A black background with white text

Description automatically generated with low confidence

MSc Data Science Project

7PAM2002-0509-2023

Department of Physics, Astronomy and Mathematics

**Data Science FINAL PROJECT REPORT**

**Project Title:**

Feature Importance and Cluster Analysis for Predictive Accuracy on Wisconsin Breast Cancer data.

**Student Name and SRN:**

Susan Vincent

21060799

Supervisor: Michael Kuhn

Date Submitted: 29/08/2024

Word Count: 6886

DECLARATION STATEMENT

This report is submitted in partial fulfilment of the requirement for the degree of Master of Science in Data Science at the University of Hertfordshire.

I have read the guidance to students on academic integrity, misconduct and plagiarism information at [Assessment Offences and Academic Misconduct](https://www.herts.ac.uk/__data/assets/pdf_file/0007/237625/AS14-Apx3-Academic-Misconduct-v17.0.pdf) and understand the University process of dealing with suspected cases of academic misconduct and the possible penalties, which could include failing the project module or course.

I certify that the work submitted is my own and that any material derived or quoted from published or unpublished work of other persons has been duly acknowledged. (Ref. UPR AS/C/6.1, section 7 and UPR AS/C/5, section 3.6). I have not used chatGPT, or any other generative AI tool, to write the reportor code (other than where declared or referenced).

I did not use human participants or undertake a survey in my MSc Project.

I hereby give permission for the report to be made available on module websites provided the source is acknowledged.

Student Name printed: Susan Vincent

Student Name signature: Susan Vincent

Student SRN number: 21060799

UNIVERSITY OF HERTFORDSHIRE

SCHOOL OF PHYSICS, ENGINEERING AND COMPUTER SCIENCE

**Abstract**

In this research, machine learning methods in diagnosis of breast cancer are analyzed with reference to the Wisconsin Breast Cancer dataset. This study intends to use logistic regression, random forest, SVM and K-means clustering and Gaussian mixture models to define what features are the most effective in differentiating between benign and malignant tumors. The research also discusses the capability of the unsupervised learning for discovering the latent structures within the data. The research outcomes show the practical use of these models in early identification and call for utilization of machine learning in clinical application.

**Table of Contents**

[1 Introduction 5](#_Toc174401353)

[1.1 Research Questions 5](#_Toc174401354)

[1.2 Objectives 5](#_Toc174401354)

[2 Background 6](#_Toc174401357)

[2.1. Dataset 6](#_Toc174401358)

[2.2. Literature Review 6](#_Toc174401359)

[3 Methodology 11](#_Toc174401361)

[3.1. Overview 11](#_Toc174401363)

3.2. [ML Algorithms 11](#_Toc174401364)

[3.2.1. Logistic Regression 11](#_Toc174401365)

[3.2.2. Random Forest Classifier 11](#_Toc174401366)

[3.2.3. Support Vector Machine (SVM) 12](#_Toc174401367)

[3.2.4. Principal Component Analysis 12](#_Toc174401368)

[3.2.5. KMeans Clustering 13](#_Toc174401369)

[3.2.6. DBSCAN 13](#_Toc174401370)

[3.2.7. Gaussian Mixture Model 13](#_Toc174401371)

[3.4. EDA and visualization 14](#_Toc174401372)

[4 Results 24](#_Toc174401372)

[4.1. Logistic Regression Results 24](#_Toc174401372)

[4.2. Random Forest Results 24](#_Toc174401372)

[4.3. Support Vector Machine Results 24](#_Toc174401372)

[4.4. KMeans Clustering 25](#_Toc174401372)

[4.5. DBSCAN Clustering 26](#_Toc174401372)

[4.6. Gaussian Mixture Model Clustering 27](#_Toc174401372)

[5. Conclusion 28](#_Toc174401372)

[6. Future Work 29](#_Toc174401372)

[7. Legal, Ethical and Professional Issues 29](#_Toc174401372)

[8. References 30](#_Toc174401372)

[7. Appendices 32](#_Toc174401372)

# **Introduction**

Various studies say that breast cancer is one of the most common cancers among women, where abnormal growth of a mass of tissue, cause the expansion of malignant cells resulting in severe breast cancer. Breast cancer is incurable mostly because of its spreading nature that frequently spreads to distant organs of the body and the bone. Early diagnosis can lead to a high survival rate. Any delay in detection can cause rapid spread of the illness and treatment-related complications.

Traditional methods used in detecting breast cancer involve mammogram images and histopathological analysis of the biopsied tissue. This process takes a lot of time, effort and requires a lot of specialized medical expertise and sometimes false detection of mammograms can risk the patient’s life. Machine learning models can be used to overcome these issues. With the exponential growth in the performance of Machine Learning and Artificial Intelligence this process can be speed up to provide a lot of support to the medical practitioners in this field (Barth, 2024).

In this research we use “Wisconsin Breast Cancer” dataset where the data is taken from UCI Machine Learning repository (Karanam, 2022). In this project machine learning is used to analyse this data and understand which feature is affecting the predictions using various algorithms. Understanding the importance of each feature is very important as it can help to enhance model accuracy and provide further insights into the behaviour of the tumour itself and also about the tumour’s progression.

Another approach is to use unsupervised learning techniques like clustering that have the capability to reveal the inherent structures in the data that may not be very apparent to a human viewing the data. By grouping patients together, it is possible to understand if a group of patients have a feature that all have same type of target variable classification. In this way the application of ML in the diagnosis can help to understand various aspect of the disease and help to provide more insights into the behaviour of the tumour.

## **Research Questions**

1. What are the most important features for classifying malignant tumours from benign tumours using Wisconsin breast cancer data?
2. Can unsupervised learning algorithms be used to detect clusters among the data that have an association with breast cancer?

## **Objectives**

* Use the Wisconsin breast cancer data to develop multiple supervised and unsupervised machine learning algorithms.
* Compare the performances of supervised ML algorithms among each other in tumour classification.
* Develop multiple unsupervised algorithms and check their associations with the known classifications (benign and malignant).
* To find out the most important features in the classification using ML algorithms.

# **Background**

## **Dataset**

The Wisconsin Breast Cancer data, which was used in this study, is publicly available through the UCI Machine Learning repository. William Wolberg, Olvi Mangasarian, and W. Nick Street are the creators of the dataset. Details about the breast tissue of several cancer patients are included in this dataset. With over 550 observations and 32 features with different properties, this dataset represents several tumor characteristics in patients. The target variable in this dataset is classified as malignant (a cancerous tumor) and benign (a non-cancerous tumor). Radius, perimeter, smoothness, area, and concavity are a few of the dataset's important characteristics.

Dataset: <https://archive.ics.uci.edu/dataset/17/breast+cancer+wisconsin+diagnostic>

## **Literature Review**

The paper by (Delshi Howsalya Devi and Deepika, 2015) will also seek to establish the efficiency of each of the said clustering algorithms in the diagnosis of breast cancer particularly in differentiating between benign and malignant tumors. It uses the Wisconsin Breast Cancer dataset, which contains 699 instances with nine integer-valued attributes and is split into two classes: Benign and malignant. The study employs five different clustering algorithms: there are DBSCAN, Farthest First, Canopy, LVQ, and Hierarchical clustering, with the experimentation done using Weka. Evaluating and comparing these clustering algorithms' reliability for predicting breast cancer outcomes is the study's main objective.

The core methodology of the study is using the data mining tool Weka to construct and test the clustering methods. The efficiency of each of those techniques is evaluated in this research using the Wisconsin Breast Cancer dataset as the source of data. To determine a possible method for breast cancer diagnosis, the studies of each clustering technique are evaluated and quantified. As compared to other clustering techniques, the results showed a considerable improvement in the accuracy of the Farthest First algorithm, which achieves an accuracy of 72%.

Firstly, the data is extracted from the source which in this case was the UCI Machine Learning Repository; preprocessing and then the clustering algorithms. DBSCAN is another clustering technique, which is available in the given region. The strength of DBSCAN is in the detection of non-spherical clusters and the indifference to noise, however, sets containing a large difference between cluster densities of areas are problematic for DBSCAN. Farthest First clustering is a simple modification of k-means whereby the initial cluster centers are located at the greatest distance from existing centers and is faster in clustering than standard k-means because reassignment and adjustment are infrequent. The kind of clustering that is used prior to k-means or hierarchy for example is the canopy clustering which involves placing data in clusters based on two distance parameters, T1 and T2. LVQ which is a competitive learning algorithm, group data items where the clusters are gained by learning the center of the clusters, and associating new data items to the nearest cluster. Hierarchical clustering is a type of clustering that forms a tree structure where each node represents a cluster, its techniques can combine clusters to form new clusters, or it can split larger clusters into smaller clusters leading to a formation of dendrogram to represent the clusters.

Farthest First method has a better average Euclidean distance compared to Near Neighbors method; it produces a correct prediction of 72%. This is an important result because it can be seen that Farthest First is more appropriate for clustering of the data set namely Wisconsin Breast Cancer. Still, the paper gives little regard to why Farthest First is superior to the other approaches and when Farthest First is likely to be more or less potent. Further, the study does not assess how different clustering techniques are affected by parameters in order to get better results that may be significant.

The paper seems to fail to consider other critical measures that would give a better comparison of the effectiveness of the clustering algorithms. For instance, measures like precision, recall, the F-measure, or the F1 score, which are important in diagnosing the efficiency of a diagnostic tool in a medical setting, are omitted. The lack of such measurements makes the work insufficient to give a proper evaluation of the clustering algorithms. In addition, the question of how to deal with missing values that are typical for real-life medical datasets is beyond the scope of the given study. The paper also briefly notes that the further research would focus on the missing data, but it does not offer any concrete approaches and methods regarding such data.

This paper (Marne, Churi and Marne, 2020) addresses the problem of breast cancer which has become the second cause of death among women through a predictive model employing machine learning. It uses K-means clustering to classify breast cancer data and analyses the probability of cells being cancerous and Decision Trees classification. This model leverages the Wisconsin Breast Cancer dataset from the UC Irvine Machine Learning Repository, which includes 699 instances with 11 distinct attributes, 9 of which describe the physical characteristics of cell nuclei present in a digitized image of a fine needle aspirate (FNA) of a breast mass. The dataset is categorized into two classes: The benign and the malignant types of the tumor where 458 benign patients and 241 patients with malignant tumor reported.

The following is the breakdown of the theoretical framework used to develop the methodology. Clustering is one of the unsupervised learning techniques that aim at categorizing data into clusters that are most similar. The K-means algorithm adopts a process of clustering of data that is by minimizing the variance between the clusters. In the present work, the given algorithm is applied with the help of orange tool which is the environment for data analysis and visualization. The data is then fed into the orange environment where the K-mean clustering procedure is applied so as to cluster the instances. The clustering process is illustrated through scatter plots, which makes it possible for example to analyze how one or another attribute correlates with the presence of cancerous cells. For example, there are certain features like bland chromatin and normal nucleoli and these form the basis of how clusters of malignancy of the cells are determined.

Supervised learning technique is utilized and the classifier chosen is the Decision Tree. Decision Trees sort data points by using a process of dividing dataset into subsets according to the input features value. The concept of the algorithm is that the initial split will be made on a feature which has the highest information gain. Information gain is calculated using entropy which is also a measure of the amount of ‘impurity’ in a set of data. The basis on which the decision is made is the information gain, with the node that yields the highest value selected for the splitting of the data; this process is a recursive one and is continued until the so-called terminal nodes or leaves of the tree have been generated or all of the data points in the data set have been classified into the same class. In the present work, the Decision Tree model is developed in Python and the dataset is divided into train and test datasets for the validation of the model. The training set contains 60% of the data set and on the other hand the testing set having 40% data of the data set.

The performance of the Decision Tree model is therefore measured as the number of correct predictions against the overall number of predictions. The employed model reaches the accuracy of 94%. As much as 16% of patients were successfully predicted to contain cancerous cells, which stipulate a high reliability of the approach. This accuracy is quite a milestone because if applied clinically, it indicates that the Decision Tree algorithm has a near perfect capability of classifying breast cancer instances from the dataset provided. The high accuracy rate also supports the utilization of machine learning algorithms in early identification and diagnosis of breast cancer as performed by the medical experts.

This paper also describes the steps taken to perform the data preprocessing in order to get ready for the analysis. All values that are disregarded as missing are archived; therefore, the dataset has only 683 rows. This step is very important because missing and incomplete data greatly affect the performance of the machine learning models. This helps the study to have a clean record of the inputs to be analyzed hence making the results more accurate and reliable.

That is why, visualization of the data is crucial for the analysis of the clustering and classification outcomes. The distribution of attributes and their relationship with the cancerous cells are also depicted in the study, employ graphical displays including scatter diagram, bar chart, and pie chart. Such visualizations allow defining which attributes are the most important for the breast cancer prediction.

The last proposition of the paper expresses the necessity of early diagnosis and subsequent treatment of breast cancer. The model that has been proposed in this paper involves the use of K-means clustering and Decision Trees classification; the proposed model has the benefit of being comprehensive in assessing the likelihood of cancerous cells in tissues taken from the breast. Hence, the high accuracy rate of the model confirms the usefulness of the model for diagnosing the breast cancer disease and its clinical applicability by medical researchers and practitioners. There is also a further research implication in terms of exploring more machine learning algorithms and data sets for improving the model’s prediction capability. The conclusions are therefore leading to the understanding of data mining techniques and their application are very useful in the development of medical diagnostic as well as in the enhancement of the health of patients.

The paper (Radha and Rajendiran, 2014) aims at assessing how K-means clustering algorithm could be used to examine breast cancer data with an aim of determining genes that help in prognosis of breast cancer. Breast cancer is one of the main diseases affecting the female population throughout the world; it is especially essential to diagnose it in the early stages. This research is centered on employing data mining approach to analyze the gene expression values of breast cancer patients, with special interest on the biomarker for classification of patients using some characteristics. The work also discusses the results of K-means clustering analysis applied to the dataset of breast cancer gene expression with similar studies with the goal to increase the knowledge about the breast cancer multiplex and its molecular basis.

Information about breast cancer is given in the introduction of the work; breast cancer is an illness that displays symptoms of uncontrolled cell division. They point that for cancer to be formed the genes that control cell growth as well as differentiation must be changed. The study also underscores the contribution of GWAS in the discovery of genetic markers including SNPs that put the patient at risk for breast cancer. Nonetheless, it points out that the findings of GWAS give only a boundary framework of these genetic variants and their roles especially concerning the emergence of breast cancer subtypes. The introduction also gives a background on the microarray technology that has allowed the discovery of molecular markers and the classification of the different classes of breast cancer through mRNA expression patterns.

K-means clustering technique is discussed in the methodology section along with its features. K-means clustering is applied because of its simple and fast application. In K-means clustering the data points are divided into a number of clusters where each data point can be a part of only one cluster. Dataset which the study uses comes from M. Zwitter and M. Soklic from University Medical Centre, Institute of Oncology in Ljubljana, Yugoslavia.

The study uses the Orange open source, Python as well as the K-means clustering methodology. The dataset consists of 286 tuples with nine attributes, including the classification of menopausal status into three categories. There were also subgroups named “premeno,” “ge40,” and “it40,” depending on the tumor size. It is done in three stages where the distance measure and the procedures of initialization are different in each stage with aim to obtain the best clustering.

The random initialization only applies to the formation of the clusters in the first place and the Manhattan distance measure is used in the first phase. They use proximity between the data points in order to cluster the data and the solutions are then judged on the basis of the likelihood of classifying the data correctly. The study gives a mean classification accuracy of 66.99% in this phase. In Phase II, the distance to centroids serves as the evaluation criterion, the distance being the Euclidean one, and the initialization is random. This phase increases the proposed average classification accuracy by an average of 74. 98%. In the third comparison of the same parameters, distance measure in CLARANS program is used with Pearson Correlation while initialization is with agglomerative clustering. This phase yields the perfect score of 1 or 100% in classification, which confirms that the used features Pearson Correlation and agglomerative clustering is useful in the separation of the different clusters with respect to tumor size and menopausal status.

In the data analysis part, the authors describe the outcomes of clustering in all the phases in detail. In this phase, the clusters are described according to the tumor size ‘ge40’ with large tumors and ‘premeno’, ‘it40’ with small tumors. In Phase II new cluster patterns are driven by size and large tumor sizes are clubbed under ‘premeno’ and small ones under ‘ge40’ and ‘it40’. The results achieved in Phase III are considerably better than in the Phase II: the clusters are clearly distinguishable; therefore, 100% of the objects are classified correctly. In this study, the clustering results are presented in the form of scatter plot diagrams since they assist in the comparison of the tumor size and the phase and or the menopausal status.

The conclusion of the study restates the use of K-means clustering in assessment and evaluation of attributes of breast cancer. This paves way for the fact that by varying different measures like distance measures, scoring methods and inauguration techniques, the process of clustering can be exploited to generate highly accurate classification. Three methods that were tested were found out to be efficient in this case, namely, agglomerative clustering-Pearson Correlation. It is also shown that the results of clustering algorithms can be and should be combined with radiologists’ decisions to increase the possibility of a correct diagnosis of breast cancer. As for the future directions, it points to probabilistic clustering algorithms and other similar sophisticated approaches towards the improvement of the examination of the breast cancer data. This approach may help in the identification of the biomarkers and also in classifying the type of breast cancer correctly, ultimately helping the patients and their prognosis.

# **Methodology**

## **Overview**

This project was done in Google Colab in Python language. Libraries such as Pandas, Matplotlib, Seaborn and NumPy are imported for loading datasets, creating qualitative statistical graphics and visualizations. For preprocessing the dataset the structure of dataset is analyzed and observed that there is no missing value.

Several libraries were imported from Sklearn and are used for model selection. For training the model the dataset is split into 70% training and 30% test datasets. The features in both sets are standardized to have a mean of 0 and standard deviation of 1. Various supervised and unsupervised models were trained with the train data and made predictions on the test data.

## **ML Algorithms**

### **Logistic Regression**

A classifier or a model employed in binary classification to predict the probabilities of a categorical nature. For classification, two different techniques are utilized: logistic regression and decision tree. Logistic regression is used for binary classification. The logistic function or sigmoid function, which only produces values between 0 and 1, is used to evaluate the risk of an outcome equal to a numerical number with two interpreted values (for example, yes or no) (Banoula, 2020). The dependent variable is frequently binary in nature; if the trait is present, it may be recorded as 1, otherwise as 0.

Through the use of a linear combination of inputs and a logistic function, the model also demonstrates the relationship between the dependent and independent variables. The advantage of this model is explained by the possibility of interpreting the coefficients that reflect the effect of each variable. The basic structure of logistic regression, in contrast to other models, accounts for its popularity in fields like finance and medical research.

However, in the case of complicated data structures, it is highly misleading as it assumes a linear relationship between independent variables and log-odds of the result. Furthermore, it assumes that there is no significant correlation between any two independent variables. However, L1 and L2 can be used to address overfitting, which is a major issue in machine learning (Aditya P, 2018).

### **Random Forest Classifier**

Random Forest Classifier is one of the powerful and stable classification and regression techniques in the machine learning. It is a type of classification technique, similar to the Random Forest, only that it creates multiple Decision Trees and combines them into a single model to have a better and more stable forecast. By constructing each tree from a random subset of the data and features, there is far less overfitting and more generalization (Sruthi, 2021).

Bootstrap sampling creates every tree by partitioning the initial data into number of subsets linked to the various features at each node. The forest makes the final decision based on all the trees’ predictions, including using simple voting, for example, based on the majority. Random Forest deals with high dimensionality and can work with missing values and outliers and handle interactions. It also gives out estimates of feature importance (Tanner, no date).

However, the issue with Random Forest is that when dealing with big data sets and lots of trees, it is quite complicated and time-consuming. This is mainly due to the fact that it is not very interpretable since the ensemble creates a ‘black box’.

### **Support Vector Machine (SVM)**

SVM is a type of classification technique, used in data mining. In many classification tasks, its function to find the best hyperplane that separates data points of different classes. This hyperplane is a line in two dimensions and a plane or a hyperplane in the higher dimensions as from the three dimensions (Berwick and Village Idiot, no date).

Support vectors machine aims at achieving the maximum margin between the closest separable data points of opposite classes. This makes for good generalization to unseen data. SVM deals with both linear separable and non-separable data. For non-linear cases, the programme applies ‘the kernel trick’ to transform it into a higher dimension so that a linear hyperplane can be used to separate classes. Some are linear, polynomial and radial basis function (RBF).

Text classification, bioinformatics, and image recognition are some of the applications of the method, which is widely used. Nonetheless, SVM is slower than other methods given the current data size and can be slow even with large data sets, also, it contains several parameters which have to be tuned properly like C, and kernel functions settings (Dhiraj, 2019).

### **Principal Component Analysis (PCA)**

PCA is an extremely powerful data analysis and machine learning tool, even though it is mostly employed for dimensionality reduction. In order to preserve as much variability as possible, the number of variables is decreased while main components are extracted from the original variables where there is correlation.

PCA determines axes of maximal variability of the data, and the first principal component captures the maximal amount of variance. Following components are supposed to capture maximum variability orthogonal to the previous components. It makes the visual representation of data easier, filters out noise and also enhances the performance of a model (Premanand, 2022).

Among other fields, it is used in image processing, genomics, and finance. Additionally, PCA does not include class labels, which somewhat limits its usefulness in some classification scenarios.

### **KMeans Clustering**

An unsupervised learning algorithm which aims to divide data according to its features into K clusters, where each instance is located as close to the other instances in the same cluster as it is to the instances in the other clusters.

K-Means initiates with K random centroids, of which they can be selected randomly, or selected through smart methods like K-means++. After that points are divided into clusters with the help of nearest centroid. Centroids are redefined as the mean of the points of given cluster, and the points are reallocated to the closes set of new centroids. This process continues until centroids do not change or for a certain number of iterations is reached (Education Ecosystem (LEDU), 2018).

It is particularly useful with big data sets, such as samples with numerous clusters, and is computationally quick and simple to apply. However, it requires knowing the number of clusters advance and only produces good results if the clusters are spherical and equal in size, which often not the case. Another drawback of K-Means is that it is sensitive to noise and the original placement of these centroids; as a result, different outcomes may be achieved depending on the influence.

### **DBSCAN (Density-Based Spatial Clustering of Applications with Noise)**

"Density-Based Spatial Clustering of Applications with Noise," or DBSCAN for short, is the name of an unsupervised clustering algorithm that can form a number of meaningful clusters out of the data points that are most condensed and are able to identify points that are density minorities. DBSCAN has an advantage over K-Means in that it doesn't require a fixed number of clusters. Rather, it labels the points in low density areas as noise and identifies clusters as regions with a high point density.

Two primary parameters are used by DBSCAN: min\_samples, which is the lowest number of points required to form a dense area, and epsilon (ε), which is the greatest distance between two points that can be regarded as neighbors. The first step in the process is selecting a random point and examining its surroundings. If it reaches the density adjusted value, it essentially starts a new cluster and grows outward with other like points neighboring the current one. Those which do not meet the criterion are referred to as noise points.

DBSCAN is deserved for arbitrary shapes of the clusters and noise which is good for datasets which contain irregular clusters or noise. It also doesn’t fare well for mixing density with the same ε value not working for the entire dataset. However, there are some limitations with DBSCAN, and that is when it comes to large datasets, it can also be computationally expensive (Satpati, 2023).

### **Gaussian Mixture Model (GMM)**

One kind of probability model for clustering, density estimation, and pattern recognition is the Gaussian Mixture Model (GMM). According to GMM, data originates from several Gaussians, each of which represents a cluster. Unlike hard clustering techniques like K-Means, the GMM calculates the probability density that each point belongs to a certain Gaussian.

Like most other GMM, the Expectation-Maximization (EM) algorithm is employed in the estimation of the account. During Expectation step, the function finds the probability of the point to belong to each cluster. These probabilities used for refines in the Maximization step the parameters of the Gaussian distribution such as means, variances and the mixing coefficients. For parameters then to stabilize, this process continues iterating (Singh, 2019).

GMM is particularly useful in cases where two clusters overlap since it takes into account the random uncertainty in assignment. GMM, however, makes the assumption that all distributions are Gaussian, which may not always be the case. Additionally, it is essential to determine the number of clusters, and choosing the appropriate number is a difficult task.

## **EDA and Visualization**

Fig. 1: Distribution of Diagnosis

The distribution of diagnoses within the dataset is shown in the first as a bar plot. Two categories exist: This study uses two classifications of breast cancer: malignant (labeled 1) and benign (labeled 0). The figure makes it evident that there are more observations in the "benign" class than in the "malignant" class, a characteristic that can be seen in many medical datasets that are used to try and detect diseases early on.

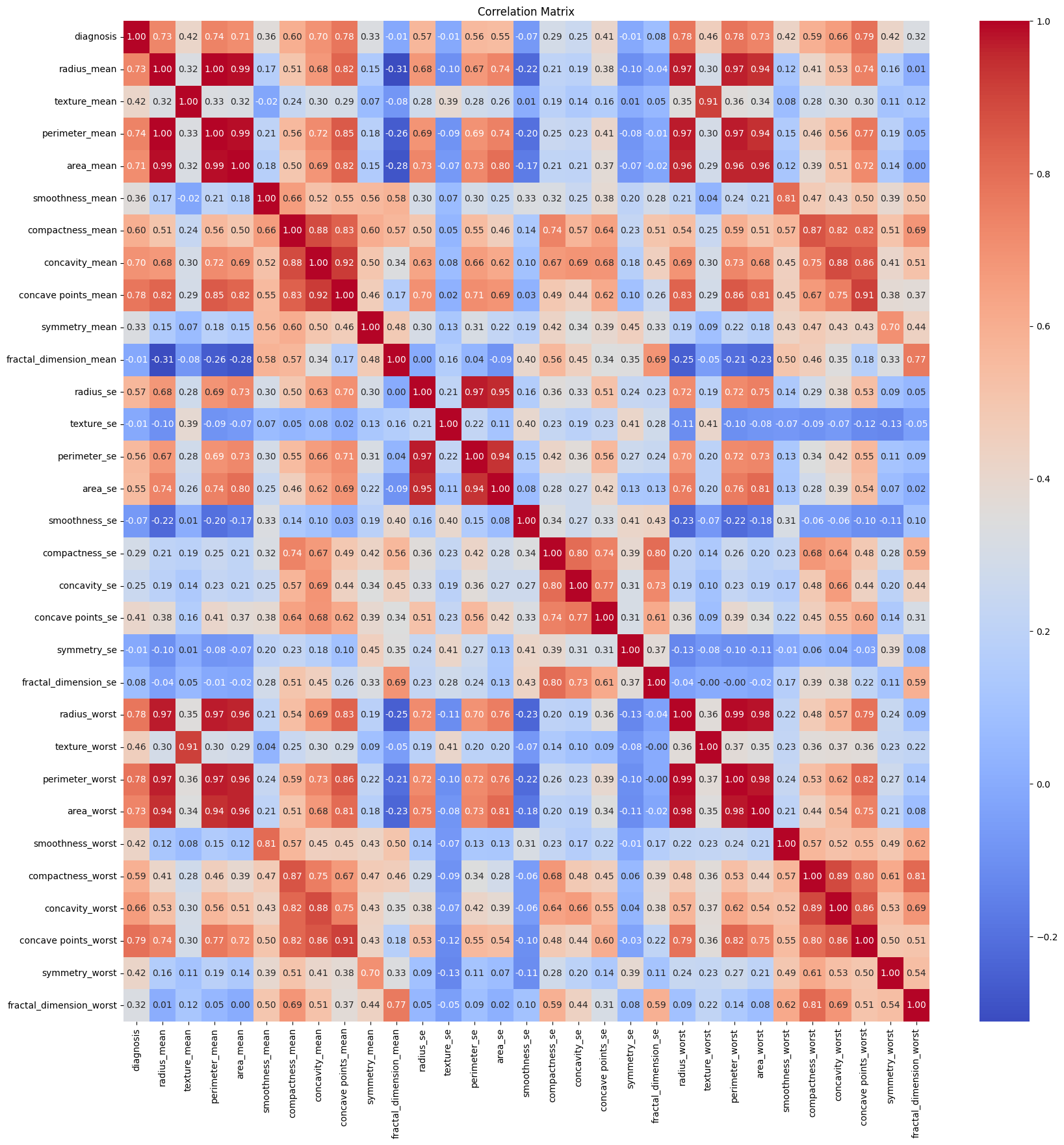


Fig. 2: Correlation Matrix

The second image is the heatmap plot representing the correlation between different features in the dataset. It is very useful to know about the pairwise correlation between features for feature selection or engineering. The degree of the color indicates the level of the correlation, the denser the color is, stronger the correlation. Zero shows there is no linear relationship between features.

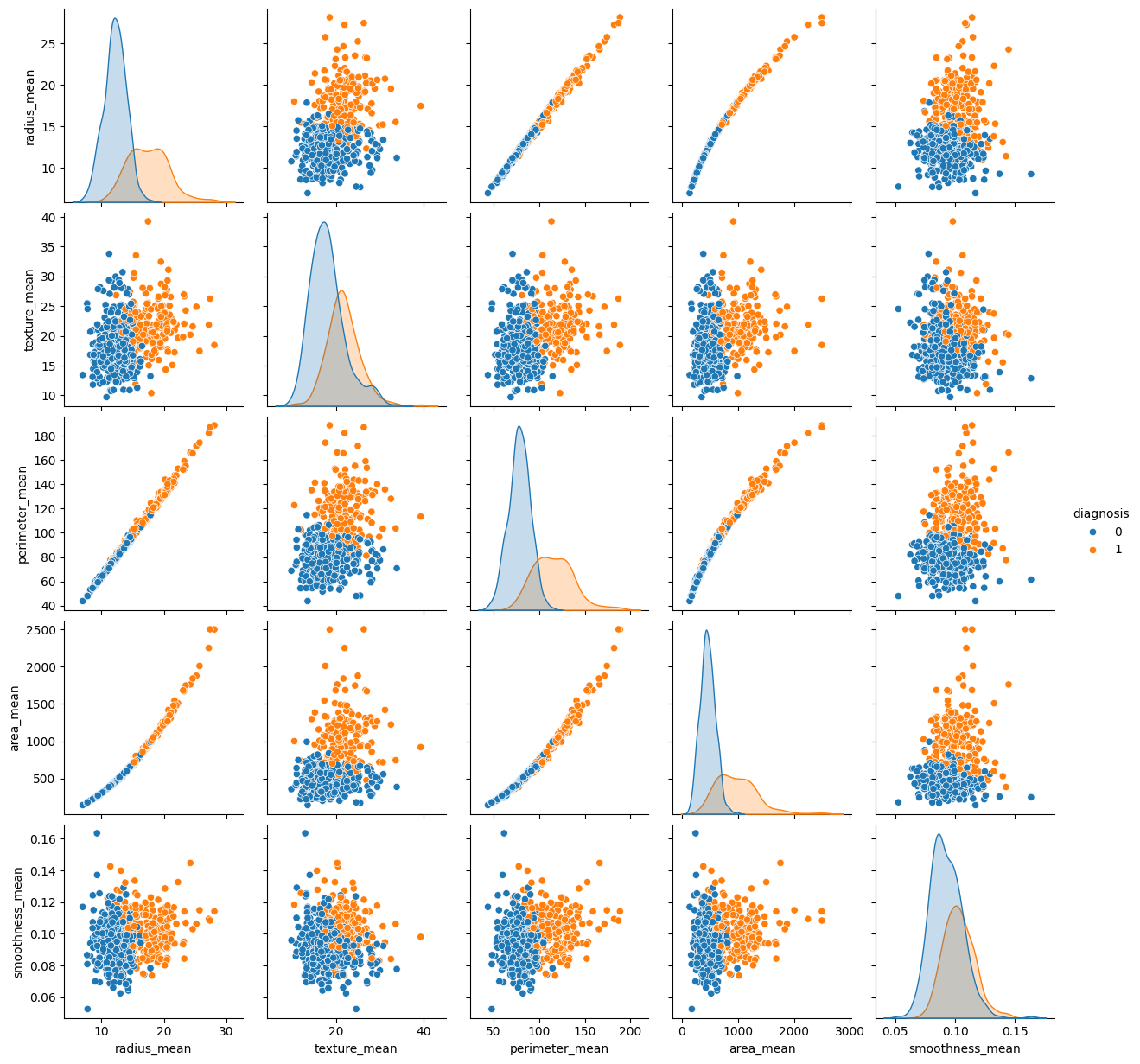


Fig. 3: Pairplot Visualization

The third image is an example of pairplot where each scatter plot shows the relationship between two features, benign representing in blue and malignant in orange. The graph clearly shows the strong positive correlation between radius\_mean, perimeter\_mean, and area\_mean. The diagonal plot shows that there is a noticeable difference in distribution benign and malignant cases and malignant cases have high higher values. So these three features are good predictors for cancer.

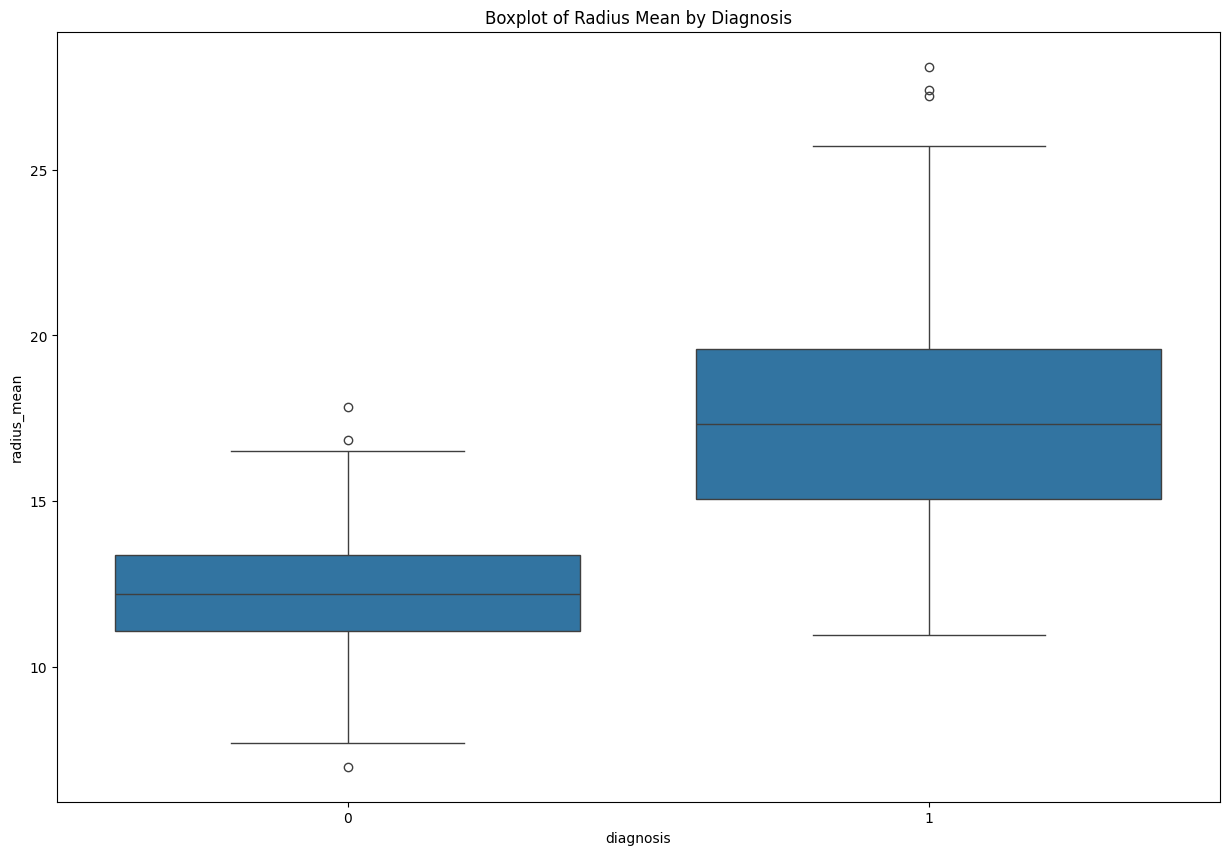


Fig. 4: Boxplot visualization

The boxplot is of radius\_mean by diagnosis and it shows the distribution of tumor radius for the cases benign (0) and malignant (1). From the boxplot it is clear that benign tumors have narrow spread with median around a lower value and the inter quartile range is also small. The median of the radius is larger for malignant tumors and the spread of values is also wider, so the larger radii are more associated with malignancy. The visualization states that the radius can be used as a distinguishing factor in the diagnosis of breast cancer.

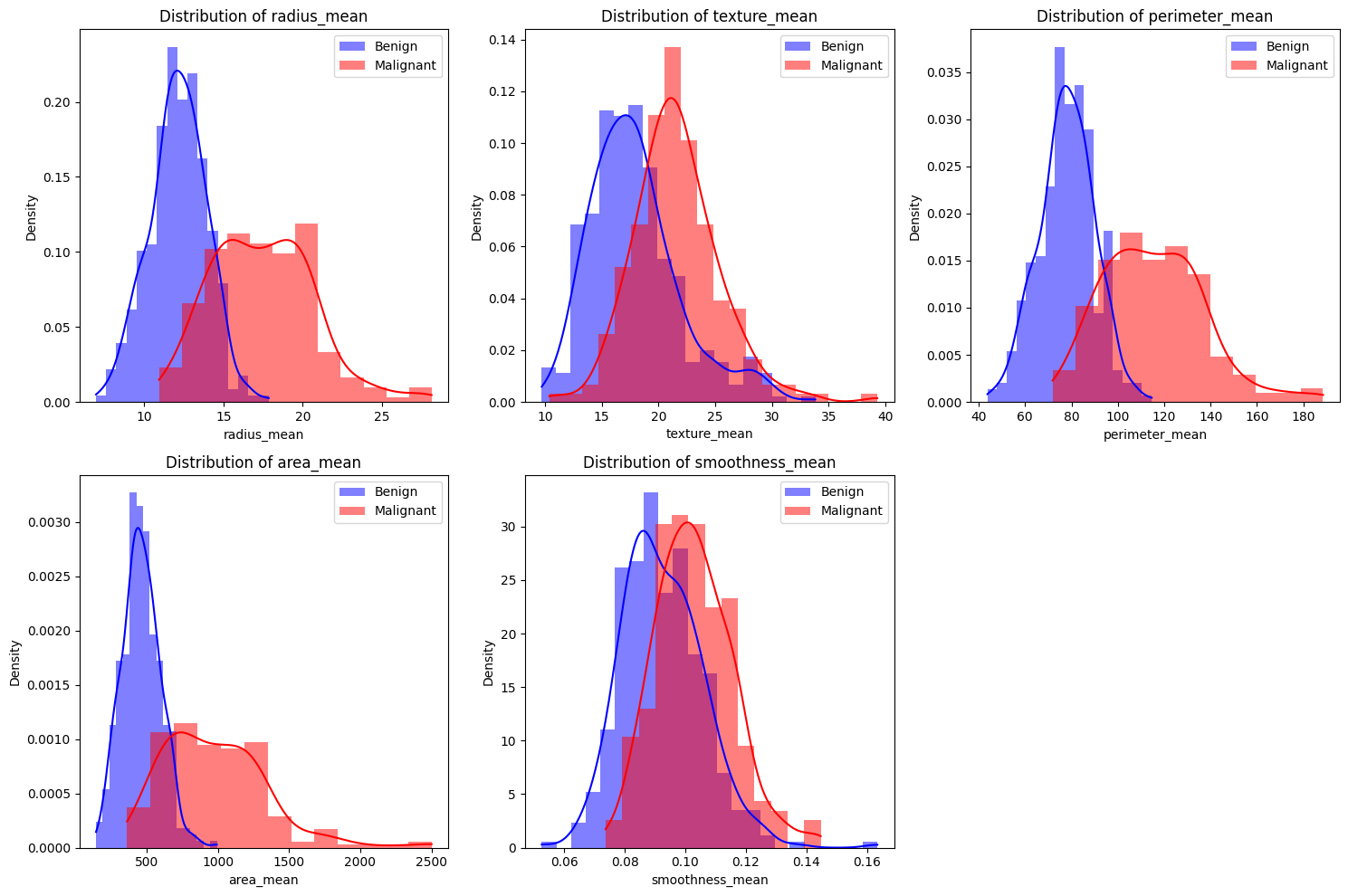


Fig. 5: Distribution of the average values of selected features (radius\_mean, texture\_mean, perimeter\_mean, area\_mean, and smoothness\_mean)

Features like radius\_mean, perimeter\_mean, area\_mean show distinct separations between both cases. This information is conveyed by these density plots depicting that for most of the features, the densities are shifted up for malignant cases than in benign cases. These characteristics seem to be useful in differentiating between benign and malignant cases, with malignant tumors typically having higher values.

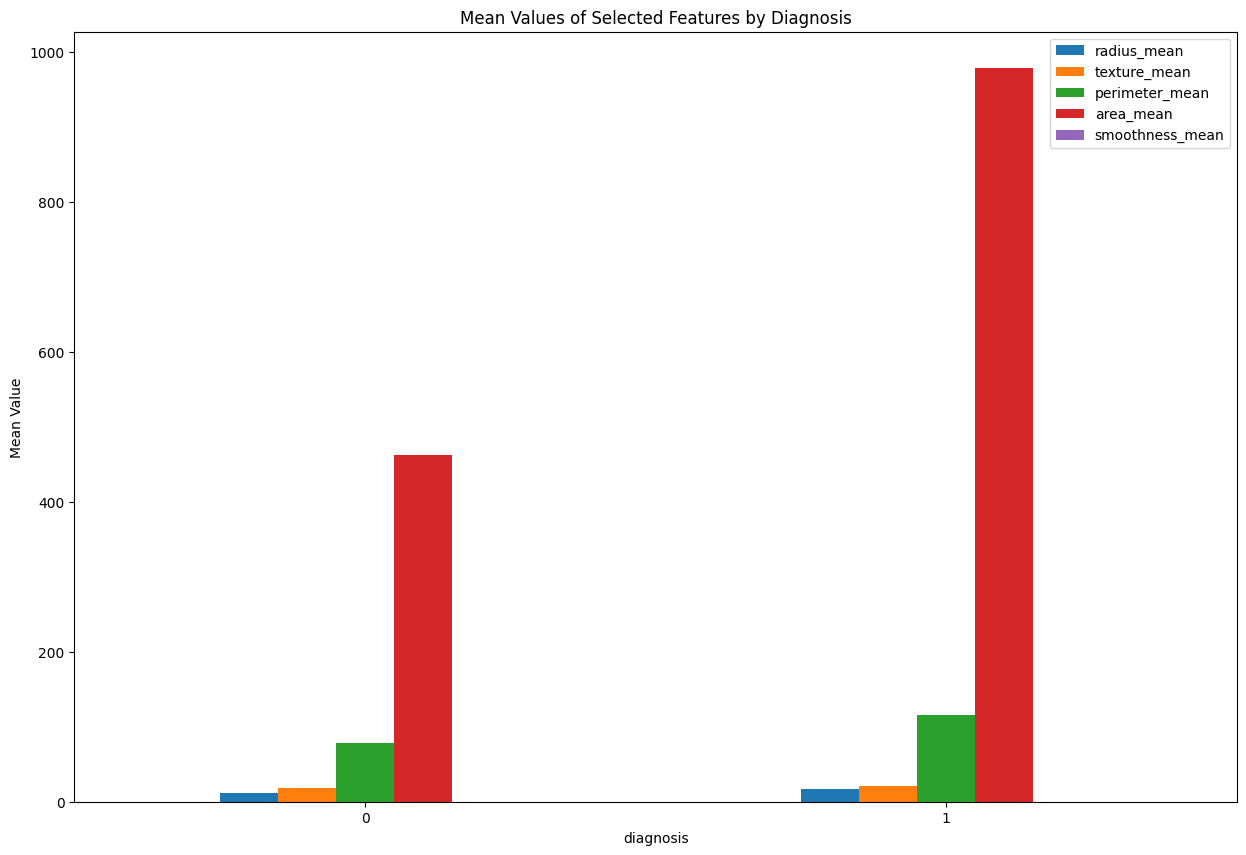


Fig. 6: Bar plot

The above graph presents the bar plot of the mean of some features with regard to cases that have benign (0) and malignant (1) disease. Its clear from the graph that area\_mean dominates the plot while compared with other features. This suggest that area\_mean coud be critical factor to distinguish malignant tumors from benign, while features like texture\_mean and smoothess\_mean contribute less individually.

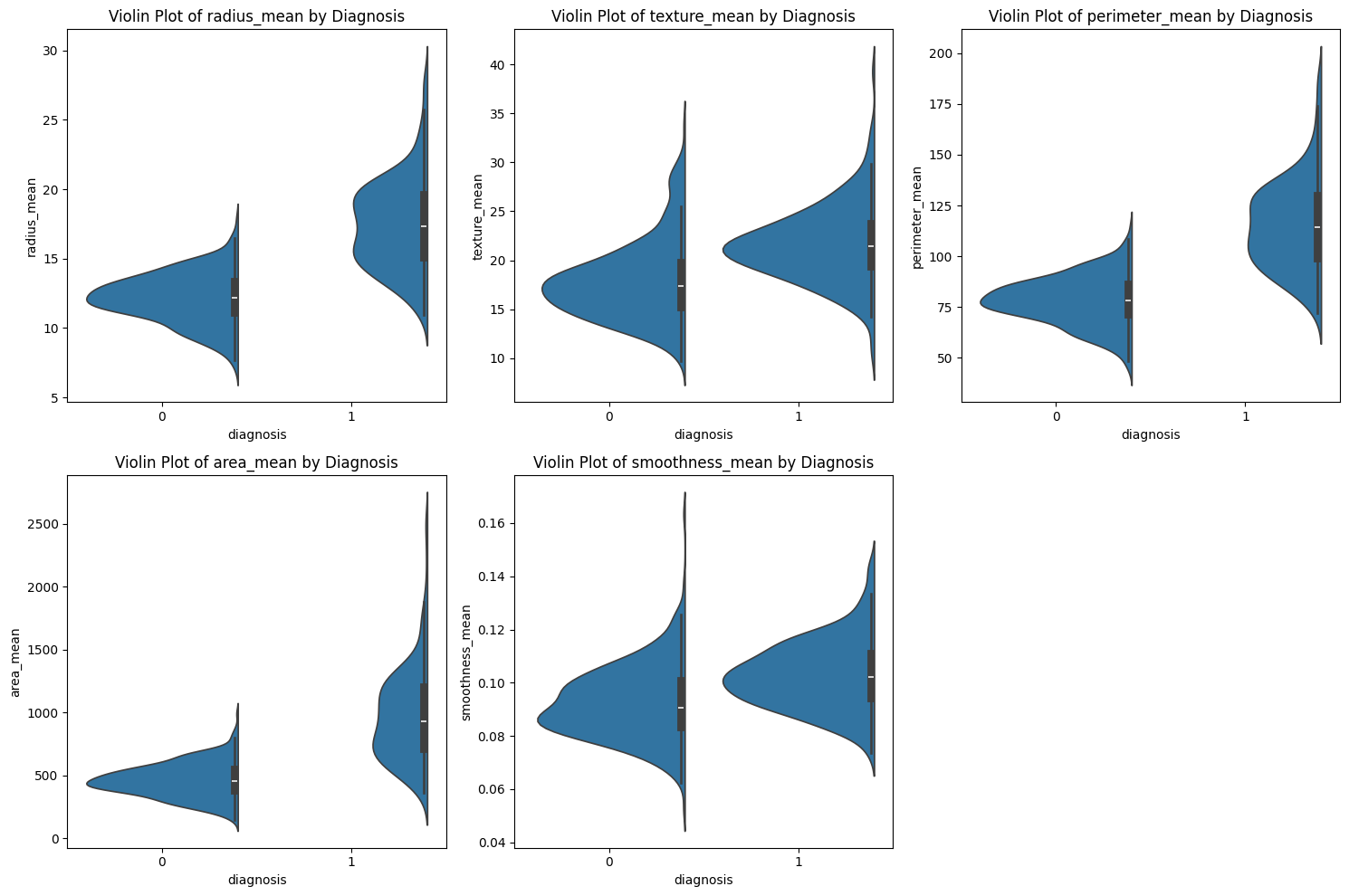


Fig. 7: Violin plot

The distribution of parameters like radius\_mean, texture\_mean, perimeter\_mean, area\_mean, and smoothness\_mean for two diagnosis results—benign or malignant—is displayed in the violin plot titled "Features by Diagnosis." This violin plot shows how all of these variables have generally larger values in malignant cases, suggesting that these characteristics may be used to discriminate between different types of breast cancer.

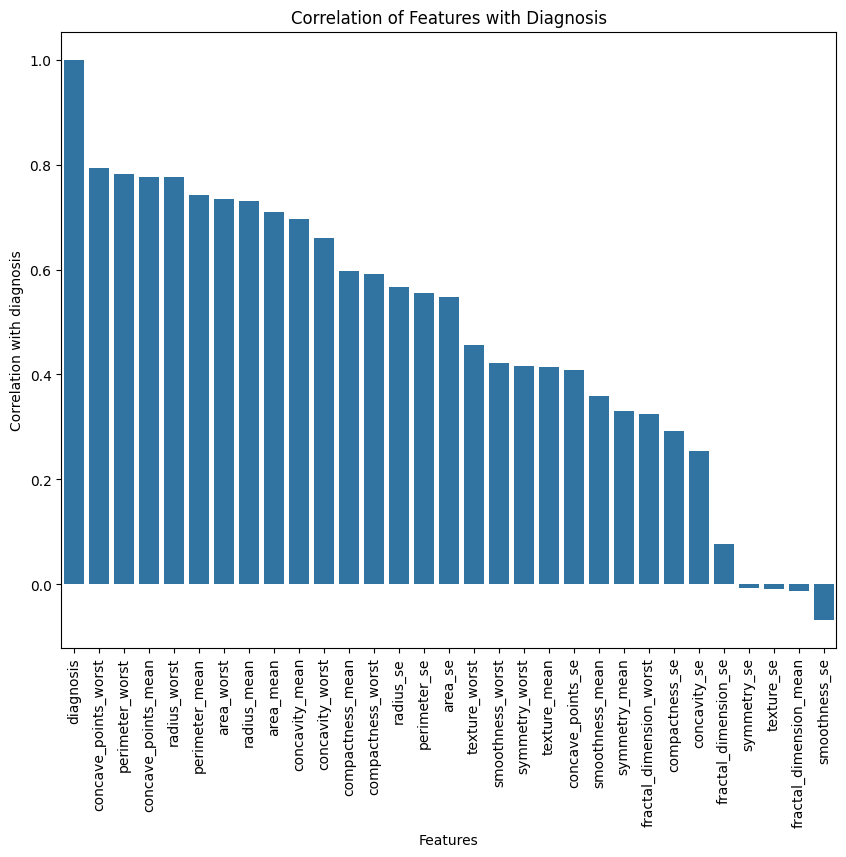


Fig. 8: Correlation bar plot

The ‘Correlation of Features with Diagnosis’ is displayed in the bar plot that signifies the amount of correlation between features and diagnosis. From the graph, its clear that ‘concave points\_worst’, ‘perimeter\_worst,’ ‘radius\_worst’ and ‘area\_worst’ shows highly positive correlation and ‘symmetry\_se’, ‘texture\_se’ and ‘smoothness\_se’ shows negative correlation against diagnosis.

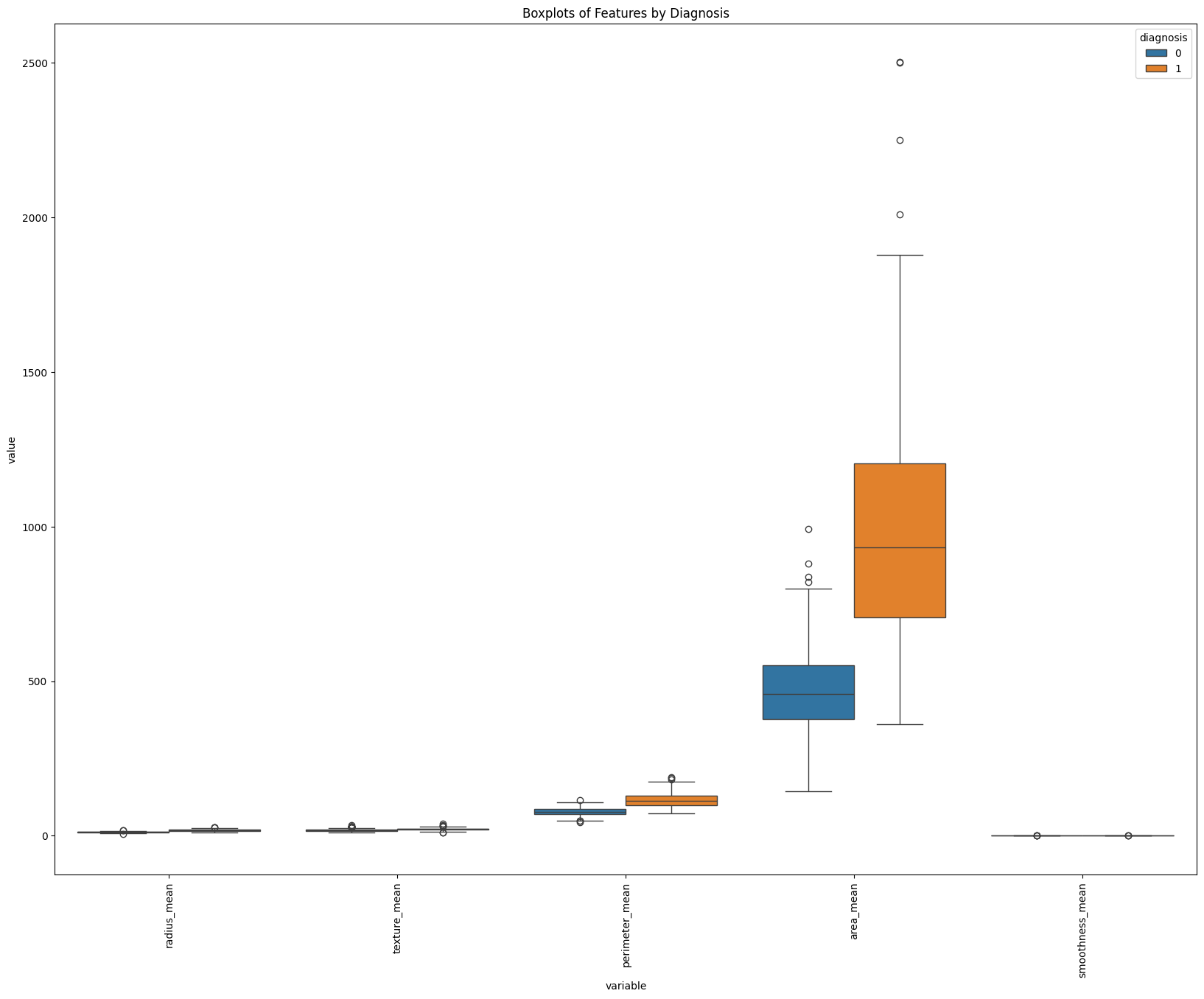


Fig. 9: Box plot

The boxplots of features by diagnosis give a clear graphical presentation of the median, dispersion as well as ranges of the features based on the diagnosis variable. For example, the `radius\_mean’ manipulate the box plot indicates that the radii of malignant tumors are much larger and have a higher variance than benign tumors, thus radii are indeed one of the most important indicators for diagnosing cancer.

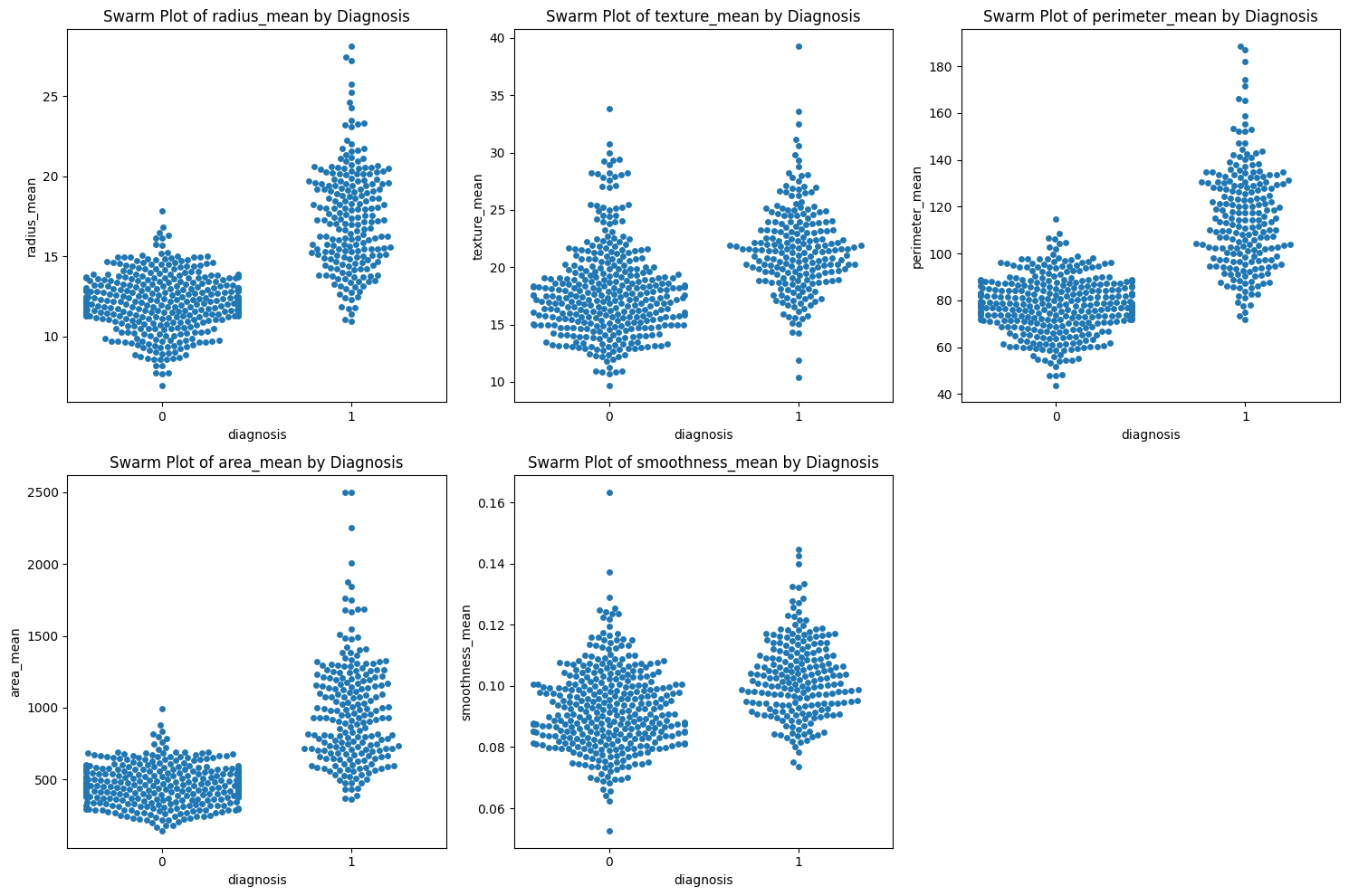


Fig. 10: Swarm plot

The pattern plot is extended by the "Swarm Plot of Features by Diagnosis," which shows the data clouds for each characteristic and the distribution and overlap of benign and malignant cases. With a focus on the feature distribution, this figure enables the comparison of the variance around the means of each diagnosis group.

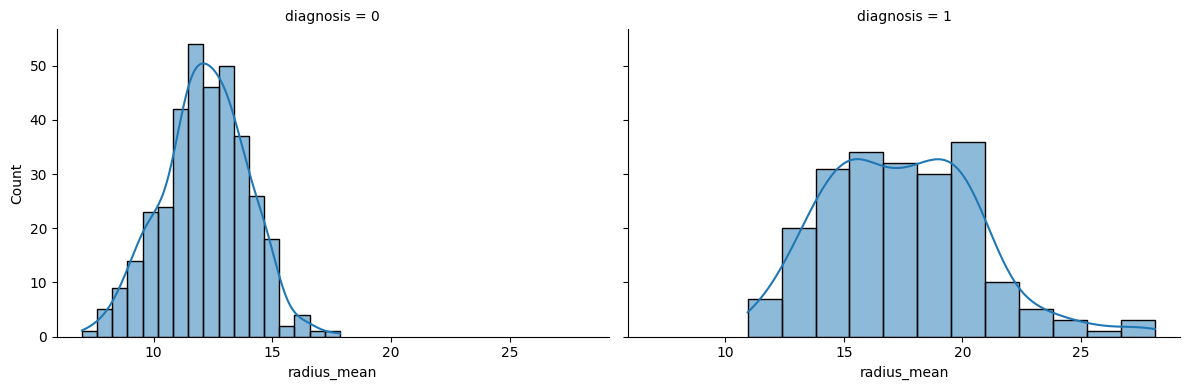


Fig. 11: Histogram plot

Benign and malignant tumors can be distinguished from one another using the radius\_mean feature in the Facet Grid display. Radius\_mean is a crucial characteristic in detecting breast cancer, as seen by the significant variation in its distribution between the two diagnoses.

# **Results**

# **Logistic Regression Results**

It is evident from the classification report and the confusion matrix that Logistic Regression yielded an accuracy of almost 98% when compared to other classifiers. It is evident that the model that is being presented has good recall, precision, and F1-scores for both the benign and malignant classes; however, the benign class has slightly higher scores, with precision of 0.99 and F1-score of 0.99. The small disparity of the performance measures that the different classes presented also indicates that Logistic Regression has good stability for this dataset.

* 1. **Random Forest Results**

Out of the algorithms used the accuracy found in the Random Forest model was found to be around 97 percent, the mean accuracy is slightly lower from the Logistic Regression. The confusion matrix likewise shows a relatively higher misclassification for the malignant class which gave four false negatives and thus the lower recall of 0.94 for this class. For all the above stated reasons the model still provides high accuracy for the malignant cases the precision being at 0 and 98%. This plot reveals that the features the model uses most are concave points\_mean, concave points\_worst and area\_worst amongst others; a significant fact which underscores the ability of the specified tumor details to classify the tumor.

* 1. **Support Vector Machine (SVM) Results**

The accuracy achieved by the SVM model varied, ranging from 97 percent to 98 percent. According to the classification report, the SVM performed well in both classes, with precision, recall, and F1-score all being very near to 0.98. In comparison to Random Forest, the model exhibits a slightly more balanced disparity between the two classes, with nearly equal recall for each class.

* 1. **KMeans Clustering**

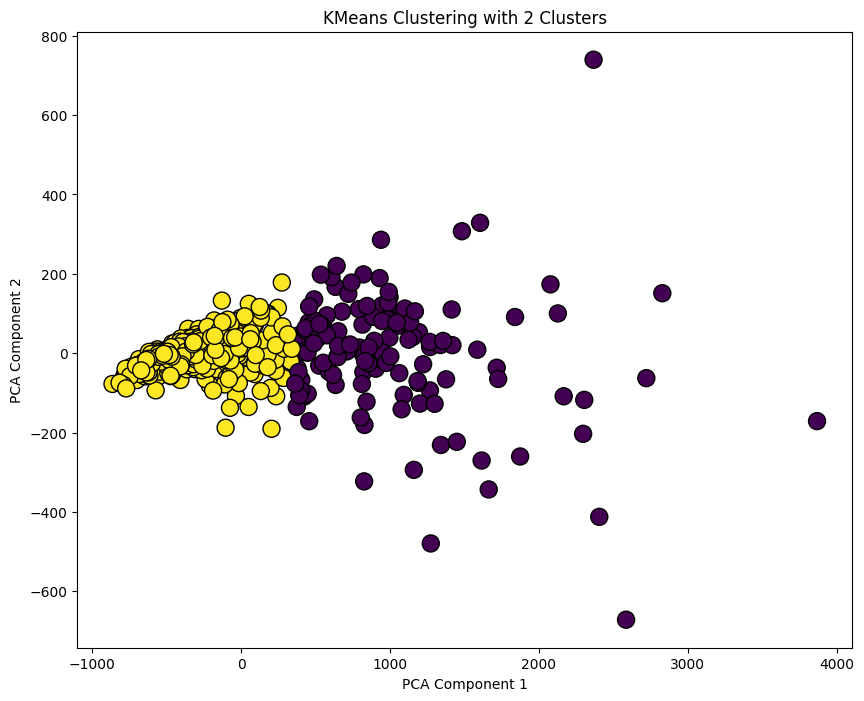
****

Fig. 12: KMeans clustering

The initial step involves selecting only the first image, from which feature vectors are created by extracting various features of sizes 1000 × 1000 and 182 x 182 for the KMeans clustering algorithm with K=2 clusters. The location of this plot is such that there is a division between the two clusters, the yellow points and the purple points. The silhouette score is approximately at 0. By reference comparison, the clustering quality is quite decent at around 697, suggesting that the clusters are well reconstructed.

In terms of "real" diagnosis labels, the cross-tabulation at the bottom demonstrates that the clustering is somewhat accurate. The purple cluster, for example, clearly has more benign cases (labeled as 0) than malignant ones, but the other clusters do not significantly deviate from the true 0/1 distribution. The accuracy that KMeans Clustering produced overall was about 85% of 4%, indicating that the clusters that the model produced are fairly near to the actual labels. This makes it easy to conclude that the model known as KMeans was somewhat successful in the clustering of similar data points.

* 1. **DBSCAN Clustering**

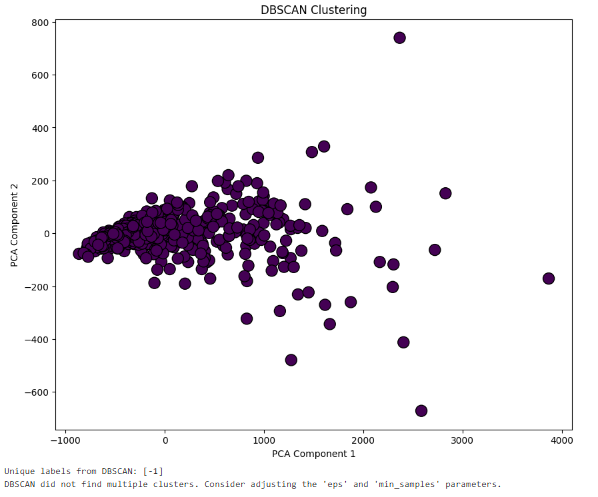


Fig. 13: DBSCAN clustering

The application of the DBSCAN clustering management technique is shown in the above image. The uniform color of the points indicates that DBSCAN was unable to distinguish distinct clusters and instead combined all of the data into a single cluster. The algorithm's inability to find additional clusters would be indicated at the bottom of the message, possibly as a result of the epsilon and min\_samples values. It suggests that the current setting of DBSCAN is not appropriate for this particular data or the data contains no density-based clusters in the PCA reduce space.

As a result of DBSCAN returning more than two clusters, performance metrics like accuracy cannot be calculated. This result also demonstrates that DBSCAN, with the selected parameters, is not suitable for density-based clusters in complicated datasets that might not have density-based clusters or where finding such clusters might require specific parameter settings.

* 1. **Gaussian Mixture Model (GMM) Clustering**

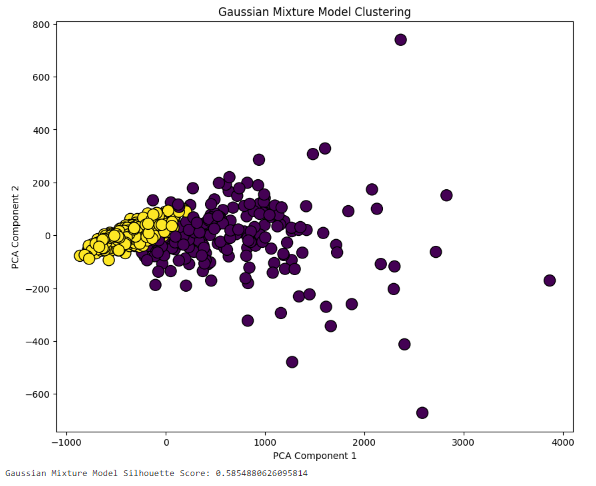


Fig. 14: GMM clustering

The above image depicts a GMM clustering of the vector, which was successful in clustering relevant images. Similarly to KMeans, GMM assigned two clusters marked with yellow and purple points. Yet, the clusters are not as well separated as in KMeans, which is traceable to the silhouette score of 0. 585. This means that GMM can recognize the structures of cluster while the separation is not very clear as there is some overlapping of clusters especially in the PCA space.

While the accuracy provided by Random Forest was approximately 90%, the accuracy provided by K Nearest Neighbor when combined with the Gaussian Mixture Model (GMM) was approximately 93%. 3.0%, outperforming KMeans. This may indicate that GMM was better at capturing the underlying distribution of the data because of the ability of the method in dealing with overlapping clusters and relationship in the data.

### **Conclusion**

From the exploratory data analysis it’s evident that features like ‘radius\_mean’, ‘perimeter\_mean’, ‘area\_mean’, ‘concave\_points\_worst’ and ‘concave\_points\_mean’ are some of the best features in identifying breast cancer. And features like ‘smoothness\_mean’ and ‘texture\_mean’ are less effective in distinguishing between benign and malignant tumors.

It was found that SVM, Random Forest, and Logistic Regression could all accurately diagnose benign and malignant tumors; however, Logistic Regression outperformed the other two by a narrow margin. The high levels of precision, recall, and F1-score on Logistic Regression for both classes underlines the capability and reliability for this dataset. While Random Forest has a moderate sensitivity, it performs better in terms of feature importance, some of which are important to distinguish benign tumors from malignant ones, for instance, concave points\_mean and area\_worst. SVM can be considered a reliable alternative to Logistic Regression because it outperformed it in both classes and on the complicated data structure.

Compared to the supervised models, the unsupervised models produced a far more distinct set of results, with several of them even outperforming the supervised models. With an accuracy of around 85%, KMeans clustering produced decent clusters of similar data, according to its results. And the KMeans silhouette score showed that while the clusters were not entirely distinct from one another, they were reasonably well defined. However, DBSCAN did not perform well, grouping all the points into a single cluster and failing to find clusters with the current dataset, indicating that the current setup has the disadvantage of not detecting clusters well.

The Gaussian Mixture Model (GMM) outperforms the KMeans, providing approximately 93% more accurate results on average. When compared with the supervised models results from the suggested model were still lower. GMM could identify overlapping clusters and could give a probability distribution of the data, but unlike KMeans, the clusters were not easily distinguished from one another.

But, as the above experiment makes clear, supervised models are far more effective and precise than unsupervised models for this classification task in a typical case involving the provided dataset. Logistic Regression and SVM provide good and comparable accuracy, and Random Forest provides additional interpretability using features importance. Although the results of the clustering are acceptable, the accuracy of KMeans and GMM is much lower than that of supervised methods, and DBSCAN is unable to detect the clusters. As a result, it may not be suitable until further parameter adjustments are performed on this dataset.

In conclusion, although nonlinear analysis based on unsupervised learning such as KMeans and GMM gives a better picture of the distribution of the data available, classification of breast cancer efficiently happens with supervised models like Logistic Regression and SVM, and so is preferred in this case. The choice amongst these models can be more precise if the application has certain requirements, such as the need for the model to be interpretable or the presence of non-linear interactions.

## **Future Work**

In additional research, the focus can be more on complex deep learning models, which may result in lower sensitivity to breast cancer and higher detection accuracy. Including larger and richer datasets, like genomics data, may also enhance the model's capacity to predict cancer risks. Variability in the model's performance due to different feature selection techniques may also be evaluated to enhance the effectiveness of the selection of crucial factors related to the diagnosis. Unfortunately, the next steps in the actual application of these models in clinical practice and evaluations of their value in supporting healthcare providers have not been addressed.

# **Legal, Ethical and Professional Issues**

Several legal, ethical, and professional issues are addressed in this study that focuses on the use of machine learning in breast cancer diagnosis. Sensitive patient information is contained in the data, thus confidentiality and privacy are crucial. Ensuring the confidentiality of patient data and managing it appropriately is a critical concern. Another area of concern is bias and fairness; a number of checks must be made on the models utilized in the algorithms to guarantee that the outcomes will not misdiagnose or treat certain patient categories unfairly. Also, the responsibility that belongs to artificial intelligence in terms of decisions made in the sphere of medicine is a major problem. Clinicians I’d always have to remain in the loop in decision making because relying on the machine learning algorithm without oversight could be catastrophic. Last but not least, these models must be transparent to medical practitioners and can be evaluated for medical expertise, suitability and clarity of the models' decision-making process is essential to winning over patients' and healthcare providers' attention.

# **References**

Aditya .P (2018) L1 and L2 regularization, Medium. Available at: <https://medium.com/@aditya97p/l1-and-l2-regularization-237438a9caa6> (Accessed: August 22, 2024).

Banoula, M. (2020) An introduction to logistic regression in machine learning, Simplilearn.com. Simplilearn. Available at:

<https://www.simplilearn.com/tutorials/machine-learning-tutorial/logistic-regression-in-python> (Accessed: August 22, 2024).

Barth, S. (2024) Machine learning in healthcare: Guide to applications & benefits, ForeSee Medical. Available at:

[https://www.foreseemed.com/blog/machine-learning-in-healthcare](https://www.foreseemed.com/blog/machine-learning-in-healthcare%20) (Accessed: June 14, 2024).

Breast cancer Wisconsin (diagnostic) dataset (2024) GeeksforGeeks. Available at: [https://www.geeksforgeeks.org/breast-cancer-wisconsin-diagnostic-dataset/](https://www.geeksforgeeks.org/breast-cancer-wisconsin-diagnostic-dataset/%20) (Accessed: June 14, 2024).

Devi, R.D.H. and Deepika, P., 2015, December. Performance comparison of various clustering techniques for diagnosis of breast cancer. In 2015 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC) (pp. 1-5). IEEE.

Dhiraj, K. (2019) Top 4 advantages and disadvantages of Support Vector Machine or SVM, Medium. Available at:

[https://dhirajkumarblog.medium.com/top-4-advantages-and-disadvantages-of-support-vector-machine-or-svm-a3c06a2b107](https://dhirajkumarblog.medium.com/top-4-advantages-and-disadvantages-of-support-vector-machine-or-svm-a3c06a2b107%20) (Accessed: August 22, 2024).

Education Ecosystem (LEDU) (2018) Understanding K-means clustering in machine learning, Towards Data Science. Available at:

[https://towardsdatascience.com/understanding-k-means-clustering-in-machine-learning-6a6e67336aa1](https://towardsdatascience.com/understanding-k-means-clustering-in-machine-learning-6a6e67336aa1%20) (Accessed: August 22, 2024).

Karanam, S. (2022) Exploratory data analysis — breast cancer Wisconsin (diagnostic) dataset, Medium. Available at:

[https://medium.com/@shashmikaranam/exploratory-data-analysis-breast-cancer-wisconsin-diagnostic-dataset-6a3be9525cd](https://medium.com/@shashmikaranam/exploratory-data-analysis-breast-cancer-wisconsin-diagnostic-dataset-6a3be9525cd%20) (Accessed: June 14, 2024).

Koli, S. (2023) How to evaluate the performance of clustering algorithms using silhouette coefficient, Medium. Available at:

[https://medium.com/@MrBam44/how-to-evaluate-the-performance-of-clustering-algorithms-3ba29cad8c03](%20https:/medium.com/@MrBam44/how-to-evaluate-the-performance-of-clustering-algorithms-3ba29cad8c03%20) (Accessed: August 22, 2024).

Marne, S., Churi, S. and Marne, M., 2020, March. Predicting breast cancer using effective classification with decision tree and k means clustering technique. In 2020 International Conference on Emerging Smart Computing and Informatics (ESCI) (pp. 39-42). IEEE.

Premanand, S. (2022) Principal Component Analysis in Machine Learning, Analytics Vidhya. Available at:

<https://www.analyticsvidhya.com/blog/2022/07/principal-component-analysis-beginner-friendly/> (Accessed: August 22, 2024).

Radha, R. and Rajendiran, P., 2014, February. Using K-means clustering technique to study of breast cancer. In 2014 World Congress on Computing and Communication Technologies (pp. 211-214). IEEE.

Satpati, S. (2023) Clustering by DBSCAN (density-based spatial clustering of applications with noise) clearly explained with coding in python, Medium. Available at: [https://medium.com/@satpatishrimanta/clustering-by-dbscan-density-based-spatial-clustering-of-applications-with-noise-clearly-f93c5c72f706](https://medium.com/@satpatishrimanta/clustering-by-dbscan-density-based-spatial-clustering-of-applications-with-noise-clearly-f93c5c72f706%20) (Accessed: August 22, 2024).

Shin, T. (2023) Understanding feature importance in machine learning, Built In. Available at: [https://builtin.com/data-science/feature-importance](https://builtin.com/data-science/feature-importance%20) (Accessed: August 22, 2024).

Simplilearn (2020) What is a Confusion Matrix in Machine Learning?, Simplilearn.com. Simplilearn. Available at:

<https://www.simplilearn.com/tutorials/machine-learning-tutorial/confusion-matrix-machine-learning> (Accessed: August 22, 2024).

Singh, A. (2019) Build better and accurate clusters with Gaussian mixture models, Analytics Vidhya. Available at:

[https://www.analyticsvidhya.com/blog/2019/10/gaussian-mixture-models-clustering/ (](https://www.analyticsvidhya.com/blog/2019/10/gaussian-mixture-models-clustering/%20()Accessed: August 22, 2024).

Sruthi, E. R. (2021) Understanding Random Forest algorithm with examples, Analytics Vidhya. Available at:

[https://www.analyticsvidhya.com/blog/2021/06/understanding-random-forest/](https://www.analyticsvidhya.com/blog/2021/06/understanding-random-forest/%20) (Accessed: August 22, 2024).

Tanner, G. (2021) Random forest, Machine Learning Explained. Available at:

[https://ml-explained.com/blog/random-forest-explained](https://ml-explained.com/blog/random-forest-explained%20) (Accessed: August 22, 2024).

# 

# **Appendices**

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

import numpy as np

from google.colab import drive

drive.mount('/content/drive')

df = pd.read\_csv('/content/drive/My Drive/data.csv')

df.head()

# Display summary statistics for numerical columns

df.describe()

df.info()

# Drop the'id' columns as it is not needed

df = df.drop(columns=['id'])

# Check for missing values

missing\_values = df.isnull().sum()

print(missing\_values[missing\_values > 0])

# Encode the 'diagnosis' column: M = 1 (Malignant), B = 0 (Benign)

df['diagnosis'] = df['diagnosis'].map({'M': 1, 'B': 0})

# Summary statistics

print(df.describe())

# Count of diagnosis

diagnosis\_counts = df['diagnosis'].value\_counts()

print(diagnosis\_counts)

# Plot the distribution of diagnosis

sns.countplot(x='diagnosis', data=df)

plt.title('Distribution of Diagnosis')

plt.show()

# Calculate the correlation matrix

corr\_matrix = df.corr()

# Plot the heatmap

plt.figure(figsize=(20, 20))

sns.heatmap(corr\_matrix, annot=True, fmt='.2f', cmap='coolwarm')

plt.title('Correlation Matrix')

plt.show()

# Select a subset of features for pairplot

selected\_features = ['radius\_mean', 'texture\_mean', 'perimeter\_mean', 'area\_mean', 'smoothness\_mean', 'diagnosis']

# Plot pairplot

sns.pairplot(df[selected\_features], hue='diagnosis', diag\_kind='kde')

plt.show()

# Plot boxplot for selected features

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

sns.boxplot(x='diagnosis', y='radius\_mean', data=df)

plt.title('Boxplot of Radius Mean by Diagnosis')

plt.show()

# Plot distribution for selected features

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

for i, feature in enumerate(selected\_features[:-1]):

    plt.subplot(2, 3, i+1)

    sns.histplot(df[df['diagnosis'] == 0][feature], color='blue', label='Benign', kde=True, stat="density", linewidth=0)

    sns.histplot(df[df['diagnosis'] == 1][feature], color='red', label='Malignant', kde=True, stat="density", linewidth=0)

    plt.title(f'Distribution of {feature}')

    plt.legend()

plt.tight\_layout()

plt.show()

# Group by diagnosis and calculate mean values for each feature

grouped\_means = df.groupby('diagnosis').mean()

# Plot the mean values for selected features

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

grouped\_means[selected\_features[:-1]].plot(kind='bar', figsize=(15, 10))

plt.title('Mean Values of Selected Features by Diagnosis')

plt.ylabel('Mean Value')

plt.xticks(rotation=0)

plt.show()

# Plot violin plots for selected features

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

for i, feature in enumerate(selected\_features[:-1]):

    plt.subplot(2, 3, i+1)

    sns.violinplot(x='diagnosis', y=feature, data=df, split=True)

    plt.title(f'Violin Plot of {feature} by Diagnosis')

plt.tight\_layout()

plt.show()

# Calculate correlation with the diagnosis

corr\_with\_diagnosis = corr\_matrix['diagnosis'].sort\_values(ascending=False)

# Plot the correlations

plt.figure(figsize=(10, 8))

sns.barplot(x=corr\_with\_diagnosis.index, y=corr\_with\_diagnosis.values)

plt.title('Correlation of Features with Diagnosis')

plt.xticks(rotation=90)

plt.xlabel('Features')

plt.ylabel('Correlation with diagnosis')

plt.show()

# Boxplots for features

plt.figure(figsize=(20, 15))

df\_melted = pd.melt(df, id\_vars='diagnosis', value\_vars=selected\_features[:-1])

sns.boxplot(x='variable', y='value', hue='diagnosis', data=df\_melted)

plt.title('Boxplots of Features by Diagnosis')

plt.xticks(rotation=90)

plt.show()

# Swarm plots for selected features

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

for i, feature in enumerate(selected\_features[:-1]):

    plt.subplot(2, 3, i+1)

    sns.swarmplot(x='diagnosis', y=feature, data=df)

    plt.title(f'Swarm Plot of {feature} by Diagnosis')

plt.tight\_layout()

plt.show()

# FacetGrid for selected features

g = sns.FacetGrid(df, col="diagnosis", height=4, aspect=1.5)

g.map(sns.histplot, "radius\_mean", kde=True)

plt.show()

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

from sklearn.preprocessing import StandardScaler

from sklearn.linear\_model import LogisticRegression

from sklearn.ensemble import RandomForestClassifier

from sklearn.svm import SVC

from sklearn.metrics import classification\_report, confusion\_matrix, accuracy\_score

from sklearn.decomposition import PCA

from sklearn.cluster import KMeans

from sklearn.metrics import silhouette\_score

import matplotlib.pyplot as plt

# Splitting the data into training and testing sets

X = df.drop('diagnosis', axis=1)

y = df['diagnosis']

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.3, random\_state=42)

# Standardizing the features

scaler = StandardScaler()

X\_train = scaler.fit\_transform(X\_train)

X\_test = scaler.transform(X\_test)

# Logistic Regression

logreg = LogisticRegression()

logreg.fit(X\_train, y\_train)

y\_pred\_logreg = logreg.predict(X\_test)

print("Logistic Regression:")

print(confusion\_matrix(y\_test, y\_pred\_logreg))

print(classification\_report(y\_test, y\_pred\_logreg))

print(f"Accuracy: {accuracy\_score(y\_test, y\_pred\_logreg)}")

# Random Forest

rf = RandomForestClassifier(n\_estimators=100, random\_state=42)

rf.fit(X\_train, y\_train)

y\_pred\_rf = rf.predict(X\_test)

print("Random Forest:")

print(confusion\_matrix(y\_test, y\_pred\_rf))

print(classification\_report(y\_test, y\_pred\_rf))

print(f"Accuracy: {accuracy\_score(y\_test, y\_pred\_rf)}")

# Feature Importance from Random Forest

importances = rf.feature\_importances\_

indices = np.argsort(importances)[::-1]

features = X.columns

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

plt.title("Feature Importance")

plt.bar(range(X.shape[1]), importances[indices], align="center")

plt.xticks(range(X.shape[1]), [features[i] for i in indices], rotation=90)

plt.show()

# Support Vector Machine

svc = SVC(kernel='linear')

svc.fit(X\_train, y\_train)

y\_pred\_svc = svc.predict(X\_test)

print("Support Vector Machine:")

print(confusion\_matrix(y\_test, y\_pred\_svc))

print(classification\_report(y\_test, y\_pred\_svc))

print(f"Accuracy: {accuracy\_score(y\_test, y\_pred\_svc)}")

# PCA for Unsupervised Learning

pca = PCA(n\_components=2)

X\_pca = pca.fit\_transform(X)

# KMeans Clustering

kmeans = KMeans(n\_clusters=2, random\_state=42)

kmeans.fit(X\_pca)

labels = kmeans.labels\_

# Plotting the clusters

plt.figure(figsize=(10, 8))

plt.scatter(X\_pca[:, 0], X\_pca[:, 1], c=labels, cmap='viridis', edgecolor='k', s=150)

plt.title('KMeans Clustering with 2 Clusters')

plt.xlabel('PCA Component 1')

plt.ylabel('PCA Component 2')

plt.show()

# Evaluating the clustering

silhouette\_avg = silhouette\_score(X, labels)

print(f'Silhouette Score: {silhouette\_avg}')

# Checking cluster associations with the diagnosis

df['cluster'] = labels

cluster\_vs\_diagnosis = pd.crosstab(df['cluster'], df['diagnosis'])

print(cluster\_vs\_diagnosis)

from sklearn.cluster import DBSCAN, AgglomerativeClustering

from sklearn.metrics import silhouette\_score

import matplotlib.pyplot as plt

# DBSCAN Clustering

dbscan = DBSCAN(eps=0.01, min\_samples=3)

dbscan\_labels = dbscan.fit\_predict(X\_pca)

# Plotting DBSCAN clusters

plt.figure(figsize=(10, 8))

plt.scatter(X\_pca[:, 0], X\_pca[:, 1], c=dbscan\_labels, cmap='viridis', edgecolor='k', s=150)

plt.title('DBSCAN Clustering')

plt.xlabel('PCA Component 1')

plt.ylabel('PCA Component 2')

plt.show()

# Trying different parameters for DBSCAN

dbscan = DBSCAN(eps=1.5, min\_samples=10)  # Adjusted parameters

dbscan\_labels = dbscan.fit\_predict(X\_pca)

# Check the unique labels

print("Unique labels from DBSCAN:", np.unique(dbscan\_labels))

# Plotting DBSCAN clusters if multiple clusters are found

if len(np.unique(dbscan\_labels)) > 1:

    plt.figure(figsize=(10, 8))

    plt.scatter(X\_pca[:, 0], X\_pca[:, 1], c=dbscan\_labels, cmap='viridis', edgecolor='k', s=150)

    plt.title('DBSCAN Clustering')

    plt.xlabel('PCA Component 1')

    plt.ylabel('PCA Component 2')

    plt.show()

    # Evaluating DBSCAN clustering

    dbscan\_silhouette = silhouette\_score(X, dbscan\_labels)

    print(f'DBSCAN Silhouette Score: {dbscan\_silhouette}')

else:

    print("DBSCAN did not find multiple clusters. Consider adjusting the 'eps' and 'min\_samples' parameters.")

from sklearn.mixture import GaussianMixture

from sklearn.metrics import silhouette\_score

# Gaussian Mixture Model Clustering

gmm = GaussianMixture(n\_components=2, random\_state=42)

gmm\_labels = gmm.fit\_predict(X\_pca)

# Plotting GMM clusters

plt.figure(figsize=(10, 8))

plt.scatter(X\_pca[:, 0], X\_pca[:, 1], c=gmm\_labels, cmap='viridis', edgecolor='k', s=150)

plt.title('Gaussian Mixture Model Clustering')

plt.xlabel('PCA Component 1')

plt.ylabel('PCA Component 2')

plt.show()

# Evaluating GMM clustering

gmm\_silhouette = silhouette\_score(X, gmm\_labels)

print(f'Gaussian Mixture Model Silhouette Score: {gmm\_silhouette}')