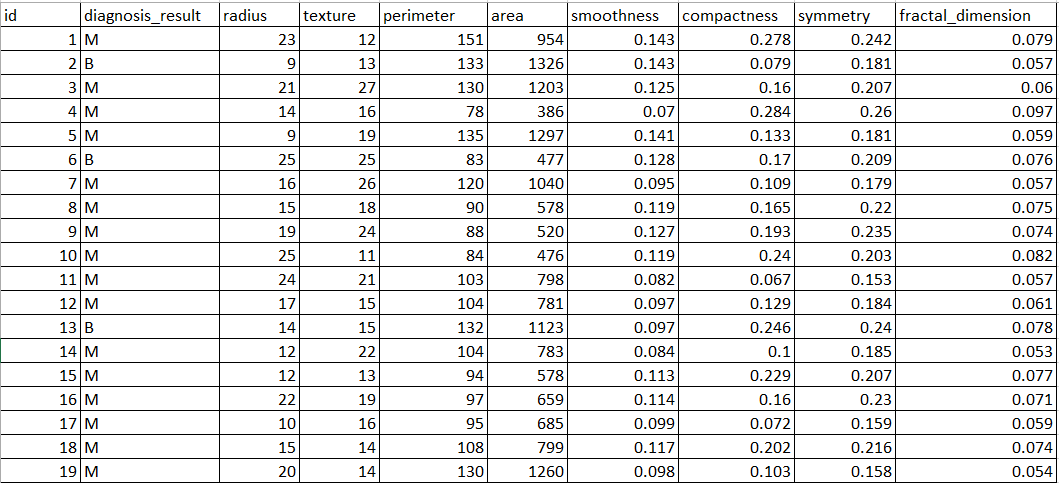
**3.5. Dataset**

The data set was retrieved from kaggle.com from <https://www.kaggle.com/stefanonormanno/support-vector-machines/comments>. The dataset was prepared and preprocessed by Stefano Normano and retrieved in June 2021.

The dataset is a 10 column dataset with 100 data instances. The column headers represent key information for the training and testing; highlighting the Diagnosis results, radius, texture, perimeter, area, smoothness, compactness, symmetry and fractal dimension.

The dataset used in this study was already preprocessed and therefore, no processing or dimension reduction was performed on the dataset, however, the diagnosis result which is in text format was converted to numeric to ensure that processing and analysis are carried out as expected.

A cross section of the dataset is shown below highlighting the columns and several instances of the data being examined.



**Figure 3.6. Cross section of dataset.**

**3.5. System Specification**

For the completion of this project certain hardware and software requirements must be met. While some requirements are expendable, these requirements must stay constant as a change in system requirements might mean that the desired results may not be achieved, for instance, omission or absence of a module will return an error.

The following subsections outline the system (hardware and software) requirements

**3.5.1. Hardware requirements**

The hardware requirements for the proposed systems should have the following specifications (or at the least very similar specifications).

1. PC workstation with Intel quad-core processor of 2.50 GHz
2. 8GB of RAM
3. 256GB hard disk capacity
4. A mouse
5. A keyboard
6. A monitor

**3.5.2. Software requirements**

The software requirements for the proposed systems should have the following specification (or at the very least close specifications).

1. Windows 7, 8, 10 operating system
2. Spreadsheet package
3. Python programming language
4. Scikit learn
5. Pycharm IDE
6. Jupyter notebook
7. Postman

**CHAPTER FOUR**

**Implementation, Results, and Discussions**

**4.0 OVERVIEW**

This chapter intends to highlight the implementation of the model defined in the previous chapter. The process of building the Support Vector Model is outlined, the results of the building process’s training and testing phase are also highlighted.

**4.1. FEATURE EXTRACTION.**

This section identifies all the feature selection tasks carried and the feature extraction activities performed. To prepare the data, for use in this project, the number of columns was reduced to make the data set more concise.

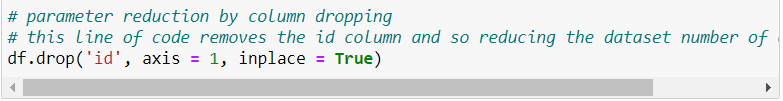


Figure 4.1 the code snippet above carried the function of reducing the data set by one column.

Then text data was changed to numeric, to enable the system to perform training operations easily.

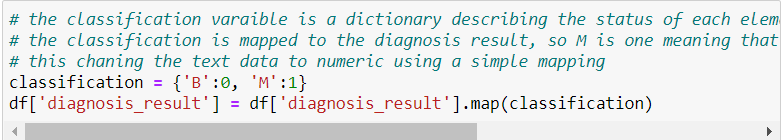


Figure 4.2. The code snippet above carries the function of changing the text data to numeric. Data into a machine learning model has to be numeric, this presents the essence of this code snippet.

**4.2. TRAINING THE MODEL**

This is the section where the model is trained.

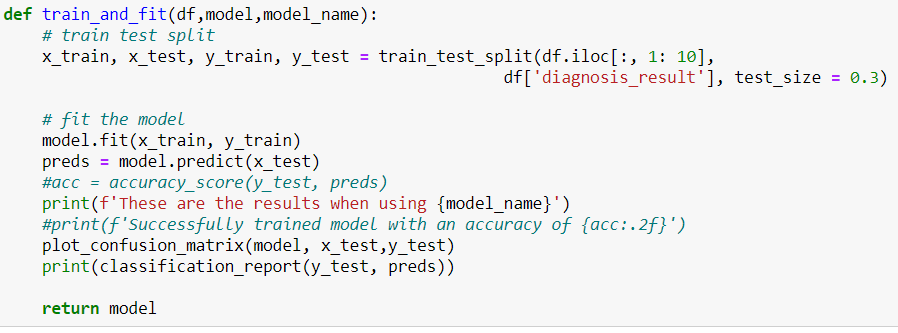


Figure 4.3. Code snippet showing function to firstly, split data set into training and test data. Secondly, train the model and print the report for the accuracy level of the model.

The evaluation of this model uses a confusion matrix to show the accuracy level of the employed model.

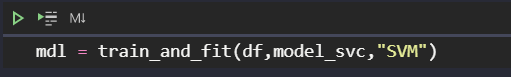


Figure 4.4. This code instance calls the train\_and\_fit function highlighted above in training the SVM using the Scikit learn SVC module – SVC is an acronym that stands for Support Vector Classifier.

**4.3. EVALUATION OF MODEL**

Following the training of the model, it was evaluated for optimum accuracy. The implementation of this model involves the plotting of a confusion matrix to express accuracy.

The confusion matrix is a graph used to diagrammatically describe the performance of a model. It is a technique that summarizes the performance of a classification algorithm. The diagonal elements of a confusion matrix consisting of the true positive and true negative elements represent the number of points in which the predicted label is equal to the true label. True positive values are the outcomes where the model correctly predicts the positive class. The true negative values are the outcome where the model correctly predicts the negative classes. These values are represented by the diagonal elements on the confusion matrix.

The off-diagonal elements representing the false positive and false negative are those that are mislabeled by the classifier. The false positive values are the outcome where the model incorrectly predicts the positive class. The false negative is an outcome where the model incorrectly predicts the negative class.

If the diagonal values of the confusion matrix are higher, it indicates that more of the predictions are accurate.

| True Positive values | False Positive Values |
| --- | --- |
| False Negative Values | True negative values |

Pictorial depiction of a confusion matrix.

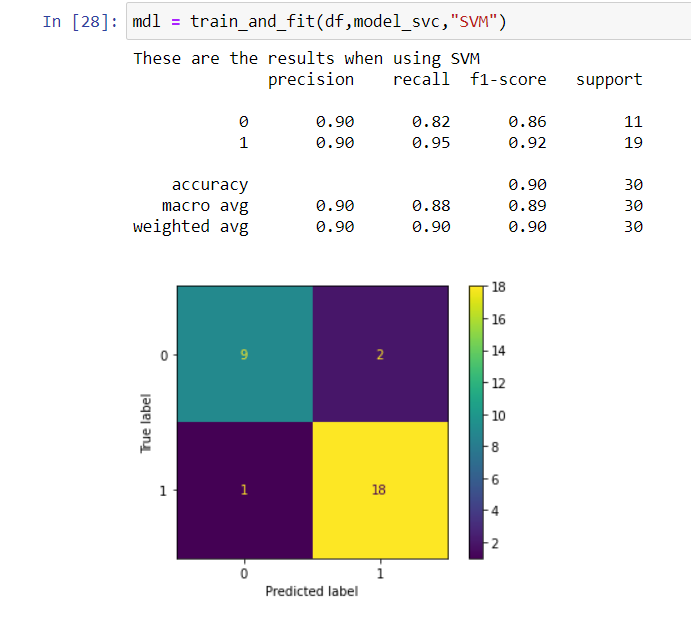
In the evaluation of the model as well, keen interest was paid to the Accuracy level of the model, the weighted average and the macro average. These values constitute the average scores of the recall, F1score and the precision, in giving a total of how well the model performs in classifying the data set.

**The Accuracy** is given as number of correct predictions by the model divided by the total observations

**The Macro average** is the average of the scores per individual class. The average of the F1score, the precision and the recall. The macro average is the mean value of the scores of all the classes. For instance, the macro average for recall would be given as

**The weighted average** is given as the scores of all the classes after multiplying their class proportions. For instance the weighted average for recall would be given as

This formula applies to all classes.



**Figure 4.5. Confusion Matrix for the SVM.**

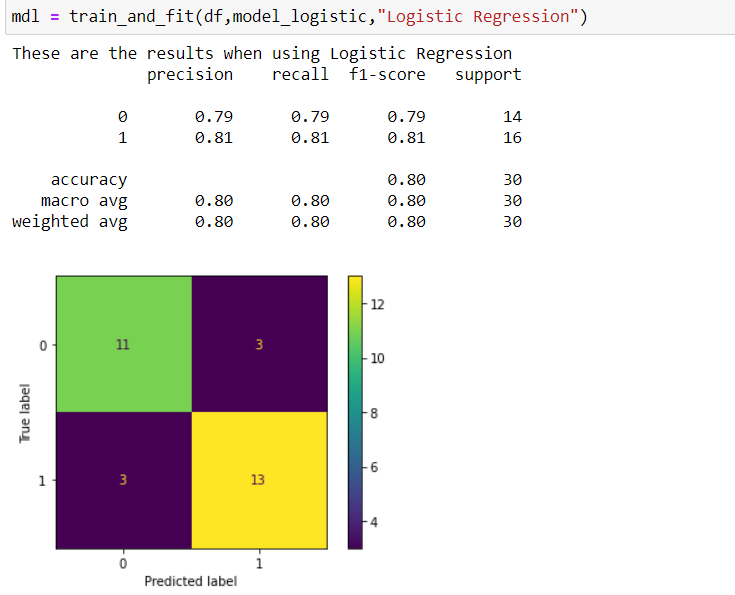
We see from the image that the confusion matrix has diagonal values of 9 and 18, which means that the model has a high prediction accuracy. The true positive values and the true negative values are 9 and 18 respectively. These values imply that there are more correctly classified values than there are incorrectly classified values of the 30 test subjects used.

The print output shows an accuracy of 90%. The Macro average of the model is at 89%, and the weighted average is 90 %.

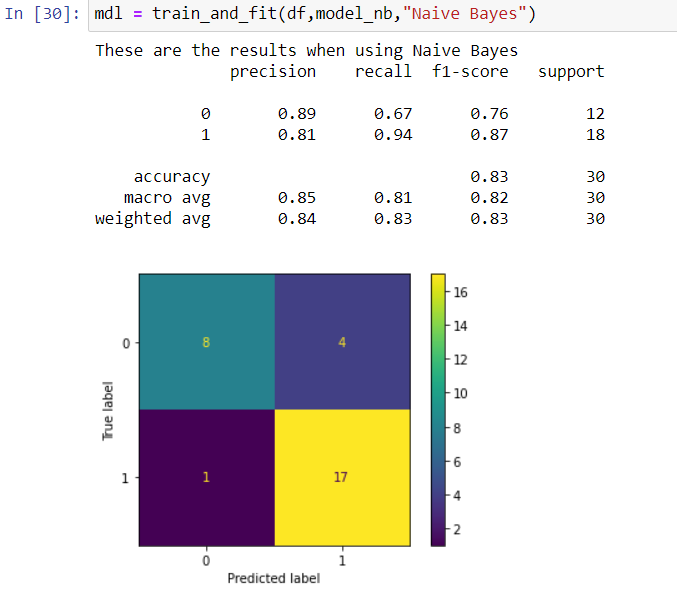
**4.4. COMPARISON OF SVM WITH OTHER MODELS.**

To better understand the accuracy of the model, a few models were trained alongside the same dataset and the accuracy was evaluated along with the accuracy of the SVM.

The following figures show the accuracy of the other models tested against the SVM.



**Figure 4.6.** The first alternative model is the Logistic Regression model. We see that the diagonal scores of the confusion matrix are less than the SVM, at 11 and 13. Also, it has an accuracy score of 80% which is far less than the 90% of the SVM's accuracy score.



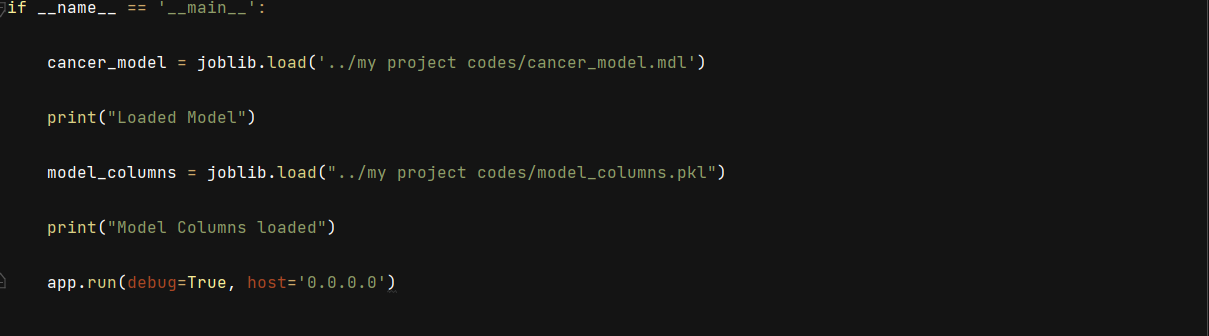
**Figure 4.7.** The second model which the accuracy of the SVM was tested against was the Naïve Bayes model. These showed a closer accuracy level to the SVM, however, the SVM at 90% is more accurate for this problem, as the Naïve Bayes model had an accuracy level of 83%. The confusion matrix for the model has a combined score of 8 and 17.

In summary, for the problem being solved the SVM model is more accurate in solving this problem than using Logistic regression or Naïve Bayes model. Conclusively, it can be inferred that the Support Vector Machine is the best model for this problem, with greater accuracy than the Logistic Regression and the Naïve Bayes.

**4.5. APPLICATION PROGRAMMING INTERFACE**

The API for this project is designed in Flask – a mini python framework, famous for its concise nature and easy-to-understand syntax.

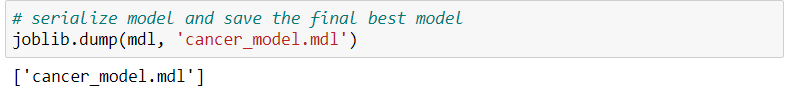
The API loads the model in .mdl format. This .mdl file is then used by the API to make the necessary predictions on the input data.



**Figure 4.8**. This code snippet shows the loading of the model into the API.

Line 1: is the entry point into the code. It tests that the name and the main module are in the same location.

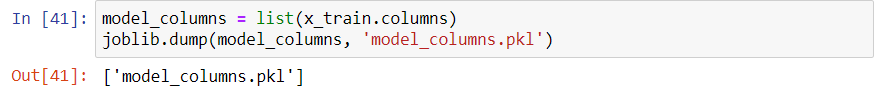
Line 2: loads the cancer model into the API. The cancer model is saved as a .mdl file. This file is created in the model training file



**Figure 4.9** is the code snippet that creates a .mdl file that is used by the API to make its predictions.

Line 3: prints output to show that the model is loaded successfully.

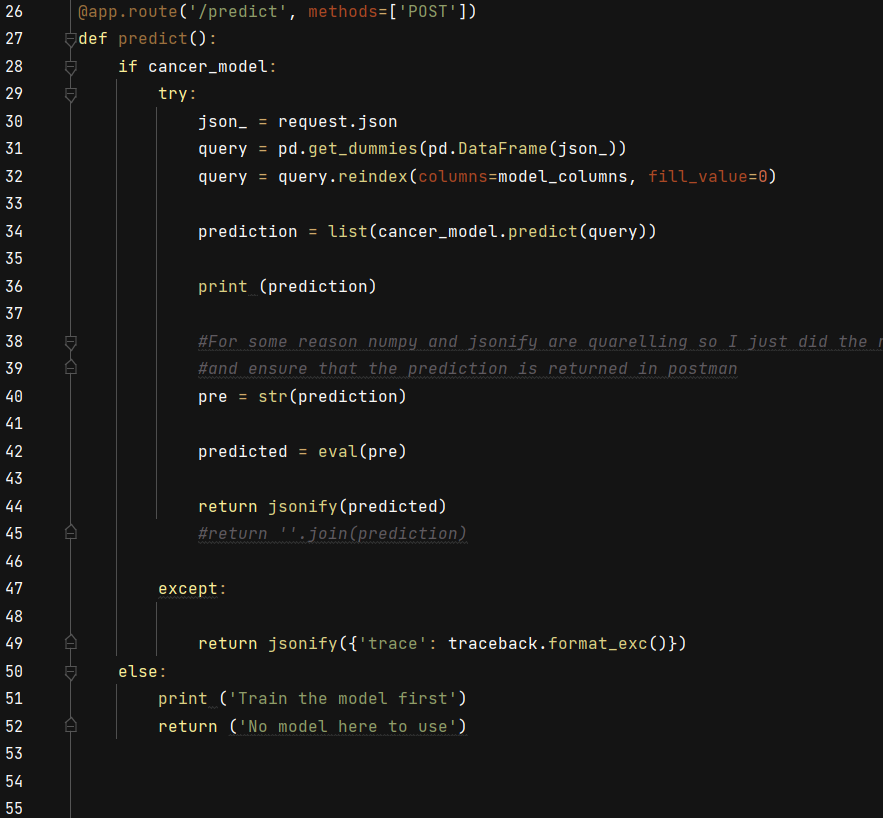
Line 4: loads the column head of the dataset to test the accuracy of the data input during testing.



**Figure 4.10**. Shows the creation of the .pkl file

Following the creation of the required .mdl and .pkl files, the API function collects the data from the endpoint and sends it to the model for the prediction.

The figure below is the API function written with python’s flask framework



**Figure 4.11** code snippet showing the function that calls the model.

Line 26 of the image is the call to the function. The "methods = [POST]" notifies the endpoint that the function is a post-call to the API. It also ensures that the program begins running at the /predict function.

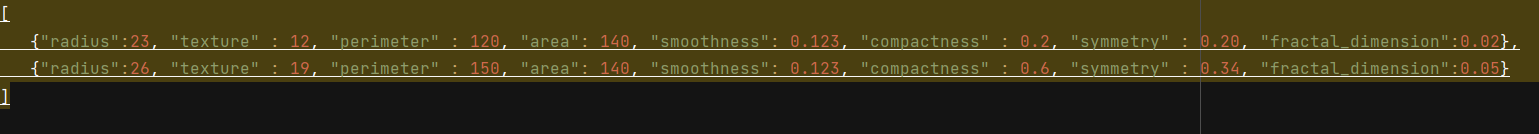
Line 27 is the function declaration.

Line 28 to Line 49 is an IF block that tests the model being inputted for correctness and availability.

Where this test is not passed, the block of code between line 50 and line 53 prints an error message in the terminal.

Finally, the data format for the test uses a Dictionary structure with the keys of the data as the column headers to ensure that the data matches the data for each column and remove all compliance issues.

The data is also saved as a JSON file and it is in this format that data output is received.



**Figure 4.12** the format for test data saved in JSON format.

**4.6. DISCUSSION**

**4.6.1 COMPARISON OF SUPPORT VECTOR MACHINE ACCURACY WITH OTHER MODELS**

At a base level, the problem being solved is a classification problem, being that the model aims to create categories based on the input data and classify the input into one of the data categories.

The parameters required to test the accuracy of a model include

* + 1. Precision: this is the number of true positives divided by the number of true positives and false positives. This describes the number of times the model is correct when it predicts a class.

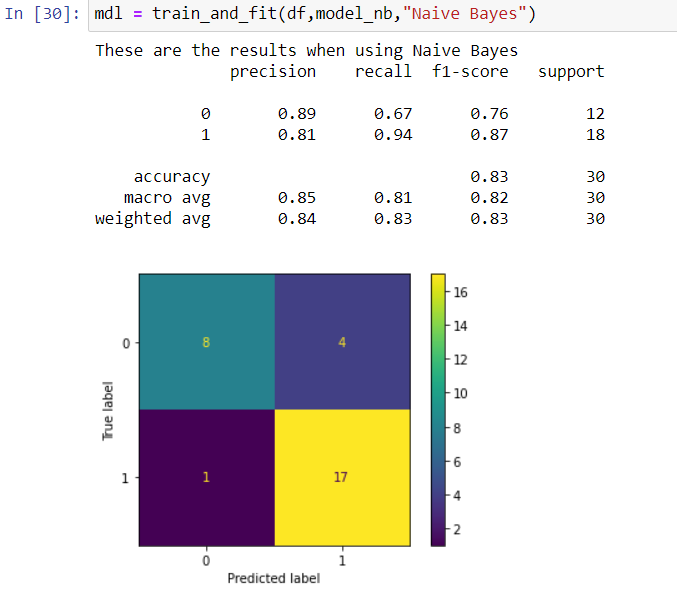
Precision =

* + 1. Recall: Recall also known as true positive rate or sensitivity is defined as the number of true positives divided by the number of true positives and false negatives. The true positive value is the outcome where the model correctly predicts the positive class. The false negative is the outcome where the model incorrectly predicts the negative.

Recall =

* + 1. F1 – score: this is

Other models can be used for the same purpose, and for this purpose the researcher compared the accuracy level achieved by other models, to make the best choice of model to use for the API. The researcher tested with the Logistic Regression and Naïve Bayes model, the result of the model is outlined below, compared with the result of the Support Vector Machine.



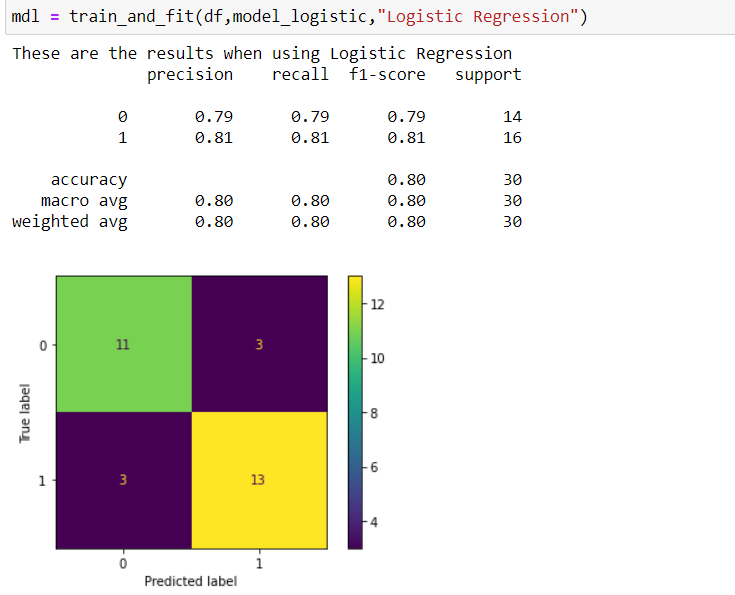
**Figure 4.13 Accuracy test for Naïve Bayes.**

It was observed that the Naïve Bayes model when applied to this problem produced an accuracy of 83%. While this is not a poor score, the 17% error level might be a problem in the API system, as this implies a 17% error margin. If this can be reduced by another model, that model automatically becomes a better model to apply.

| Class | Precision | Recall | F1-score | Support |
| --- | --- | --- | --- | --- |
| 0(benign) | 0.89 | 0.67 | 0.76 | 12 |
| 1(Malignant) | 0.81 | 0.94 | 0.87 | 18 |

**Table 4.1. Box scores for the Naïve Bayes model, describing performance about key parameters.**

30% of the input was used to train and fit the model. From that 30% the precision of properly identified benign tumors is at 89% with a recall of 67% and an F1-score of 76%.

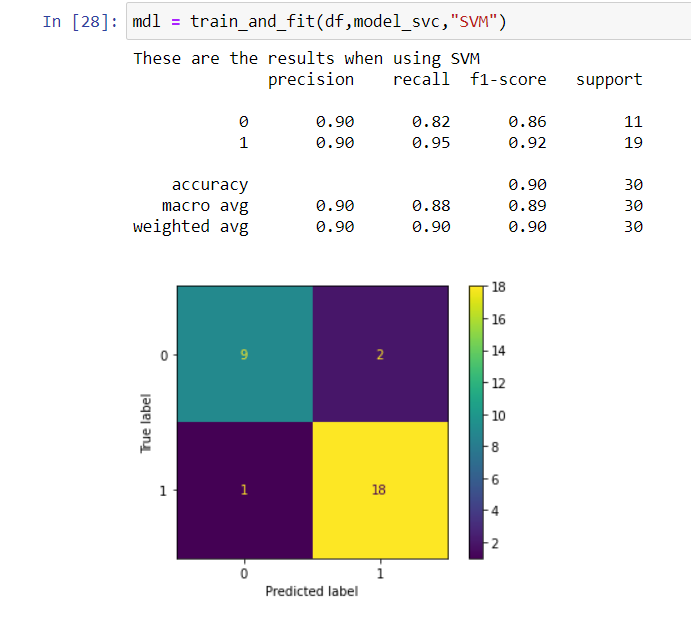


**Figure 4.14 Accuracy test for Logistic regression**

This model compared to the Naïve Bayes model produced an even less accuracy score of 80%, leaving an error margin estimated at 20%.

| Class | precision | Recall | F1-score | Support |
| --- | --- | --- | --- | --- |
| 0(benign) | 0.79 | 0.79 | 0.79 | 14 |
| 1(malignant) | 0.81 | 0.81 | 0.81 | 16 |

**Table 4.2. Box scores for Logistic regression model, describing performance about key parameters.**



**Figure 4.15 Accuracy for Support Vector Machine**

This model produced the best score of the three models tested at an accuracy of 90%. The error margin is reduced to 10% implying a much lower error margin than the other two models tested.

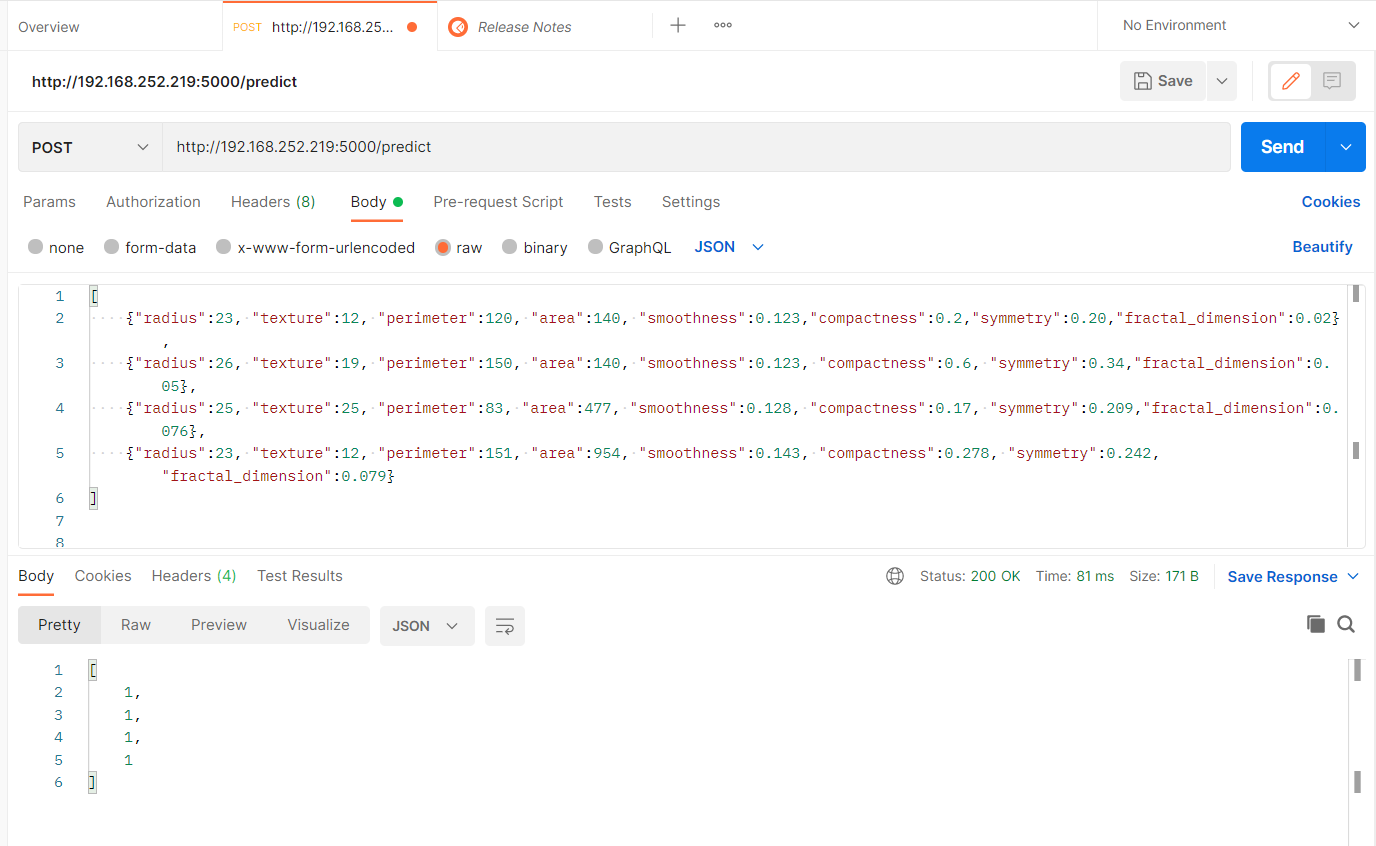
| Class | precision | recall | F1-score | Support |
| --- | --- | --- | --- | --- |
| 0 (benign) | 0.90 | 0.82 | 0.86 | 11 |
| 1 (malignant) | 0.90 | 0.95 | 0.92 | 19 |

**Table 4.3. Box scores for Support Vector Machine, describing performance about key parameters.**

It is safe to conclude from the box scores of each model that the Support Vector Machine presents the best scores in each category of classifier validation.

**4.6.2. API RESULT.**

Following the test and the validation of the model and the selection of the model best suited for the problem being solved, an API was built to collect user data and make the prediction for each data element being tested for the presence of prostate cancer or the absence of the same.



**Figure 4.16. Screenshot of the API being tested.**

Firstly, the local host link is being loaded in postman, and a call to the /predict function in the code base is made through this link.

Secondly, the raw data is sent through the link to the predict function, where it is then tested. In a case where the input does not match the format, an error is returned by the function.

Finally, the output of the test is returned in the specified format. This format can be changed to match the usage requirement of the final user.

**CHAPTER 5**

**Conclusions and Recommendations.**

**5.0. Overview**

This chapter of the project is the summary of the entire project. It reviews the challenges encountered. States recommendation for future works and in the final section, a brief review of the methodology and the output gotten from implementation is performed.

**5.1. Summary**

This project outlines the development of API, to predict the presence of prostate cancer in given data input.

The research introduces the paper by analyzing the risk level of the disease and the current diagnosis method and the flaws of that process. It states the scope of the study and the objectives of the paper, paying keen attention to the value it could add to medical diagnosis.

Following this is the literature that analyzes similar applications of machine learning and Support Vector Machines in medical diagnosis. It states the advantages that machine learning adds to medicine while analyzing the research gaps present in this domain of knowledge.

Chapters three and four are the chapters that explain the methodology applied in attempting this problem, the implementation of the methodology, and the results that were gotten at the end of the trial.

**5.2. Conclusion**

The project proposed the use of the Support Vector Machine in the design and implementation of an Application Programming Interface.

The implementation of this model was compared with other models to define which classification model performs the best with the data set. Used

The average F1 – score – the balance between precision and recall – for each model is tabulated to show the total score. Critical attention was paid to the accuracy of each model in defining which class each data element should be in.

|  | Accuracy | Macro Average | Weighted Average |
| --- | --- | --- | --- |
| SVM | 0.90 | 0.89 | 0.90 |
| Logistic regression | 0.80 | 0.80 | 0.80 |
| Naïve Bayes | 0.83 | 0.82 | 0.83 |

**TABLE 5.1. Summary of F1-score of each model.**

The table shows how each model performed at the training phase of the model.

Following the training and testing phases of the model, the model with the perfect score was used in the implementation of the API. Testing the API with the proper data format showed that the model was successful in predicting the presence or absence of data.

**5.3. Future Works**

Future researchers can take this paper a step further by implementing a front-end system using the API. A lot of the groundwork for the backend has been done. Therefore, the implementation of a frontend system makes the research more relevant in commercial domains.

The same method, particularly, the ML model can be tested on other disease datasets to verify the validity of the method in diagnosing other diseases and the prediction of the presence of other health conditions.

**5.4. Recommendation**

Emphasis should be placed on data collection and collation. Applying a lot of solutions to the Nigerian populace would be much easier with the right data and properly and accurately defined statistical values.

I recommend that the college pay more attention to courses in numerical analysis and statistics. This should equip her students with the necessary skills to venture more into mathematical computation. These skills and knowledge are also very crucial to gaining a career in data analysis and machine learning.

Artificial Intelligence should be taught with a more practical approach. Students should be encouraged to build and test their models. This should afford them more experience in machine learning.

**References.**

Badawi, O., Brennan, T., Celi, L. A., Feng, M., Ghassemi, M., Ippolito, A., MIT Critical Data Conference 2014 Organizing Committee. (2014). making big data useful for health care: a summary of the inaugural mit critical data conference. *JMIR medical informatics*, *2*(2), e22. Brookshear J.G, (2013). “*Computer Science: an overview* (11th ed. p 323).” Pearson Education Limited. Edinburgh Gate, England.

Bishop. C.M. (2006). *Pattern Recognition and Machine Learning.* Springer Science + Business Media LLC. Singapore. Springer.com

Bhattacharjee. S, Park H, Kim. C, Prakash D, Madusanka. N, So. J, Cho. N, Choi. H. (2019). *Quantitative Analysis of Benign and Malignant Tumours in Histopathology: Predicting Prostate Cancer Grading Using SVM.* Applied Sciences. [Applied Sciences | An Open Access Journal from MDPI](https://www.mdpi.com/journal/applsci). Appl. Sci. 2019, 9, 2969; DOI: 10.3390/app9152969

Brownlee. J. (2020, August 19). 4 Types of Classification Tasks in Machine Learning. <https://machinelearningmastery.com/types-of-classification-in-machine-learning/>. Retrieved 19 July 2021.

Brownlee. J. (2020, August 19). Classification Accuracy is not Enough: More Performance Measures you can use. <https://machinelearningmastery.com/classification-accuracy-is-not-enough-more-performance-measures-you-can-use/>. Retrieved 19 July 2021.

Ching T et al. (2018). *Opportunities and obstacles for deep learning in biology and medicine.* J. R. Soc. Interface 15: 20170387. <http://dx.doi.org/10.1098/rsif.2017.0387>

Gadde S.S., Kalli. V. D. (2020) *Descriptive Analysis of Machine Learning and its Application in Healthcare.* International Journal of Computer Science Trends and Technology (IJCST) – Volume 8 Issue 2, pp.189 -193. Mar-Apr 2020.

Gandhi R. (2018, May 5). Naïve Bayes Classifier. <https://towardsdatascience.com/naive-bayes-classifier-81d512f50a7c>. Retrieved August 30, 2021.

Gleason, D.F. Histologic grading of prostate cancer: A perspective. Hum. Pathol. 1992, 23, 273–279

IBM Cloud Education (2020, July 15). Machine Learning. <https://www.ibm.com/cloud/learn/machine-learning>. Retrieved 28th July 2021

Igual. L, Segui. S. (2017). *Overview to Data Science*. Springer International Publishing Switzerland.

# Lutins. E. (2017, August 19). Ensemble Methods in Machine Learning: What are They and Why Use Them? <https://towardsdatascience.com/ensemble-methods-in-machine-learning-what-are-they-and-why-use-them-68ec3f9fef5f>. Retrieved 10 August 2021.

Mayo. M. (2021). Frameworks for approaching the machine Learning Process. <https://www.kdnuggets.com/2018/05/general-approaches-machine-learning-process.html>. Retrieved July 2021.

Merriam Webster (2020). <https://www.merriam-webster.com/dictionary/diagnosis>. Retrieved 28th July 2021

Melendez. I. (2019, May 16). Working with High Dimensional Data. <https://medium.com/working-with-high-dimensional-data/working-with-high-dimensional-data-9e556b07cf99#:~:text=SVM,-Support%20vector%20machines&text=SVMs%20are%20well%20known%20for,than%20the%20number%20of%20samples>. Retrieved 20 July 2021

Mukherjee. S. (2017, March 27). A.I. Versus M.D. What happens when diagnosis is automated? <https://www.newyorker.com/magazine/2017/04/03/ai-versus-md>. Retrieved 19th July 2021.

MuleSoft LLC. (2021). what is an API (Application Programming Interface)? <https://www.mulesoft.com/resources/api/what-is-an-api>. Retrieved 29th July 2021

Oyewo O.A, Boyinbode O.K. (2020). *Prediction of Prostate Cancer Using Ensemble of Machine Learning Techniques.* (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 11, No. 3, 2020

Panch T, Szolovits P, Atun R. (2018). *Artificial Intelligence, Machine learning, and Health systems.* www.jogh.org/DOI: 10.7189/jogh.08.020303.

Pedregosa, F.,Varoquaux, G.,Gramfort, A.,Michel, V., Thirion, B., Grisel, O.,Blondel, M., Prettenhofer, P. Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E. (2011). *Scikit-learn: Machine Learning in Python*. Journal of Machine Learning Research, vol 12. pp 2825 – 2830.

Pietrangelo. A Chun. C (2019, February 13). How Quickly Cancer Spreads. <https://www.healthline.com/health/cancer/how-fast-does-cancer-spread>. Retrieved July 15, 2021.

Rice, D. M. (2013). Casual Reasoning. Calculus of thought: neuromorphic logistic regression in cognitive machines. (pp. 95-125)Academic Press.

Russell. S, Norvig. P. (2010). “*Artificial Intelligence, A modern Approach*” (3rd ed). Pearson Education, Inc., Upper Saddle River, New Jersey 07458.

Singh T. (2020). Prostate Cancer: In-Depth Understanding, EDA, Model. . <https://www.kaggle.com/tanulsingh077/prostate-cancer-in-depth-understanding-eda-model>. Retrieved 15th July 2021.

UCLA Health (2021). What is Prostate Cancer? <https://www.uclahealth.org/urology/prostate-cancer/what-is-prostate-cancer>. Retrieved 12 July 2021.

The United States of America. The American Cancer Society (2021). Cancer Facts and Statistics. <https://www.cancer.org/research/cancer-facts-statistics.html>. Retrieved 15th July, 2021.

The United States of America. The American Cancer Society (2021). Key Statistics for prostate cancer. <https://www.cancer.org/cancer/prostate-cancer/about/key-statistics.html>. Retrieved 16th July 2021.

The American Society of Clinical Oncology (ASCO). (2021). Prostate Cancer: Overview. [https://www.cancer.net/cancer-types/prostate-cancer/Overview](https://www.cancer.net/cancer-types/prostate-cancer/introduction). Retrieved 16 July 2021.

The World Health Organization (W.H.O.). (2021, March 3). *Cancer.* <https://www.who.int/news-room/fact-sheets/detail/cancer>. Retrieved 1st August, 2021.

Yu. W, Liu. T, Valdez. R, Gwinn. M, Khoury. J.M. (2010). *Applications of Support Vector Machine Modeling for Prediction of Common Disease: the Case of Diabetes and Pre-Diabetes.* BMC Medical Informatics and Decision Making. <http://www.biomedcentral.com/1472-6947/10/16>

**Appendix.**

**MODEL TRAINING**

import pandas as pd

import numpy as np

from matplotlib import pyplot as plt

import joblib

# magic class is used here

%matplotlib inline

df = pd.read\_csv("../my project codes/Datasets/Prostate\_Cancer.csv")

print(df.head())

print(df.columns)

df.drop('id', axis = 1, inplace = True)

print(df.head(5))

classification = {'B':0, 'M':1}

df['diagnosis\_result'] = df['diagnosis\_result'].map(classification)

print(df.head(5))

df.shape

print(df.describe())

print(type(df.columns))

print(df.columns)

print(df.iloc[:, 1: 10])

def train\_and\_fit(df,model,model\_name):

# train test split

x\_train, x\_test, y\_train, y\_test = train\_test\_split(df.iloc[:, 1: 10], df['diagnosis\_result'], test\_size = 0.3)

model.fit(x\_train, y\_train)

preds = model.predict(x\_test)

#acc = accuracy\_score(y\_test, preds)

print(f'These are the results when using {model\_name}')

#print(f'Successfully trained model with an accuracy of {acc:.2f}')

plot\_confusion\_matrix(model, x\_test,y\_test)

print(classification\_report(y\_test, preds))

return model

# this is the import of the svc module form the sklearn module

# SVC stands for Support Vector Classification

from sklearn.svm import SVC

from sklearn.linear\_model import LogisticRegression

from sklearn.naive\_bayes import GaussianNB

# this splits arrays or matrices into random trains and train subsets

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

from sklearn.metrics import classification\_report, confusion\_matrix, plot\_confusion\_matrix

model\_svc = SVC()

model\_logistic = model = GaussianNB()

model\_nb = GaussianNB()

mdl = train\_and\_fit(df,model\_svc,"SVM")

mdl = train\_and\_fit(df,model\_logistic,"Logistic Regression")

mdl = train\_and\_fit(df,model\_nb,"Naive Bayes")

x\_train, x\_test, y\_train, y\_test = train\_test\_split(df.iloc[:, 1: 10], df['diagnosis\_result'], test\_size = 0.3)

from sklearn.model\_selection import GridSearchCV

grid = GridSearchCV(estimator = SVC(), param\_grid={'C': [0.1,1,10,100,1000,10000], 'kernel': ['rbf'], 'gamma':[1,0.1,0.01,0.001,0.0001,0.00001]}, refit=True, verbose = 3)

grid.fit(x\_train, y\_train)

grid.best\_params\_

pred = grid.predict(x\_test)

print(confusion\_matrix(y\_test, pred))

plot\_confusion\_matrix(grid, x\_test,pred)

from joblib import dump, load

joblib.dump(mdl, 'cancer\_model.mdl')

model\_columns = list(x\_train.columns)

joblib.dump(model\_columns, 'model\_columns.pkl')

**API BUILDING**

import flask

import io

import string

import time

import os

import numpy as np

import pandas as pd

import traceback

import joblib

from flask import Flask, jsonify, request, render\_template

#cancer\_model = joblib.load('../my project codes/cancer\_model.mdl')

app = Flask(\_\_name\_\_)

#This function is the contains the actual facilty that is used for the api. The endpoint is the /predict which is what you attach to the end of the

#url that allows you to call and use the built model.

@app.route('/predict', *methods*=['POST'])

def predict():

    if cancer\_model:

        try:

            json\_ = request.json

            query = pd.get\_dummies(pd.DataFrame(json\_))

            query = query.reindex(*columns*=model\_columns, *fill\_value*=0)

            prediction = list(cancer\_model.predict(query))

            print (prediction)

            #For some reason numpy and jsonify are quarelling so I just did the next two lines to by pass that

            #and ensure that the prediction is returned in postman

            pre = str(prediction)

            predicted = eval(pre)

            return jsonify(predicted)

            #return ''.join(prediction)

        except:

            return jsonify({'trace': traceback.format\_exc()})

    else:

        print ('Train the model first')

        return ('No model here to use')

@app.route('/', *methods*=['GET'])

def index():

    return 'Machine Learning Model For Final Year Project. For  quick clarification, ' \

           'the key for the API is simple if the output is 1 the tumour in question is Malignant and if the output is 0, ' \

           'then the output is Benign.  Thank you for choosing our API, it means a lot. '

if \_\_name\_\_ == '\_\_main\_\_':

    cancer\_model = joblib.load('../my project codes/cancer\_model.mdl')

    print("Loaded Model")

    model\_columns = joblib.load("../my project codes/model\_columns.pkl")

    print("Model Columns loaded")

    app.run(*debug*=True, *host*='0.0.0.0')