Project

MSc Applied AI and Data Science
2022
Edidiong Iton

Smart health predictive system for breast cancer recurrence

SOLENT UNIVERSITY
FACULTY OF BUSINESS LAW AND DIGITAL TECHNOLOGIES

MSc Applied AI and Data Science
Academic Year 2021-2022

Edidiong Iton

Smart Health Predictive System for Breast Cancer Recurrence

Supervisor: Peyman Heydarian

September 2022

This report is submitted in partial fulfilment of the requirements of Solent University for the degree of MSc Applied Artificial Intelligence and Data Science

Acknowledgements

Abstract

Breast cancer is one of the major causes of deaths in females worldwide. This project is focused on creating a system to identify the chances of the disease recurring in both symptomatic and asymptomatic patients in order to mitigate the chances of death and begin early treatment where necessary.

Contents

Figures

Tables

Acronyms

PICOT	- Patient, Intervention, Comparison, Outcome, Timeframe.
deg-malig	– Degree of malignancy
breast-quad	–Breast quadrant
irradiat	– Irradiation
KNN	-KNearest neighbours
SVM	-Support vector machines

1. Introduction

Technological advancements have led to the adoption of artificial intelligence knowledge in diverse industries with health care being one of such sectors.

A smart health prediction system is defined by Kamble *et al*. (2017) as a system that utilizes computing and technical knowledge to collect, retrieve and store useful health information for the optimization of medical processes. Wibamanto, Das & Chelliah (2020) emphasized the need for this predictive health system by identifying the issues that hinder the effective and timely maintenance of a patient's health by a medical institution such as disorganized data and health records, inability to provide immediate medical services, insufficient amount of qualified medical personnel etc. and tackling these issues will lead to the overall well being of citizens and the community as a whole.

This project aims at creating a smart health system to predict breast cancer as focusing solely on one disease will aid in prediction accuracy since similar symptoms commonly occur across numerous diseases. Another reason why breast cancer was selected is because it is one of the most commonly misdiagnosed form of cancer and the presumed cause of this is down to incomplete medical history, insufficient time to evaluate patients and missing information (McLaughlin & Lauricella PC) and adopting artificial intelligence knowledge and skills will aid in tackling these factors.

1.1. Background

Breast cancer is the most prevalent cancer found in women globally as a total of 2.3 million women were diagnosed with breast cancer at the end of 2020 with 685,000 recorded deaths due to the condition (WHO, 2022). Breast cancer is prevalent in both developed and less-developed counties, occurring in women of different age groups typically after puberty, however there is a greater risk of the disease occurring at a later age to women who have been through menopause (WHO, 2022) as statistics from breast cancer UK indicate that women born after 1960 have a 1 in 7 chance of developing the condition and NHS UK emphasizes that 8 out of 10 cases of breast cancer occurs in women over 50 (NHS, 2022)

The cause of breast cancer is not certain as there is no known reason why breast cancer affects one woman and not the other (NHS,2022) however, there are numerous factors that increase the risks of a person having cancer. Some of these factors include, a family history of breast or ovarian cancer, dense breast tissue, hormones or hormonal medicine, exposure to radiation and lifestyle factors ranging from alcohol use and obesity.

Recurrence in medical terms is where the cancer resurges after at least a year of remission to either the same organ, an organ that is close-by or an organ located in a different body part. Breast cancer recurrence events usually occur after initial diagnosis and treatment of a prior breast cancer incidence with a high likelihood of it being curable when detected early while the patient is asymptomatic. (Alva, 2018)

There are numerous instances where patients are unable to easily access medical consultation or services and one of such instance is a record of the covid-19 pandemic posing as a hindrance to breast cancer screening leading to an estimated total of 12,000 people living with undiagnosed breast cancer in the UK alone. Furthermore, statistics indicate that the amount of people referred for breast cancer checks have significantly fallen by more than 20,000 in 2020/21 compared to the previous year(Reynolds, 2022). This information demands for a convenient, reliable system to be put in place to categorically predict a patient's breast cancer recurrence status promptly and refer them to the appropriate medical professionals where necessary as early detection of this disease will prevent fatal outcomes and save numerous lives.

Data mining which deals with knowledge extraction of big data will contribute significantly as the nature of healthcare data is typically of a high volume with numerous information thus demanding the need for filtering and extracting useful information. Other artificial intelligence and machine learning models will be used to tackle the identified problems and predict breast cancer.

1.2. Research question

The research question addresses the problem that the project aims to solve, and the crucial elements from the derived research question are outlined utilizing the PICOT framework.

The PICOT framework is a mnemonic that represents patient, problem or population, intervention, comparison or control, outcome, timing. (Deng, 2020). For further emphasis, this framework defines the patient or problem in question, the proposed action to carry out, an alternative intervention considered, the expected outcome and the time frame to reach this desired outcome.

The research question is, "Can an artificial intelligence health system be implemented using prior breast cancer data to accurately predict the recurrence of breast cancer in both symptomatic and asymptomatic patients compared to the conventional breast screening method of mammogram."

- P – symptomatic and asymptomatic prior breast cancer patients
- I – smart health system using machine learning algorithms and techniques on breast cancer data of patients to determine possibility of breast cancer recurrence to facilitate early treatment as required.
- C - mammogram data of breast cancer patients
- O - percentage result to represent likelihood of breast cancer recurrence (percentage result to represent level of model accuracy to determine best model fit.)
- T - after a year of remission for patients

1.3. Aim

The aim of this project is to research and critically analyse data, journal articles and literature on breast cancer in its entirety and implement technical skills and knowledge to develop an innovative smart health system that will facilitate the precise prediction of breast cancer recurrence in patients to curtail the chances of patient's relapse and enable immediate treatment where necessary thus improving patients life expectancy.

1.4. Objectives

- To critically analyse similar studies on breast cancer prediction to gain an understanding on previous work conducted and identify areas of improvements.
- To select a suitable real-world breast cancer dataset to accurately perform analysis and prediction of breast cancer recurrence in patients.

- To propose a suitable artificial intelligence model(s) that best solves the identified problem with justification on model choice.
- To train the model using the data to be able to accurately decipher the likelihood of breast cancer recurring in a patient.
- To provide an avenue to refer a diagnosis of breast cancer to a medical professional for immediate treatment.
- To create a wireframe prototype based on research conducted to influence the model of the web-based system.

2. Literature Review

A comprehensive, critical review of journal articles and books on the research topic were carried out and the sources were obtained using the online academic database Science Direct. Relevant keywords were applied on this website to ensure the relevant literature is obtained. These keywords are, [Breast cancer, Breast cancer recurrence, Factors, Influence, Smart health system, Ethical impacts, Societal impacts, Artificial intelligence in healthcare] and the following inclusion and exclusion criteria were applied:

INCLUSION CRITERIA

- Studies highlighting the factors that contribute to breast cancer occurring and recurring.
- Studies outlining methods used and results acquired.
- Research conducted in the studies must have taken place within the last 10 years
- Published in peer review journals
- Research articles
- English language only

EXCLUSION CRITERIA

- Non-English language
- Non-peer reviewed sources
- Studies took place over 10 years ago
- Studies lacking methods and results

A mammogram is a breast screening method that doctors commonly adopt to detect breast cancer as it facilitates detection of the condition to people exhibiting symptoms. For the scope of this project, this method is not very ideal as it is usually geared towards symptomatic patients. It is also intrusive, and the users of the system may not have a complete trust to have their breast scanned and uploaded on a web system regardless of confidentiality assurance.

It has been emphasized that a mammogram also bears the risk of producing results that are false positive or false negative. In a false positive result, an anomaly is detected and is presumed to be a sign of breast cancer, but it is actually benign i.e., cells are non-cancerous. A false negative on the other hand is where the cancer is malignant and poses a risk of spreading to other organs but is not detectable as it appears to be concealed by normal breast tissues. (Nathiya and Sumitha, 2021).

The image below was retrieved from neeter et al. (2021) study of the diagnostic value of contrast-enhanced 2D mammography in everyday clinical use. In images B1 and B2 a mass was detected but it was a benign cell and in images C1 and C2 cancer was present, but the case was not classified as malignant by doctors.

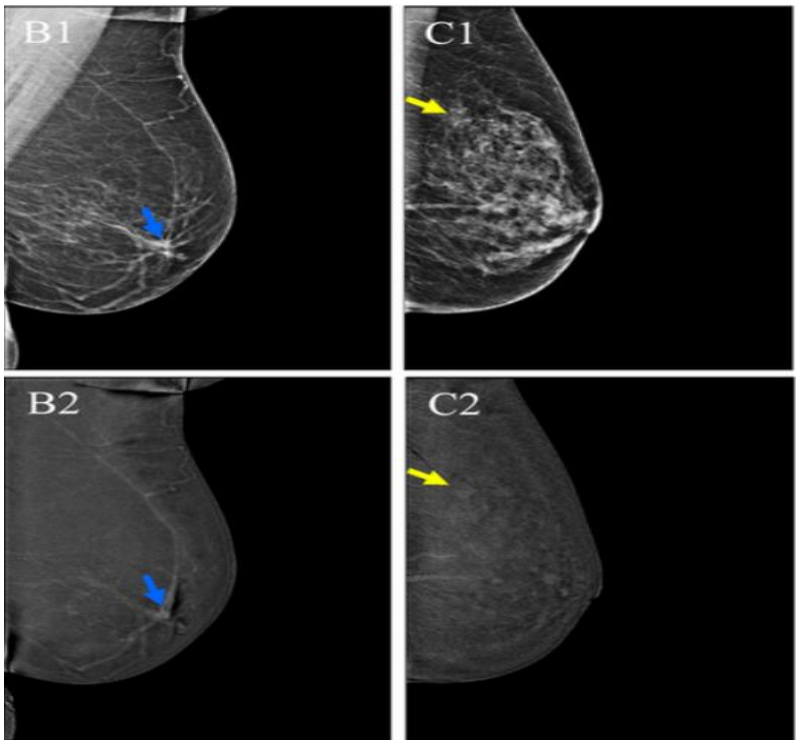


Figure 1: False positive and false negative in a mammogram result

These are the points that drove the researcher to explore alternative, innovative methods that would produce a more reliable system to conduct breast cancer predictions and from the research conducted, it was discovered that typically, the methods used for disease prediction in the healthcare industry are data mining and machine learning models.

Data mining is the extraction of useful information, discovery of patterns and acquisition of knowledge from a large dataset (Ahmad, Qamar and Rizvi 2015).

Numerous studies detail the use of data mining in the creation of a smart health prediction system. Supervised learning and unsupervised learning are two main methods used in machine learning. This project utilizes a supervised learning approach where the system is trained based on a given input and learns to generate results. A classification method which is a category of supervised learning will also be implemented as the nature of the dataset demands for this method.

A study that utilizes a combination of data mining and machine learning was illustrated by Mohapatra et al. (2018) in the article smart health care system using data mining. In it, a health care system that utilizes clustering techniques and K-means algorithm was proposed to predict heart disease, liver disease and chronic kidney disease.

The use of machine learning to perform predictions in breast cancer have also been carried out in various studies. A weighted decision tree model was proposed by Juneja et al. (2020) who used the Wisconsin breast cancer data set and breast cancer data set to predict the presence of cancer and in comparing the results to that of the results generated from using decision trees, naïve bayes, random trees and random forest classifier, the proposed weighted decision tree model performed better in accurately spotting breast cancer.

Muktevi utilized the machine learning algorithms, support vector machine, random forest, naïve bayes and logistic regression to a Kaggle breast cancer dataset to predict cancer and random forest produced the highest accuracy score of 98.24% compared to the other algorithms. Similarly, Ashok et al. compared logistic regression, random forest, K-Nearest Neighbor, support vector machine and decision tree, to determine which algorithm best predicts breast cancer using the Wisconsin breast cancer dataset and the random forest algorithm produced the best result in the comparative analysis with a 96.505 accuracy level.

Through the research conducted, it can be seen that utilizing different models produces different accuracy results with the exception of Muktevi and Ashok et al. study where random forest produced the highest accuracy score. This may be due to the datasets bearing similar attributes in common but as it can be seen, there is no standard algorithm that can be relied on to give the highest level of accuracy therefore selecting two or more algorithms is necessary to conduct comparisons with

parameters set to evaluate the algorithms. These parameters are; accuracy score, precision, recall and F1-score.

The chart below showcases the algorithms that are commonly used in healthcare industries to perform predictions. It can be seen that support vector machines and neural networks are the models implemented the most in numerous studies.

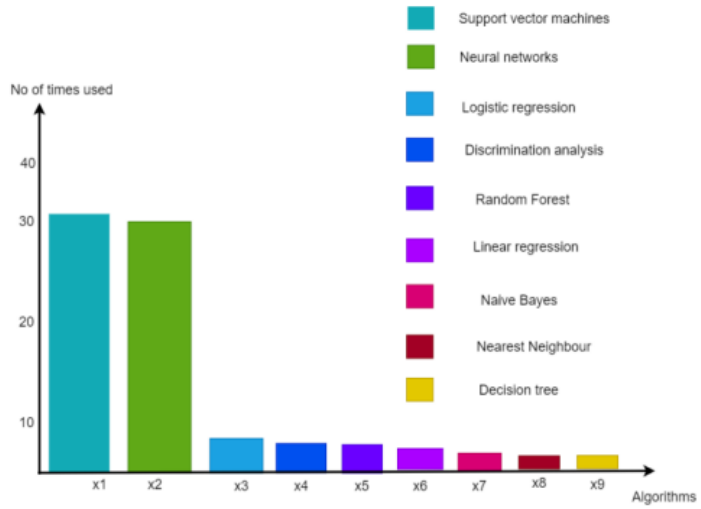


Figure 2: Machine learning models used in healthcare

This project will make use of the algorithms, support vector machines, decision trees, KNearest neighbour, random forest classifier and naïve bayes algorithm. It will be interesting to compare the results of a commonly used algorithm to those that are rarely used to see how much the results differ in prediction accuracy as this could influence the adoption of some of these models in future work.

2.1. Societal Impact

The implementation of a system that predicts the recurrence of breast cancer is advantageous as it bridges the gap in the shortcomings of current proceedings in the health sector. It also provides a number of benefits to citizens, the health industry and society as a whole. These benefits include,

- It is non-invasive therefore users can be comfortable engaging with the system.
- It saves time and resources that usually involve multiple medical personnel in the traditional method of breast cancer follow-up.

- The system is free thereby making it readily accessible to anyone who wants to perform a check-up and eliminating the costs of consultation.
- The efficacy of breast screening diagnostic techniques such as mammogram have been challenged in numerous medical trials and are reserved for use on patients exhibiting symptoms of breast cancer or its recurrence. This factor coupled with how expensive the procedure is demands for a more cost-effective reliable system such as this to cater to both asymptomatic and symptomatic patients.
- The effective implementation of this smart health system will increase the survival rates of breast cancer patients as early detection of the disease will commence treatments thus minimizing the risks of fatalities occurring.

2.2. Development

The following are the step-by-step processes to be undertaken in the project development phase.

- Data collection: Collect a suitable breast cancer data from a reliable source to perform analysis and aid in prediction.
- Data Preparation: Prepare the data by cleaning it i.e., handling missing data and conducting variable conversion where datatypes are misinterpreted.
- Exploratory data analysis: Explore the data to gain insights into the variables and this will be done with a combination of statistical and visualization analysis.
- Build the model: Import the necessary libraries and split the data into train and test for analysis. Utilize confusion matrix to compare parameter scores.

2.3. System Operation

The diagram below indicates that the system operates using a two tier architecture where a user fills in their requested details in the patient report page. The prediction will then be carried out for the patient based on input and if there is a likelihood of recurrence, then that information is made known to the patient and a contact list of qualified medical professionals is presented to refer the patient to book an appointment for immediate examination. On the backend, the dataset is trained using a machine learning model and algorithm to ensure accuracy in prediction and the database produces a result to respond to the patient's query.

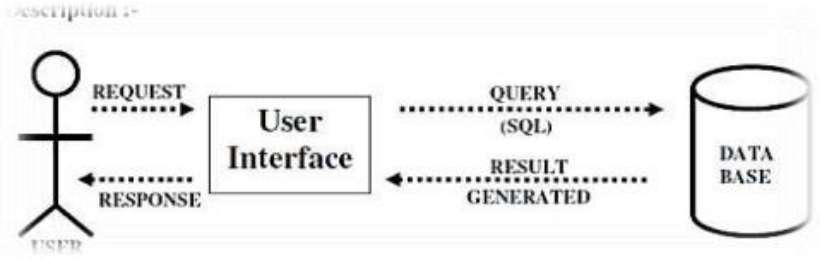


Figure 3: Two-tier architecture for the web-based system

2.4. System Features

This section details the features of the proposed smart health system for breast cancer. The system will have two main modules for the patient and the doctor.

Patient Module: Here a patient fills the form with the requested information and then clicks on the submit button where a result will be generated to inform them of their status.

Doctor Module: Here the doctor logs in with their details and views the patient report. Here they can perform consultations with the patient by giving advice on the next steps to take and schedule appointments for face-to-face meetings.

3. Methodology

This section covers the research scope by outlining the methods and techniques carried out in this project. A quantitative research method was utilized rather than a qualitative method or a mixed methods approach because the nature of the research problem demands for the collection of numerical data to perform analysis, discover patterns, and make predictions.

3.1. Quantitative Research method

Quantitative research methods could either be descriptive, correlational, or experimental (Bhandari, 2022). A descriptive research seeks to understand a phenomenon by relaying what, where, when and how queries. (McCombes, 2022).

An experimental research method deals with manipulating a target variable to evaluate its effects on the dependent variable (Bevans, 2022). This is an effective method as the researcher derives findings that are free from participant and researcher bias and results accrued are replicable in other settings however, despite its effectiveness, this method could not be applied in this project as it typically ought to i.e., with the researcher, sample breast cancer participants, variables and test hypothesis in a controlled environment because these are difficult to acquire and involves sensitive data and procedures that participants could find invasive. Furthermore, there is a risk of breaching ethical guidelines set by Solent University and possibly data protection regulations when human participants are involved in a healthcare context. With these factors put into consideration, the researcher measured the cause and effect of a target variable on the dependent variable by acquiring a real world breast cancer dataset comprising of information that will aid in analysis of the relationship between the variables and carried out this analysis using machine learning skills and algorithms on PyCharm environment.

Correlational research method is where the relationship between variables is examined without any manipulation or influence from the researcher (Bhandari, 2022) correlational analysis was carried out on the variables in the breast cancer dataset to investigate the strength of their association. The results of this research method are explained in the implementation section of this report.

3.2. Research sources

Secondary research was conducted using Google scholar along with the online academic database Science direct to acquire the relevant breast cancer literature. Science direct was a preferred option in conducting this research because it comprises of options that enables the researcher to precisely search for the articles needed through the aid of the keyword function and a filter in the form of an inclusion and exclusion criteria to ensure that only the relevant literature sources specified by the researcher are generated.

Using these functions, priority was given to articles released within the last 10 years to ensure that information obtained is up-to-date and still applicable in present day. The inclusion and exclusion criteria were also applied to ensure that literature sources comprised of methods used and results acquired from the research studies as this will enable the researcher to draw insights into the cause of certain outcomes and aid in comparative analysis in order to determine the best approach to take in solving the research problem. Google scholar lacks these useful attributes present in science direct but despite this, relevant literature sources were still obtained from this site, however one major setback faced with using google scholar is that some books and journal articles containing relevant information were inaccessible as they demanded for administrative access or were restricted from public view.

3.3. Dataset

The dataset for this project is the Wisconsin breast cancer data collected from the UCI machine learning repository. This data includes 201 occurrences of a single class and 85 occurrences of a different class totalling 286 instances altogether and it is summarized by 9 attributes. The data contains missing values which will have to be handled during data cleaning and preparation stage. The target variable in this data is the class variable as it determines if the recurrence event has occurred or not and the independent variables are the other 9 variables (age, menopause, tumor-size, inv-nodes, node-caps, deg-malig, breast, breast-quad, irradiat). The table below describes these attributes along with the information it is comprised of.

Table 1: Dataset composition

Attribute Name	Description	Information
Class	The event that the cancer has reappeared after a year of remission or not to decrease breast cancer risks	No recurrence events, recurrence events
Age	The patient's age at the time of diagnosis	10-19, 20-29, 30-39, 40-49, 50-59, 60-69, 70-79, 80-89, 90-99
Menopause	Menopausal status to determine whether or not a patient has ceased the period of menstruation	lt40, ge40, premeno
Tumor-size	The size of the tumour on the breast in diameter	0-4, 5-9, 10-14, 15-19, 20-24, 25-29, 30-34, 35-39, 40-44, 45-49, 50-54, 55-59.
Inv-nodes	The amount of lymph nodes that contain cancer cells	0-2, 3-5, 6-8, 9-11, 12-14, 15-17, 18-20, 21-23, 24-26, 27-29, 30-32, 33-35, 36-39.
Node-caps	Indicating whether node is present in the breast cap or not.	Yes, no
Deg-malig	It is the degree of malignancy with a grade range for the tumour indicating levels of cell anomaly i.e. breast cancer stages.	1,2,3
Breast	Either side of the breast where cancer could occur	Left, right
Breast-quad	The breast split into four quadrants with the nipple being the central point.	Left-up, left-low, right-up, right-low, central.
irradiat	Radiation therapy to eradicate cancerous cells.	Yes, no

3.4. Resources

The resources used for the effective implementation of this project include:

- PyCharm IDE: The environment for the development of the system where necessary libraries are imported for the effectual analysis of the best machine

learning algorithm that can be utilized and implemented for the prediction of breast cancer recurrence.

- Microsoft excel: Microsoft spreadsheet for the initial analysis of the breast cancer dataset.
- Machine learning algorithms: The appropriate five supervised machine learning algorithms that will best aid in prediction accuracy.
- Domain knowledge: A thorough understanding of the breast cancer community i.e., the terminologies used in breast cancer, the statistics involved, the actions that will help minimize risks etc.

4. Design and Implementation

This section explicitly reviews the process carried out by the researcher in developing the predictive model. Discussion on the data pre-processing carried out from data preparation, data cleaning to exploratory data analysis are expressed here along with accompanying code snippets as evidence to back up the explanations. The model evaluation is also discussed here.

4.1. Libraries used

The following are the libraries used in the PyCharm environment to develop the system.

- pandas
- numpy
- matplotlib
- seaborn

From sklearn:

- pre-processing
- Train_test_split
- GaussianNB
- DecisionTreeClassifier
- KNeighborsClassifier
- RandomForestClassifier
- SVC
- Accuracy_score, classification_report, confusion_matrix

4.2. Data pre-processing

Real-world data presents certain quality inconsistencies which could adversely compromise the performance of the model. Omer (2022) emphasizes that poor quality data produces inaccurate or wrong results thereby deriving unreliable conclusions to the research problem therefore, data pre-processing is a crucial stage that must be carried out to reduce these complexities and prepare the data for the machine learning model (Azevedo, 2022).

4.2.1. Data cleaning

Data cleaning is a step in data pre-processing that involves improving the quality of the data by identifying any missing values, irrelevant values, incomplete values, or

duplicates in the data and applying the suitable modifications to them (Azevedo, 2022).

Firstly, the data was opened in an excel spreadsheet and examined to ensure that there were no redundant columns or information lacking in the dataset and it was discovered that there were no columns for header names to their related rows. This was because the header names were specified in a separate document from the csv file containing the breast cancer data when obtained from the UCI machine learning repository. With this information, a column for the header names was added in PyCharm when loading the csv file to match their corresponding rows as seen in the code snippet below.

```
df = pd.read_csv('breast-cancer.csv', names=["Class", "age", "menopause", "tumor-size", "inv-nodes",
"node-caps", "deg-malig", "breast", "breast-quad", "irradiat"])
print(df)
```

```
                    Class    age menopause tumor-size inv-nodes node-caps  \
0    no-recurrence-events  30-39   premeno      30-34       0-2        no
1    no-recurrence-events  40-49   premeno      20-24       0-2        no
2    no-recurrence-events  40-49   premeno      20-24       0-2        no
3    no-recurrence-events  60-69      ge40      15-19       0-2        no
4    no-recurrence-events  40-49   premeno        0-4       0-2        no
..                    ...    ...       ...        ...       ...       ...
281     recurrence-events  30-39   premeno      30-34       0-2        no
282     recurrence-events  30-39   premeno      20-24       0-2        no
```

Figure 4: Add header names to the csv file

The next step in the data cleaning process was to check for any missing values using the isnull() function and it is vital to ensure that missing values found are handled appropriately as this can affect the accuracy result of the model if overlooked (Anon, 2021).

The code snippet below reveals that the dataset contained null values as the variables node-caps and breast-quad generated eight and 1 null values respectively when checked.

```
#check for null values
df.isnull().sum()
```

	data
tumor-size	0
inv-nodes	0
node-caps	8
deg-malig	0
breast	0
breast-quad	1
irradiat	0

Figure 5: Checking for missing values

On discovery of these missing values, the next process was to handle them. Studies on handling missing data illustrate several methods to use in tackling the problem ranging from dropping the rows or columns containing the missing values, imputing the missing values using mean, median or mode strategies to interpolation. However, before any values should be altered, the researcher needs to analyse the dataset or check the documentation to gain an understanding on why the data is missing (Dancuk, 2021).

The breast cancer dataset's documentation does not contain any information on why the data is missing but despite this, it should be noted that the node caps and breast quadrant variables comprise of crucial information that will be needed for analysis as the node caps indicates whether the lymph node is present in the breast cap or not and with more aggressive breast cancer cases, the lymph node could be replaced by the tumour and penetrate the capsule (Alva, 2018). The breast quadrant are the portion of the breasts when split into four. With this information taken into consideration, it was concluded that the null values should be replaced rather than drop their rows or columns.

The code snippet below indicates that the missing values were replaced with the string 'unknown'.

```
#handle missing data in the column by filling them with values
df["node-caps"].fillna("unknown", inplace = True)
#df
#print out specific row containing missing data
display(df.loc[163])
```

```
Class          no-recurrence-events
age                           60-69
menopause                      ge40
tumor-size                    25-29
inv-nodes                       3-5
node-caps                   unknown
deg-malig                         1
breast                        right
breast-quad                 left_up
irradiat                        yes
Name: 163, dtype: object
```

Figure 6: Replacing null values

After handling the missing values, the datatypes of the variables are explored to ensure that it is in a format that allows for statistical analysis to commence.
The code snippet below indicates what the initial datatypes of the variables were, and it can be seen that the system recognized all the variables with the exception of the deg-malig variable as categorical data by assigning an object datatype to them. It is understood that the variables age, tumor-size and inv-nodes comprise of quantitative data however the use of the hyphen(-) to indicate the number range prevents the system from recognizing the values as statistical data therefore it needs to be converted to an appropriate machine learning format.

```
#print the datatypes
print (df.dtypes)
```

```
Class          object
age            object
menopause      object
tumor-size     object
inv-nodes      object
node-caps      object
deg-malig       int64
breast         object
breast-quad    object
```

Figure 7: Initial datatypes

To commence the conversion of the string datatypes to numerical data, the class, node-caps and irradiat variables were binarized using the label encoding process. This process was best suited to these variables because they comprise of categorical data.

The code snippet below displays the label encoding process, and the result indicates that the string values in the rows of variables; class, node-caps and irradiat have been converted to binary values of 0 and 1 for their respective unique values.

```
#binarize class, node-caps and irradiat columns

new_data = df.copy()

encoder = preprocessing.LabelEncoder()

for col in ['Class', 'node-caps', 'irradiat'] :
    new_data[col] = encoder.fit_transform(new_data[col])

new_data.head()
#print(new_data.dtypes)
```

	Class	age	menopause	tumor-size	inv-nodes	node-caps	deg-malig	breast	breast-quad	irradiat
0	0	30-39	premeno	30-34	0-2	0	3	left	left_low	0
1	0	40-49	premeno	20-24	0-2	0	2	right	right_up	0
2	0	40-49	premeno	20-24	0-2	0	2	left	left_low	0
3	0	60-69	ge40	15-19	0-2	0	2	right	left_up	0
4	0	40-49	premeno	0-4	0-2	0	2	right	right_low	0

Figure 8: Label encoding class, node-caps and irradiat variables

The breast variable was converted from string to numerical data type using an ordinal encoding process. The code snippet below indicates that the left breast was assigned a value of 1 while the right breast was assigned a value of 2 and the new numerical values replaced the string values in the breast variable.

```
#Convert *Breast* string descriptive Information into number (Left = 1, Right = 2)
#Create a Dictionary of the mapping & Replace Values
breast = {'left':1, 'right':2}
new_data = new_data.replace({'breast': breast})
new_data.head()
```

	Class	age	menopause	tumor-size	inv-nodes	node-caps	deg-malig	breast	breast-quad	irradiat
0	0	30-39	premeno	30-34	0-2	0	3	1	2.0	0
1	0	40-49	premeno	20-24	0-2	0	2	2	3.0	0
2	0	40-49	premeno	20-24	0-2	0	2	1	2.0	0
3	0	60-69	ge40	15-19	0-2	0	2	2	1.0	0
4	0	40-49	premeno	0-4	0-2	0	2	2	4.0	0

Figure 9: Encoding the breast variable

The menopause variable was converted from string to numerical data type using an ordinal encoding process to indicate if the patient is pre or post-menopausal.

The code snippet below indicates that a dictionary containing the string value was created and assigned specific numerical values. The premeno was assigned a value of 1, ge40 was assigned a value of 2 and lt40 was assigned a value of 3 and these values were appended into the menopause variable.

```
#Convert *menopause* string descriptive Information into number.
#Create a Dictionary of the mapping & Replace Values
menopause = {'premeno':1, 'ge40': 2, 'lt40':3}
new_data = new_data.replace({'menopause': menopause})
new_data.head()
```

	Class	age	menopause	tumor-size	inv-nodes	node-caps	deg-malig	breast	breast-quad	irradiat
0	0	30-39	1	30-34	0-2	0	3	1	2.0	0
1	0	40-49	1	20-24	0-2	0	2	2	3.0	0
2	0	40-49	1	20-24	0-2	0	2	1	2.0	0
3	0	60-69	2	15-19	0-2	0	2	2	1.0	0
4	0	40-49	1	0-4	0-2	0	2	2	4.0	0

Figure 10: Encoding the menopause variable

The inv-nodes variable was converted from string to numerical data type using a dictionary to manually store the median of the range of numbers. These median values were then called to replace the string values in the inv-nodes variable.

The code snippet below indicates that the numerical values are now appended into the inv-nodes variable.

```
nodes = {'0-2':1, '3-5':4,'6-8':7,'9-11':10, '12-14':13,'15-17':16,'18-20':19,'21-23':22,'24-26':25,'27-29':28,'30-32':31,'33-35':34,
        '36-38':37,'39':39}
new_data = new_data.replace({'inv-nodes': nodes})
new_data['inv-nodes'] = new_data['inv-nodes'].apply(pd.to_numeric, downcast='integer', errors='coerce')
new_data[new_data.isnull().any(axis = 1)]
new_data = new_data.dropna()
new_data.head()
```

	Class	age	menopause	tumor-size	inv-nodes	node-caps	deg-malig	breast	breast-quad	irradiat
0	0	30-39	1	30-34	1.0	0	3	1	2.0	0
1	0	40-49	1	20-24	1.0	0	2	2	3.0	0
2	0	40-49	1	20-24	1.0	0	2	1	2.0	0
3	0	60-69	2	15-19	1.0	0	2	2	1.0	0
4	0	40-49	1	0-4	1.0	0	2	2	4.0	0

Figure 11: Encoding the inv-nodes variable

The age variable was converted from string to numerical data type using a dictionary to manually store the average of the range of numbers. These average values were then called to replace the string values in the age variable.

The code snippet below indicates that the specified numerical values are now appended into the age variable.

```
#Convert age to the numerical average of its average range.
age = {'20-29':24.5, '30-39':34.5,'40-49':44.5,'50-59':54.5, '60-69':64.5,'70-79':74.5,'80-89':84.5,'90-99':94.5}
new_data = new_data.replace({'age': age})
new_data.head()
```

	Class	age	menopause	tumor-size	inv-nodes	node-caps	deg-malig	breast	breast-quad	irradiat
0	0	34.5	1	30-34	1.0	0	3	1	2.0	0
1	0	44.5	1	20-24	1.0	0	2	2	3.0	0
2	0	44.5	1	20-24	1.0	0	2	1	2.0	0
3	0	64.5	2	15-19	1.0	0	2	2	1.0	0
4	0	44.5	1	0-4	1.0	0	2	2	4.0	0

Figure 12: Encoding the age variable

The tumor-size variable was converted from string to numerical data type using a dictionary to manually store the average of the range of numbers. These average values were then called to replace the string values in the tumor-size variable.

The code snippet below indicates that the numerical values specified are now appended into the tumor-size variable.

```
#Convert tumor-size to the numerical average of its average range.
Tumor = {'0-4':2, '5-9':7,'10-14':12,'15-19':17, '20-24':22,'25-29':27,'30-34':32,'35-39':37,'40-44':42,'45-49':47,'50-54':52}
new_data = new_data.replace({'tumor-size': Tumor})
new_data.head()
```

	Class	age	menopause	tumor-size	inv-nodes	node-caps	deg-malig	breast	breast-quad	irradiat
0	0	34.5	1	32	1.0	0	3	1	2.0	0
1	0	44.5	1	22	1.0	0	2	2	3.0	0
2	0	44.5	1	22	1.0	0	2	1	2.0	0
3	0	64.5	2	17	1.0	0	2	2	1.0	0
4	0	44.5	1	2	1.0	0	2	2	4.0	0

Figure 13: Encoding the tumor-size variable

After performing the data conversion through encoding, the datatypes were examined again to ensure that they are all of the expected numerical datatype.

The code snippet below indicates that the conversion process was successful as the former object datatypes are now integers and floats therefore statistical analysis can commence.

```
print (new_data.dtypes)
```

```
Class            int32
age            float64
menopause        int64
tumor-size       int64
inv-nodes      float64
node-caps        int32
deg-malig        int64
breast           int64
breast-quad    float64
irradiat         int32
```

Figure 14: New datatypes after encoding

The describe function was used to examine the mean, standard deviation, minimum and maximum value, and the first, second and third quartile represented as 25%, 50% and 75% respectively to aid in understanding the statistics of the data. The min, 25%, 50% and 75% are 0 in class, node-caps and irradiat because the binary values in these variables are labels rather than values that can be equated.

It can also be deduced that the average age of patients in this data is 54 and the minimum age of patients is 24 while the maximum age is 74.

```
new_data.describe()
```

	Class	age	menopause	tumor-size	inv-nodes	node-caps	deg-malig	breast	breast-quad	irradiat
ount	284.000000	284.000000	284.000000	284.000000	284.000000	284.000000	284.000000	284.000000	284.000000	284.000
mean	0.292254	51.154930	1.496479	26.383803	2.542254	0.415493	2.042254	1.471831	2.158451	0.235
std	0.455601	10.144222	0.548034	10.561475	3.406755	0.795242	0.736463	0.500087	1.194614	0.425
min	0.000000	24.500000	1.000000	2.000000	1.000000	0.000000	1.000000	1.000000	1.000000	0.000
25%	0.000000	44.500000	1.000000	22.000000	1.000000	0.000000	1.750000	1.000000	1.000000	0.000
50%	0.000000	54.500000	1.000000	27.000000	1.000000	0.000000	2.000000	1.000000	2.000000	0.000

Figure 15: Statistical analysis

4.3. Exploratory data analysis

Exploratory data analysis is a visual representation of the data, conducted to draw insights.

The diagrams below are histograms showing the distribution of the variables in relation to the target variable class features, recurrence events and no-recurrence events.

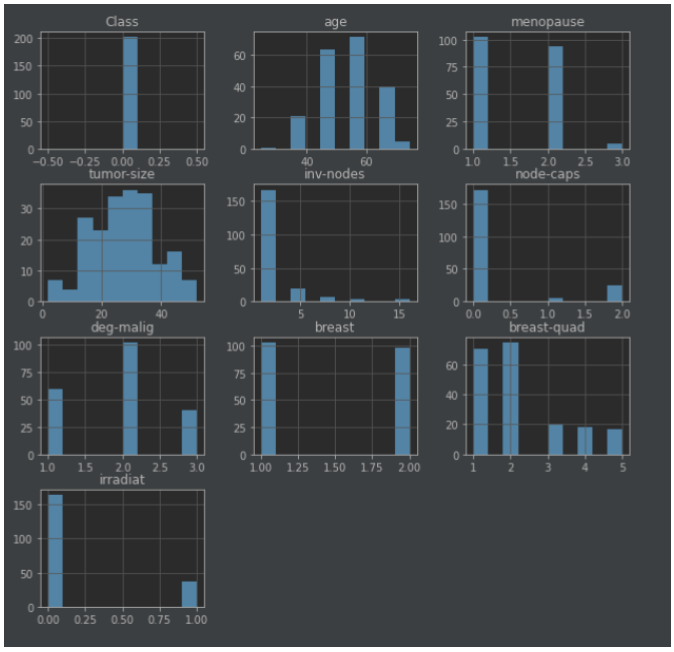


Figure 16: Histogram indicating no-recurrence events

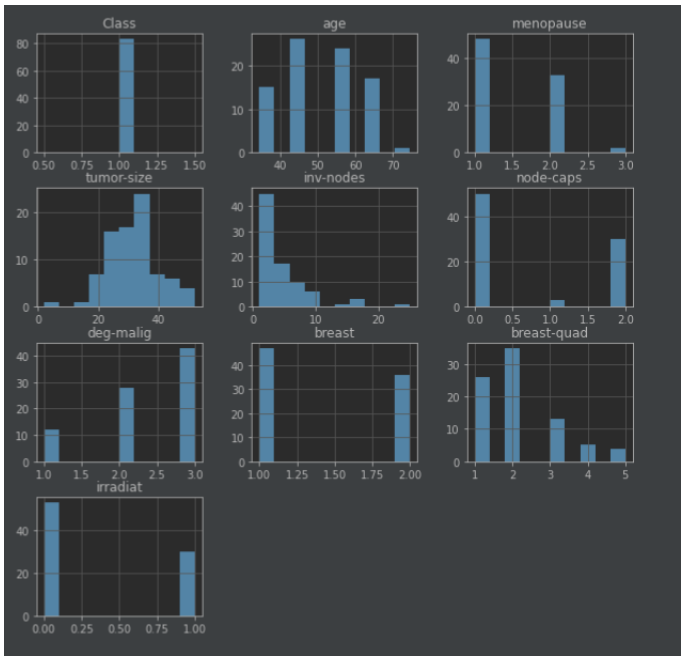


Figure 17: Histogram indicating recurrence events

A correlation matrix was created to examine the relationship between the variables to understand if there are any strong or weak correlations amongst them.

The diagram below shows that menopause and age have a strong positive correlation. This is because menopause is associated with older age in females.

```
plt.figure(figsize = (15, 10))

corr = new_data.corr()
mask = np.triu(np.ones_like(corr, dtype = bool))

sns.heatmap(corr, mask = mask, linewidths = 1, annot = True, fmt = ".2f")
plt.show()
```

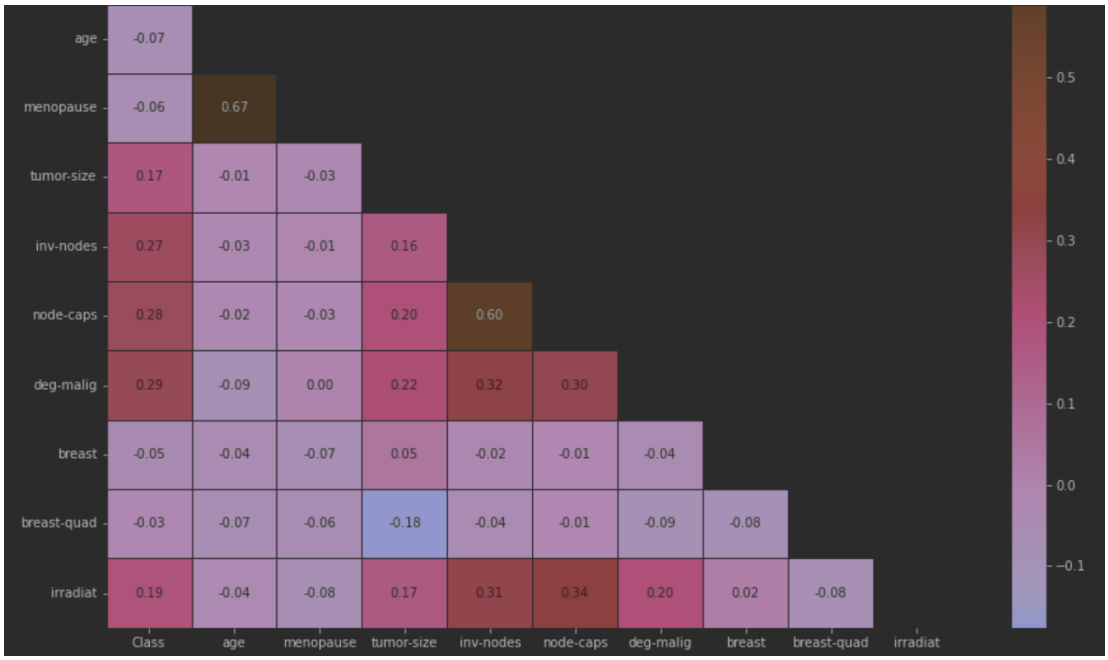


Figure 18: Correlation matrix

The code snippet below indicates that the features that are highly correlated should be dropped as it is redundant to have multiple features that will have the same effect on the model. The threshold for this action was set at 50%.

```
# # removing highly correlated features

corr_matrix = new_data.corr().abs()

mask = np.triu(np.ones_like(corr_matrix, dtype = bool))
tri_df = corr_matrix.mask(mask)

to_drop = [x for x in tri_df.columns if any(tri_df[x] > 0.50)]

df = df.drop(to_drop, axis = 1)

print(f"The reduced dataframe has {df.shape[1]} columns.")
```

Figure 19: Drop highly correlated features

4.4. Data modelling

Data modelling is a vital and critical aspect in development as this is where the data is trained and tested to be able to produce accurate predictions.

As stated earlier in the report, a thorough understanding of the data enlightens the researchers on what machine learning models to adopt. In this case the research problem is a classification problem therefore classification models are required. Five different classification models were used to compare prediction accuracy. The models used were KNeighbors classifier, decision tree classifier, naïve bayes classifier, random forest and support vector classifier and they are discussed briefly in this section along with justification on their selection.

The first step taken to model the data is to import the train_test_split library from sklearn. The train and test data is then split and to do that X and y variable names are called to store the independent variable and the target variable respectively. The data is split into X_train, X_test, y_train, y_test and a random state is set to iterate through the data. The train data is split 70% while the test data is split to 30%

```
X = np.array(new_data[[col for col in df.columns if col!='Class']])
y = np.array(new_data['Class'])

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.30, random_state=0, stratify=y)
```

Figure 20: Splitting the data for modelling

Data scaling was carried out to ensure the data is stabilized thus improving its performance.

```
scaler = StandardScaler()

X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)
```

Figure 21: Data scaling

4.4.1. KNeighbors classifier

This is a classifier that implements the k-nearest neighbors vote. It has been expressed in articles (Kumar, 2020) that this classifier works best when the dataset is small, appropriately labelled and noise-free. The pre-processed breast cancer dataset for this project fits these criterion perfectly.

This classifier takes on the following parameters; n-neighbors which signifies the number of neighbors to use with the default value being 5, however a value of 3 was set instead, weights is a function used for prediction with the default being uniform, metric is a function used for calculating distance and the default is minkowski, it works together with the power parameter 'p' to specify the type of distance i.e if its is standard Euclidean, minkowski or Manhattan distance for this modelling the standard Euclidean distance was used where p = 2.

The code snippet below indicates that the model generated an accuracy level result of 66% on the test data.

```
my_model = KNeighborsClassifier(n_neighbors = 3, metric = 'minkowski', p = 2)
my_model.fit(X_train, y_train)
y_pred = my_model.predict(X_test)
KN_model_accuracy = accuracy_score(y_test, y_pred)
print("Model accuracy on Test data: {:.2f}".format(KN_model_accuracy), '\n')

matrix_info = confusion_matrix(y_test, y_pred)
print("The Confusion Matrix: \n", matrix_info,'\n')

class_report = classification_report(y_test, y_pred)
print("Report of Classification: \n", class_report)
```

```
Model accuracy on Test data: 0.66

The Confusion Matrix:
 [[50 11]
 [18  7]]
```

Figure 22: KNeighbors model

4.4.2. Decision tree classifier

Decision trees is a supervised learning method that can be implemented in both classification and regression problems. A decision tree classifier is able to perform classification in both binary and multiclass data and it is easy to interpret.

The classifier takes on the following parameters; max_depth where the default is none but in the code snippet, the maximum depth of the tree is set at 4, min_samples_split which is the minimum amount of samples to split the internal node of the tree and the default is 2. Max_leaf_nodes is a parameter to set the best nodes with minimal impurity, in the code snippet, this parameter was set at 19. Criterion is the parameter used to weigh split quality and the default entry is gini.

The code snippet below indicates that the model generated an accuracy level result of 69% on the test data after the parameters were set.

```
my_model = DecisionTreeClassifier(max_depth=4, min_samples_split=2, max_leaf_nodes=19,
                                  criterion ="gini", random_state=0)
my_model.fit(X_train, y_train)
y_pred = my_model.predict(X_test)
DTC_model_accuracy = accuracy_score(y_test, y_pred)
print("Model accuracy on Test data: {:.2f}".format(DTC_model_accuracy), '\n')

matrix_info = confusion_matrix(y_test, y_pred)
print("The Confusion Matrix: \n", matrix_info,'\n')

class_report = classification_report(y_test, y_pred)
print("Report of Classification: \n", class_report)
```

```
 Model accuracy on Test data: 0.69

 The Confusion Matrix:
  [[52  9]
  [18  7]]
```

Figure 23: Decision tree classifier model

4.4.3. Naïve bayes classifier

The naïve bayes classifier are of various types; multinomial naïve bayes which deals with categorizing documents, Bernoulli naïve bayes where Boolean variables are utilized for predictions and gaussian naïve bayes where the variables are continuous values. In this case a gaussian naïve bayes classifier was selected. Naïve bayes algorithms work best when the predictor is independent. The class variable in the data is the independent variable.

The code snippet below shows that the gaussian naïve bayes classifier used does not take on any parameters and the model generated an accuracy level result of 70% on the test data after the parameters were set.

```
my_model = GaussianNB()
my_model.fit(X_train, y_train)
y_pred = my_model.predict(X_test)
NB_model_accuracy = accuracy_score(y_test, y_pred)
print("Model accuracy on Test data: {:.2f}".format(NB_model_accuracy), '\n')

matrix_info = confusion_matrix(y_test, y_pred)
print("The Confusion Matrix: \n", matrix_info,'\n')

class_report = classification_report(y_test, y_pred)
print("Report of Classification: \n", class_report)
```

```
Model accuracy on Test data: 0.70

The Confusion Matrix:
 [[52  9]
 [17  8]]
```

Figure 24: Naïve bayes model

4.4.4. Random forest classifier

Random forest is an estimator that fits an amount of decision tree classifiers on sub-samples of the data. This model utilizes an average method to handle over-fit and improve prediction accuracy hence why it was selected for this dataset.

The classifier takes on the following parameters: criterion which is used to weigh split quality and the default entry is gini but in the implementation entropy was used instead. Max_depth was set at 2 and the min_samples_leaf, min_sample_split, n_estimators and max_features were set to the random forest classifier in-built default values.

The code snippet below shows that the random forest classifier model generated an accuracy level result of 74% on the test data after the parameters were set.

```
my_model = RandomForestClassifier(criterion = 'entropy', max_depth = 2, max_features = 'sqrt', min_samples_leaf = 2,
                                  min_samples_split = 2, n_estimators = 100)
my_model.fit(X_train, y_train)
y_pred = my_model.predict(X_test)
RFC_model_accuracy = accuracy_score(y_test, y_pred)
print("Model accuracy on Test data: {:.2f}".format(RFC_model_accuracy), '\n')

matrix_info = confusion_matrix(y_test, y_pred)
print("The Confusion Matrix: \n", matrix_info,'\n')

class_report = classification_report(y_test, y_pred)
print("Report of Classification: \n", class_report)
```

```
Model accuracy on Test data: 0.74

The Confusion Matrix:
 [[60  1]
 [21  4]]
```

Figure 25: Random forest model

4.4.5. Support vector machines

Support vector machines are used for classification, regression, and detection of outliers. This model allows the specification of kernel functions therefore it is flexible. The kernel was specified during modelling as linear, and this will adopt the one-versus-one approach where each built classifier trains data from two classes and a random state was set at 0.

The code snippet below indicates that the support vector machines classifier model generated an accuracy level result of 70% on the test data after the parameters were set.

```
my_model = SVC(kernel = 'linear', random_state = 0)
my_model.fit(X_train, y_train)
y_pred = my_model.predict(X_test)
svc_model_accuracy = accuracy_score(y_test, y_pred)
print("Model accuracy on Test data: {:.2f}".format(svc_model_accuracy), '\n')

matrix_info = confusion_matrix(y_test, y_pred)
print("The Confusion Matrix: \n", matrix_info,'\n')

class_report = classification_report(y_test, y_pred)
print("Report of Classification: \n", class_report)
```

```
 Model accuracy on Test data: 0.70

 The Confusion Matrix:
  [[52  9]
  [17  8]]
```

Figure 26: Support vector machine model

5. Discussion

This section discusses the findings acquired from the implementation of the models. It should be noted that the accuracy score defined by each of the models in the implementation section is rounded up to the nearest number therefore there are no inconsistencies between the score generated from the code snippets in the model implementation and the tabular result below.

Table 2: Model evaluation

Models	Accuracy score
KNeighbors classifier	66.27%
Decision tree classifier	68.60%
Naïve bayes classifier	69.76%
Random forest classifier	74.41%
Support vector classifier	69.76%

After carrying out the model accuracy tests, it can be seen that the random forest classifier bears the highest accuracy score at 74% compared to the other classification models used. It is understood that this score is fairly low, for this model to be implemented in a health care field and there are several limitations that were identified in the process of carrying out the implementation specifically from the selected dataset which when tackled, will help improve the model accuracy.

Firstly, the variables for age, size of the tumour and number of lymph nodes were provided in ranges and as such they needed to be manually modified to numeric formats of integers and floats. This modification gives rise to bias and inaccuracies in the data and although the quality of the data was good enough for the machine to process, the model loses valuable insights that could have been acquired from the variables.

The dataset also lacks several risk factors associated with breast cancer recurrence such as, family history, hormone receptor status, HER2 status etc. Inclusion of these variables would aid in improving the prediction accuracy score. Addition of these variables and other risk factors will also aid in the development of a reliable application system with input fields for patients to be able to select their symptoms and receive a referral for treatment where necessary as originally proposed as one of the objectives of this project.

6. Conclusion and Further Work

This project sought to develop a smart health system to facilitate the precise prediction of breast cancer recurrence in symptomatic and asymptomatic patients and curtail the chances of patient's relapse. However, there were certain roadblocks that hindered this overall development of the system thus leading to the development of only the back end of the system.

Beginning from the choice of dataset, the size of the data was small therefore it is not a reliable representative of the probability of breast cancer recurring. It also lacked the variables and information required to build a system that is sustainable, reliable and fulfil user's needs. The risk factors of breast cancer which is a crucial factor was absent from the dataset therefore it would have been ineffective in predicting the possibilities of breast cancer recurrence.

Being that this is also a system intended for the clinical domain, a high level of accuracy in model prediction is required to mitigate any associated risks as humans are involved. Unfortunately, the selected models could not achieve the expected threshold due to the insufficiencies in the dataset.

The UCI machine learning repository have another breast cancer data with more data instances with variables about the cancer cell nuclei. These variables contained in this data can be used to predict breast cancer. However, it can not be used to predict its recurrence neither can an application be made using it.

With these factors put into consideration, the recommendations for future research studies will be for clinicians to populate a breast cancer dataset with extensive variable factors that could influence the chances of breast cancer recurring, following ethical guidelines. This would enable researchers and developers to build a software that will accurately predict this disease and even extend the same process in dealing with other high risks diseases and mitigating its effects.

7. References

Ahmad, P., Qamar, S. and Rizvi, S.Q.A., 2015. Techniques of data mining in healthcare: a review. *International Journal of Computer Applications*, *120*(15).

Alva, N., 2018. *Using machine learning techniques to predict the recurrence of breast cancer*. [online] Linkedin.com. Available at: <https://www.linkedin.com/pulse/using-machine-learning-techniques-predict-recurrence-breast-alva> [Accessed 3 July 2022].

Archive.ics.uci.edu. 2022. *UCI Machine Learning Repository: Breast Cancer Data Set*. [online] Available at: <https://archive.ics.uci.edu/ml/datasets/Breast+Cancer> [Accessed 8 July 2022].

Ashok Deulkar and J.A. Laxminarayana, "Breast Cancer Prediction using Machine Learning Technique", *Journal of Emerging Technologies and I*.

Azevedo, N., 2022. *6 Techniques of Data Preprocessing | Scalable Path®*. [online] Scalable Path. Available at: <https://www.scalablepath.com/data-science/data-preprocessing-phase> [Accessed 9 August 2022].

Bevans, R., 2022. *A Quick Guide to Experimental Design | 5 Steps & Examples*. [online] Scribbr. Available at: <https://www.scribbr.co.uk/research-methods/guide-to-experimental-design/> [Accessed 8 August 2022].

Bhandari, P., 2022. *What Is Quantitative Research? | Definition & Methods*. [online] Scribbr. Available at: <https://www.scribbr.co.uk/research-methods/introduction-to-quantitative-research/#:~:text=Quantitative%20research%20methods%20%20%20%20Research%20method,To%20assess%20whether%20attitudes%20towards%20clim%20...%20> [Accessed 8 August 2022].

Bhandari, P., 2022. *Correlational Research | Guide, Design & Examples*. [online] Scribbr. Available at: <https://www.scribbr.co.uk/research-methods/correlational-research-design/> [Accessed 8 August 2022].

Bouchrika, I., 2021. [online] Research.com. Available at: <https://research.com/research/how-to-write-a-research-question> [Accessed 11 August 2022].

Dancuk, M., 2021. *Handling Missing Data in Python: Causes and Solutions*. [online] Knowledge Base by phoenixNAP. Available at: <https://phoenixnap.com/kb/handling-missing-data-in-python#:~:text=1%20Import%20and%20View%20the%20Data.%20Download%20the,Depending%20on%20the%20data%20type%20and%20the%20> [Accessed 9 August 2022].

Deng, D., 2020. *PICO, PICOTS, PICOTT Framework for Clinical Questions as a Way to Design Clinical Trials*. [online] Onbiostatistics.blogspot.com. Available at: <https://onbiostatistics.blogspot.com/2020/03/pico-picots-picott-framework-for.html> [Accessed 7 August 2022].

Hossain, M.E., Khan, A., Moni, M.A. and Uddin, S., 2019. Use of electronic health data for disease prediction: A comprehensive literature review. *IEEE/ACM transactions on computational biology and bioinformatics*, *18*(2), pp.745-758.

Juneja, K. and Rana, C., 2020. An improved weighted decision tree approach for breast cancer prediction. *International Journal of Information Technology*, *12*(3), pp.797-804.

Kamble, N., Harmalkar, M., Bhoir, M. and Chaudhary, S., 2017. Smart Health Prediction System Using Data Mining. *International Journal of Scientific Research in Computer Science Engineering and Information Technology*, *2*(5).

Kumar, A., 2020. *KNN Algorithm: What?When?Why?How?*. [online] Medium. Available at: <https://towardsdatascience.com/knn-algorithm-what-when-why-how-41405c16c36f#:~:text=KNN%20is%20one%20of%20the,how%20its%20neighbor%20is%20classified.&text=KNN%20classifies%20the%20new%20data,dataset%20of%20tomatoes%20and%20bananas.> [Accessed 9 August 2022].

McCombes, S., 2022. *Descriptive Research Design | Definition, Methods & Examples*. [online] Scribbr. Available at: <https://www.scribbr.co.uk/research-methods/descriptive-research-design/> [Accessed 8 August 2022].

Mohapatra, S., Patra, P.K., Mohanty, S. and Pati, B., 2018, December. Smart health care system using data mining. In *2018 International Conference on Information Technology (ICIT)* (pp. 44-49). IEEE.

Muktevi Srivenkatesh, "Prediction of Breast Cancer Disease Using Machine Learning Algorithms", *International Journal of Innovative Technology and Exploring Engineering*, vol. 9, no. 4, pp. 2868-2878, 2020.

Nathiya, S. and Sumitha, j. (2021) "A Comparative Study on Breast Cancer Prediction using Optimized Algorithms," *2021 2nd International Conference on Smart Electronics and Communication (ICOSEC)*, pp. 1401-1405, doi: 10.1109/ICOSEC51865.2021.9591787.

Neeter, L. & Raat, H. & Meens-Koreman, S. & Stiphout, R. & Timmermans, S. & Duvivier, K. & Smidt, Marjolein & Wildberger, J. & Nelemans, P. & Lobbes, Marc. (2021). The diagnostic value of contrast-enhanced 2D mammography in everyday clinical use. Scientific Reports. 11. 22224. 10.1038/s41598-021-01622-7.

nhs.uk. 2022. *Breast cancer in women - Causes*. [online] Available at: <https://www.nhs.uk/conditions/breast-cancer/causes/> [Accessed 2 July 2022].

Reynolds, L., 2022. *Facts and figures | Breast Cancer UK*. [online] Breast Cancer UK. Available at: <https://www.breastcanceruk.org.uk/about-breast-cancer/facts-figures-and-qas/facts-and-figures/> [Accessed 8 July 2022].

Who.int. 2022. *Breast cancer*. [online] Available at: <https://www.who.int/news-room/fact-sheets/detail/breast-cancer> [Accessed 8 July 2022].

Wibamanto, W., Das, D., & Chelliah, S.A. (2020). Smart Health Prediction System with Data Mining. *International Journal of Current Research and Review* DOI: http://dx.doi.org/10.31782/IJCRR.2020.122332

8. Appendices

Model evaluation

```
my_model_eva = pd.DataFrame({'my_model': ['KNeighbors Classifier', 'Decision Tree Classifier', 'Naive Bayes Classifier', 'Random Forest
Classifier', 'Support Vector Classifier'],
                             'Accuracy': [KN_model_accuracy * 100, DTC_model_accuracy * 100,  NB_model_accuracy * 100, RFC_model_accuracy
                              * 100, svc_model_accuracy * 100]})
my_model_eva
```

	my_model	Accuracy
0	KNeighbors Classifier	66.279070
1	Decision Tree Classifier	68.604651
2	Naive Bayes Classifier	69.767442
3	Random Forest Classifier	74.418605
4	Support Vector Classifier	69.767442

Handling missing data for breast quadrant

```
#handle missing data in the column by filling them with values
df["breast-quad"].fillna("unknown", inplace = True)
#print out specific row containing missing data
display(df.loc[206])
```

```
Class          recurrence-events
age                        50-59
menopause                   ge40
tumor-size                 30-34
inv-nodes                    0-2
node-caps                     no
deg-malig                      3
breast                      left
breast-quad              unknown
irradiat                      no
Name: 206, dtype: object
```

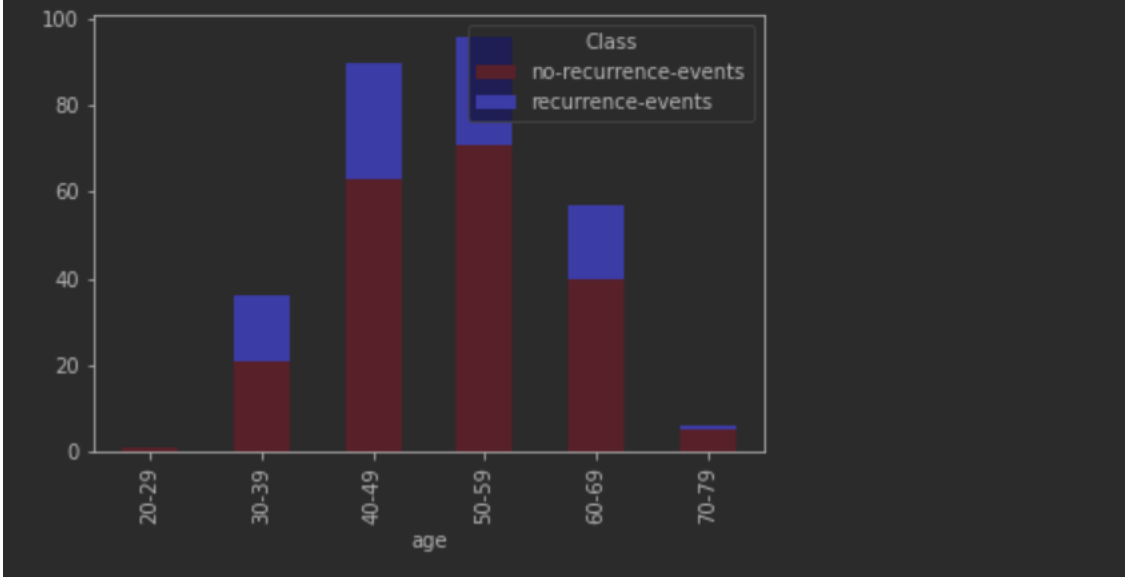
Simple imputer function attempt

```
imputer = SimpleImputer(strategy='most_frequent')
imputer.fit(df)
new_data = imputer.transform(df)
new_data = pd.DataFrame(new_data)
print(new_data.tail(10))
```

```
     recurrence-events  50-59  premeno  35-39  15-17  yes  3  right
278  recurrence-events  50-59  premeno  35-39  15-17  yes  3  right
279  recurrence-events  50-59     ge40  40-44    6-8  yes  3   left
280  recurrence-events  50-59     ge40  40-44    6-8  yes  3   left
281  recurrence-events  30-39  premeno  30-34    0-2   no  2   left
282  recurrence-events  30-39  premeno  20-24    0-2   no  3   left
283  recurrence-events  60-69     ge40  20-24    0-2   no  1  right
284  recurrence-events  40-49     ge40  30-34    3-5   no  3   left
285  recurrence-events  50-59     ge40  30-34    3-5   no  3   left
```

Bar plots indicating the age of patients with no recurrence or recurrence cases.

```
x.plot(kind='bar', stacked=True, color=['pink', 'blue'], grid=False)
plt.show()
```



This was an attempt to conduct feature importance using the random forest classifier to determine what features are most relevant to the target variable the outcome of this code showed that the deg-malig variable was of significant prominence to the target variable class while menopause, breast and irradiat where of low significance to the target variable.

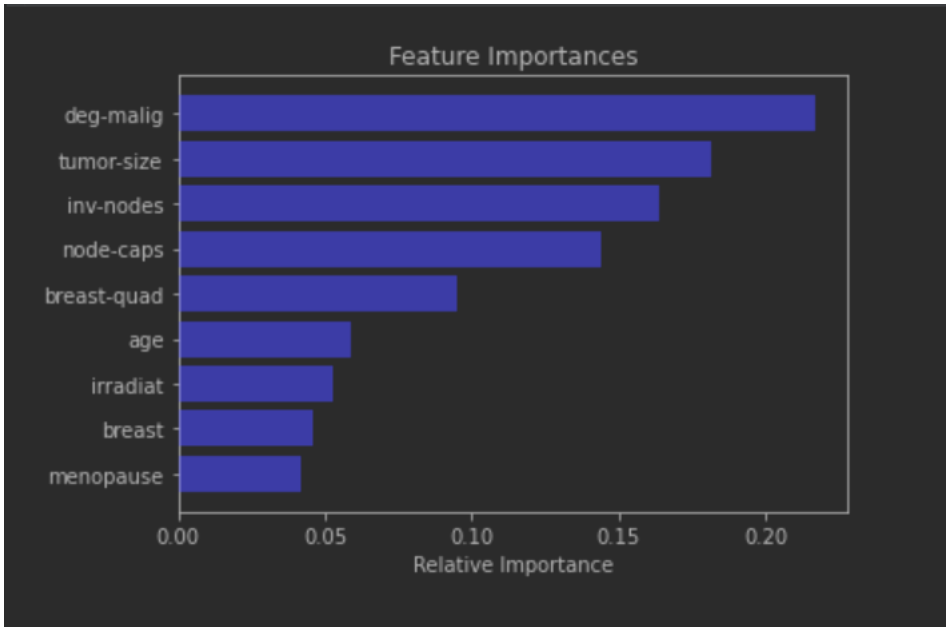
```
cols = ["Class", "age", "menopause", "tumor-size", "inv-nodes", "node-caps", "deg-malig", "breast", "breast-quad", "irradiat"]

# drop target variable and symbol column because symbol is categorical data
X_data= df.drop(['Class'], axis=1)


rf_model = RandomForestClassifier(random_state=1, max_depth=4)
rf_model.fit(X_data,new_data.Class)

train_features = X_data.columns
importances = rf_model.feature_importances_
indices = np.argsort(importances)[-9:]
plt.title('Feature Importances')
plt.barh(range(len(indices)), importances[indices], color='b', align='center')
plt.yticks(range(len(indices)), [train_features[i] for i in indices])
plt.xlabel('Relative Importance')
plt.show()
```

```
Traceback…
ValueError: could not convert string to float: '30-39'
```



Classification report for svc model

```
class_report = classification_report(y_test, y_pred)
print("Report of Classification: \n", class_report)
```

```
 Report of Classification:
               precision    recall  f1-score   support

           0       0.75      0.85      0.80        61
           1       0.47      0.32      0.38        25

    accuracy                           0.70        86
   macro avg       0.61      0.59      0.59        86
weighted avg       0.67      0.70      0.68        86
```