Enhancing Breast Tumor Diagnosis: A Comprehensive Analysis

Data Science II: Deliverable 1

COSC 4337

Submitted to

Dr. Ricardo Vilalta

Ly Ha 1920058

Erica Hay 2049545

Khuong Ngo 1857210

The diagnosis of tumors has traditionally been performed by a full biopsy, an invasive surgical procedure. Other procedures, such as fine needle aspirations, provide less intrusive methods to examine the tumor tissue. However, the results from these procedures are not always met with success. By examining the characteristics of cells and contextual features, there may be a way to increase the correctness of the diagnosis process by figuring out which features are correlated with malignancy.

Thus, research was conducted at the University of Wisconsin in 1992, jointly in the departments of Computer Science and Surgery to find such a correlation. The primary objective of this study was to diagnose breast tumors using interactive image processing techniques and linear programming-based inductive classification. A total of 569 images were analyzed during the research. Various combinations of features were tested to determine the most effective in distinguishing between benign and malignant tumor samples. The, now named, Breast Cancer Wisconsin data set can be found and is sourced from the **UC Irvine Machine Learning Repository**.

When taking a closer look at the data set, we have twelve main features. The data set is introduced by two attribute information features: ‘id’ and ‘diagnosis’. Then, ten real-valued features are computed for each cell nucleus. Below is a table that demonstrates each column, their data type, and its description:

Column	Data Type	Description
ID	int	Unique identification number for each cell nucleus
Diagnosis	str	Malignant (M) or Benign (B) tumor diagnosis
Radius	float	Mean distance from the center to points on the perimeter
Texture	float	Standard deviation of gray-scale values
Perimeter	float	Perimeter of the cell nucleus
Area	float	Area of the cell nucleus

Smoothness	float	Local variation in radius lengths
Compactness	float	$Perimeter^2 \div area - 1.0$
Concavity	float	Severity of concave portions of the contour
Symmetry	float	Symmetry of the cell nucleus
Fractal Dimension	float	$coastline\ approximation - 1$

Furthermore, there are 30 columns of the real-valued features due to the addition of ‘mean’, ‘standard error’, and ‘worst’ (mean of the three largest values) variables for each of the features listed in the table above. Meaning, each cell has a column for each of the ten features times the three variables. I.e. ‘radius_mean’, ‘radius_se’, and ‘radius_worst’.

To refine our dataset and gain deeper insights, a pivotal first step involves engaging in both data cleaning and exploratory data analysis. Key steps in data cleaning, such as handling missing values, addressing duplicates and outliers, standardizing data, correcting typos and inconsistencies, dealing with data formats, addressing categorical variables, ensuring data integrity, handling data skewness, and validating data quality, collectively enhance the overall quality, accuracy, and reliability of our data. By conducting exploratory data analysis, we can systematically uncover patterns, relationships, and trends within the refined dataset. This approach contributes to a more comprehensive understanding, fostering informed decision-making and laying a solid foundation for subsequent analyses or modeling methods.

In executing the essential steps of data cleaning, we started by generating a bar chart **(Figure 3)** to visualize the distribution of values across each variable. This graphical representation provided us with an immediate insight into any missing values within the dataset. Upon careful examination of the chart, a notable observation emerged: all variables, with the exception of the one labeled ‘Unnamed: 32’, exhibited a consistent count of 569 values. However, the variable ‘Unnamed: 32’ contained no values, rendering it redundant for our analysis. Consequently, we have opted to eliminate this unnecessary variable from the dataset. To address the issue of duplicates, we employed the built-in “duplicate” function, which revealed the absence of any duplicate entries. Further refinement was accomplished through the utilization of the “head” function **(Figure 4)** to display the initial five observations in our dataset. This enabled us to review the values of each variable for any inconsistencies or typos. During this inspection, we noted the existence of an ‘id’ variable, which we determined unnecessary for further analysis, prompting its removal. Furthermore, our examination through the “head” function revealed that the ‘diagnosis’ variable comprised two distinct categorical values: ‘M’ and ‘B’, as stated earlier on. To ensure consistency in our dataset, we opted to replace the ‘M’ values with 1 and the ‘B’ values with 0, facilitating a standardized approach to our subsequent analyses.

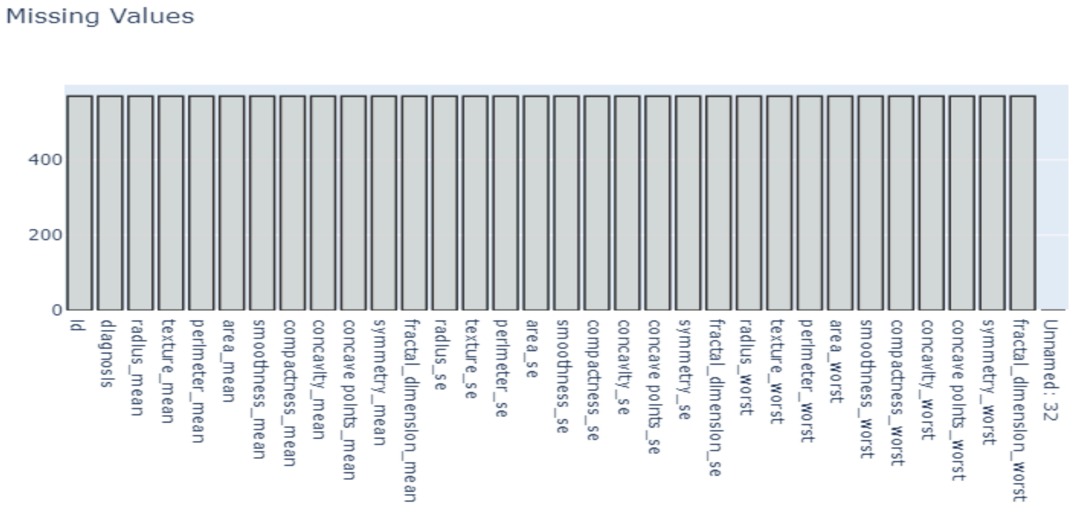
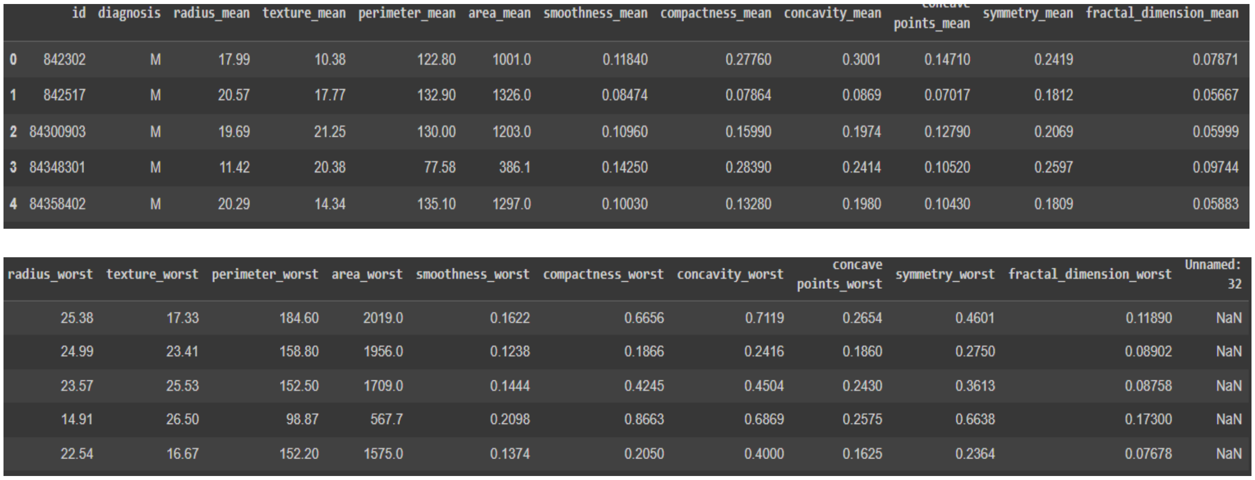


Figure 3: Bar plot displaying the distribution of values to identify missing values.

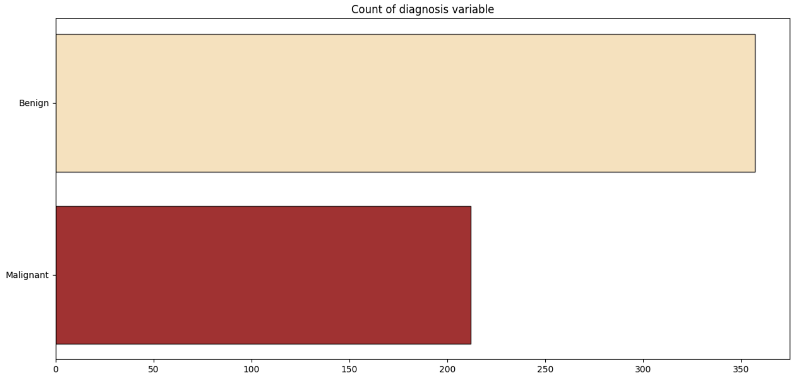


	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	symmetry_mean	fractal_dimension_mean
0	842302	M	17.99	10.38	122.80	1001.0	0.11840	0.27760	0.3001	0.14710	0.2419	0.07871
1	842517	M	20.57	17.77	132.90	1326.0	0.08474	0.07864	0.0869	0.07017	0.1812	0.05667
2	84300903	M	19.69	21.25	130.00	1203.0	0.10960	0.15990	0.1974	0.12790	0.2069	0.05999
3	84348301	M	11.42	20.38	77.58	386.1	0.14250	0.28390	0.2414	0.10520	0.2597	0.09744
4	84358402	M	20.29	14.34	135.10	1297.0	0.10030	0.13280	0.1980	0.10430	0.1809	0.05883

radius_worst	texture_worst	perimeter_worst	area_worst	smoothness_worst	compactness_worst	concavity_worst	concave points_worst	symmetry_worst	fractal_dimension_worst	Unnamed: 32
25.38	17.33	184.60	2019.0	0.1622	0.6656	0.7119	0.2654	0.4601	0.11890	NaN
24.99	23.41	158.80	1956.0	0.1238	0.1866	0.2416	0.1860	0.2750	0.08902	NaN
23.57	25.53	152.50	1709.0	0.1444	0.4245	0.4504	0.2430	0.3613	0.08758	NaN
14.91	26.50	98.87	567.7	0.2098	0.8663	0.6869	0.2575	0.6638	0.17300	NaN
22.54	16.67	152.20	1575.0	0.1374	0.2050	0.4000	0.1625	0.2364	0.07678	NaN

Figure 4: Output from the built-in function data.head().

Transitioning into exploratory data analysis, it remains paramount to recognize that the insights obtained serve as a guide for subsequent steps in our analysis. These findings assist us in discerning whether feature selection, feature extraction, or classification procedures are warranted. As we unveil patterns, trends, and potential relationships within the data, we are better equipped to tailor our approach, ensuring it aligns with the characteristics of our dataset. We begin by producing a comprehensive chart presenting key statistical measures for each variable, including the count, mean, standard deviation, minimum value, 25th percentile, median (50th percentile), 75th percentile, and maximum value. This representation provides an insightful snapshot of the distribution and central tendencies across our dataset. Building on this newfound insight, a more in-depth exploration of the data is essential for a nuanced understanding. This prompts a shift in attention toward the target variable, 'diagnosis'. We delve into the distribution of this variable by creating a bar plot **(Figure 5)** illustrating the count (malignant = 212, benign = 357), and a pie chart **(Figure 6)** unveiling the percentage distribution (malignant = 37.3%, benign = 62.7%). Observing this distribution, it becomes apparent that the data exhibits a moderate imbalance, with the benign class prevailing. Recognizing the dataset as a binary classification problem, this insight guides us in selecting an optimal algorithm to navigate and interpret the data effectively. Among the potential candidates are logistic regression, random forest classifier, and ensemble classifier, each tailored to handle the intricacies of our dataset with precision.



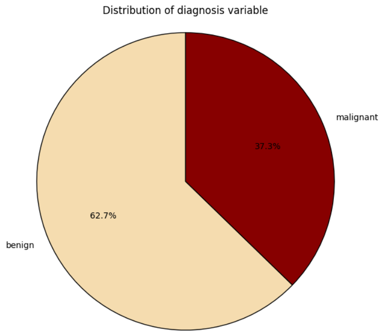


Figure 5: Bar Plot for Target Count

Figure 6: Pie Chart for Target Percentage

Extending our exploration from the distribution of the target variable, our focus now turns to the visualization of predictor variables based on the target variable, employing density plots **(Figure 7)**. These plots serve as a lens through which we examine central tendencies, shape, spread, and overlap. By evaluating the degree of overlap and spread in class distributions concerning predictor variables, we gain valuable insights into feature utility for classification algorithms. Features revealing substantial overlap in distribution are flagged for potential limited discriminatory power. A closer inspection of the plots shines a spotlight on specific attributes – ‘fractal_dimension_mean’, ‘texture_se’, ‘smoothness_se’, ‘symmetry_se’, and ‘fractal_dimension_se’. These features exhibit pronounced overlap between both classes, signaling a thoughtful consideration for their inclusion in classification algorithms.

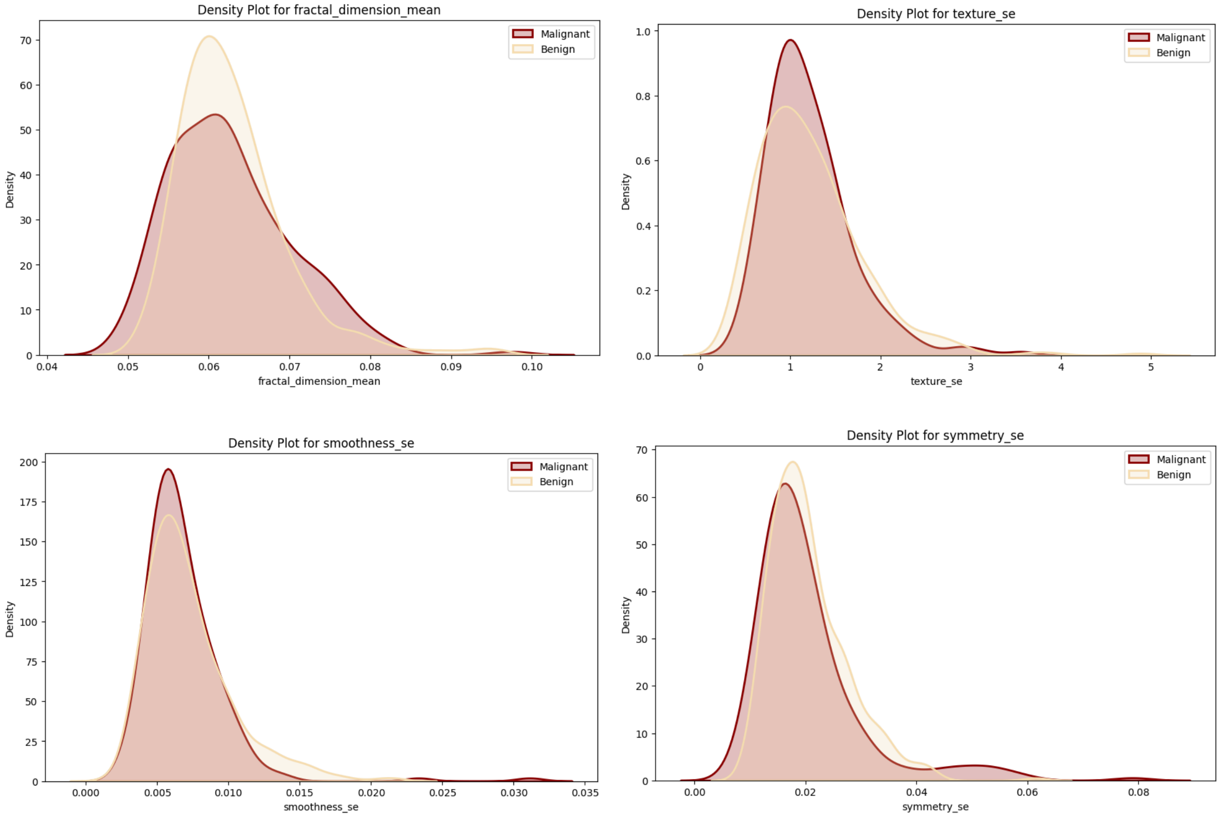


Figure 7: Density plots for features’ distribution.

Alongside assessing the overlap between classes, another critical aspect demanding consideration is multicollinearity. This phenomenon manifests when variables exhibit high, or even perfect, correlation with each other. Such correlation indicates redundancy in the information provided by these variables, leading to potential consequences, including numerical instability and diminished model interpretability. To address this potential issue, we employed a heat map representation **(Figure 8)** of the correlation matrix, providing us with a better understanding of the relationships between variables. Leveraging the accompanying legend, we can readily identify variables that exhibit high positive or negative correlation. In this case, our heat map showcased that many variables had a correlation coefficient close to or even equal to 1.0. This means that these have a high correlation, however, that does not give enough information to show us if these variables should be deemed redundant.

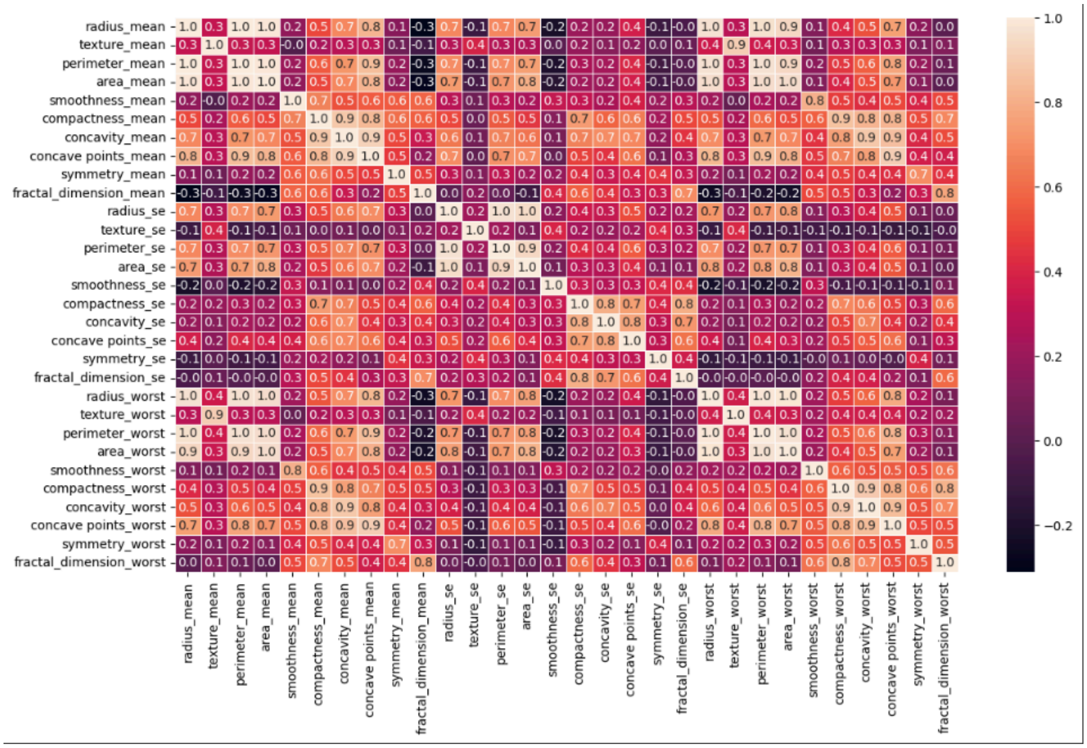


Figure 8: Heat map visualizing the correlation matrix.

Moving forward, we created a series of scatterplots to show and understand the relationships between different pairs of features. We split these charts **(Figure 9)** into three categories: features with positive correlation, negative correlation, or no correlation. In the positive category, we have ‘perimeter_mean v. radius_worst’, ‘area_mean v. radius_worst’, ‘texture_mean v. texture_worst’, and ‘area_worst v. radius_worst’. For the most part, these scatterplots are tightly clustered as they present a perfect correlation coefficient of 1.0. From these plots, we can observe that as both features increase in value, the chance of the cell having a malignant diagnosis also increases. In contrast with the other maps in this category, the ‘texture_mean v. texture_worst’ plot carries a slightly loose clustering meaning that there is more variability. However, it still showcases the same relationship as the others. When comparing with the scatter plots illustrating variables with positive correlation, it becomes evident that variables exhibiting the highest negative correlation within the dataset only reach a correlation coefficient of -0.3. Visually, this manifests as a discernibly more scattered arrangement in the scatter plot compared to its positively correlated counterparts. Despite the increased dispersion, the negative correlation scatter plots unmistakably reveal a consistent negative linear relationship between the two variables depicted in their respective plots. As mentioned earlier, the presence of perfect correlation may impose potential issues. Given that the positive correlation plots exhibit such a phenomenon, thoughtful consideration should be given to the possibility of removing one of the variables involved. Doing so will enhance the interpretability and stability of our analytical framework.

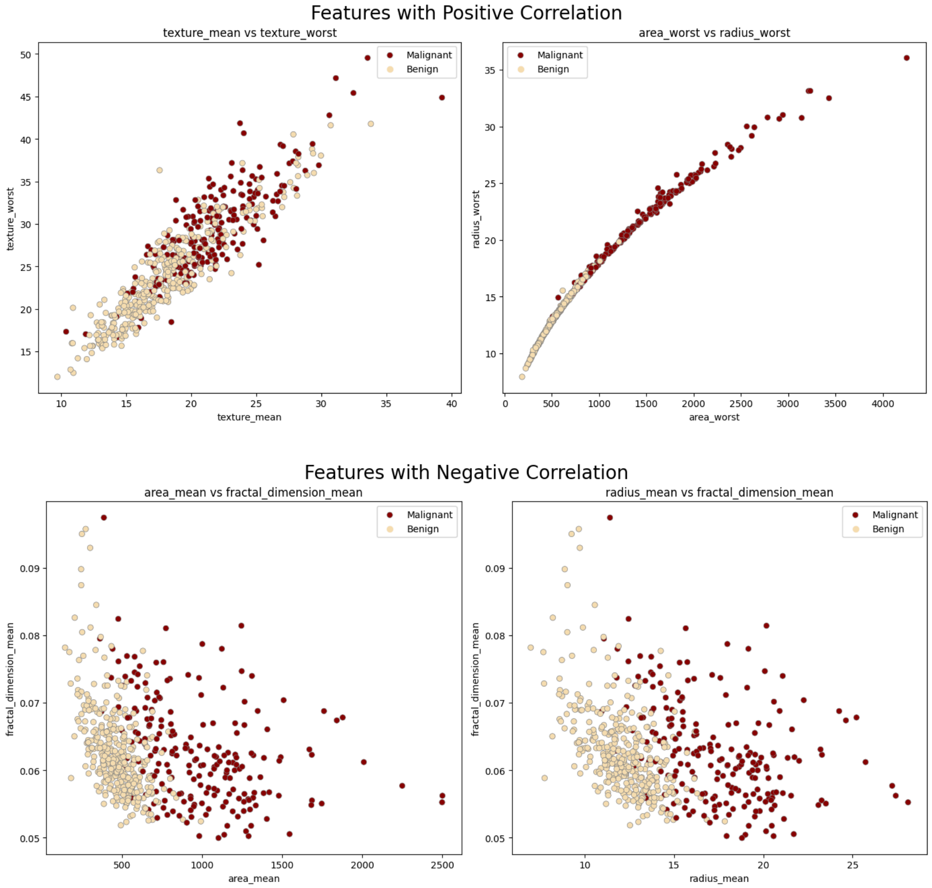


Figure 9: Scatterplots from the Positive and Negative Correlation Categories

After cleaning the data by removing unnecessary variables and performing exploratory data analysis, we conduct feature engineering on the dataset. Feature engineering is another crucial stage in data preprocessing, as it transforms raw data into informative features that improve the performance of the learning model. In the context of using the Breast Cancer Wisconsin dataset to diagnose breast tumors, feature engineering involves feature selection, transforming existing features, dimensionality reduction, and selecting the top features for predictive modeling.

In the first step of feature engineering, we computed mutual information scores for each feature with respect to the target variable, ‘diagnosis’. Similar to correlation values, mutual information scores help quantify the relationship between the features and the target variable, where higher mutual information scores are more informative and indicate that the feature is significant in predicting the target variable. However, mutual information scores do not have a fixed upper limit and may range from 0 to infinity. This characteristic allows mutual information scores to capture both linear and non-linear relationships between variables. However, it's important to note that low mutual information scores may be misleading without context. Therefore, it's essential to consider the specific characteristics of the dataset when interpreting mutual information scores.

Computing each mutual information score was done using the ‘mutual_info_regression’ function, which determines the dependency between two variables. The scores were then plotted in descending order **(Figure 10)**. The barplot of the mutual information scores reveals that ‘perimeter_worst’, ‘area_worst’, ‘radius_worst’, ‘concave points_mean’, and ‘concave points_worst’ contain the highest mutual information scores amongst the features. With values all above 0.4, these features indicate a higher potential importance in diagnosing breast tumors. The top ten features shown in **Figure 10** will be heavily considered in feature selection.

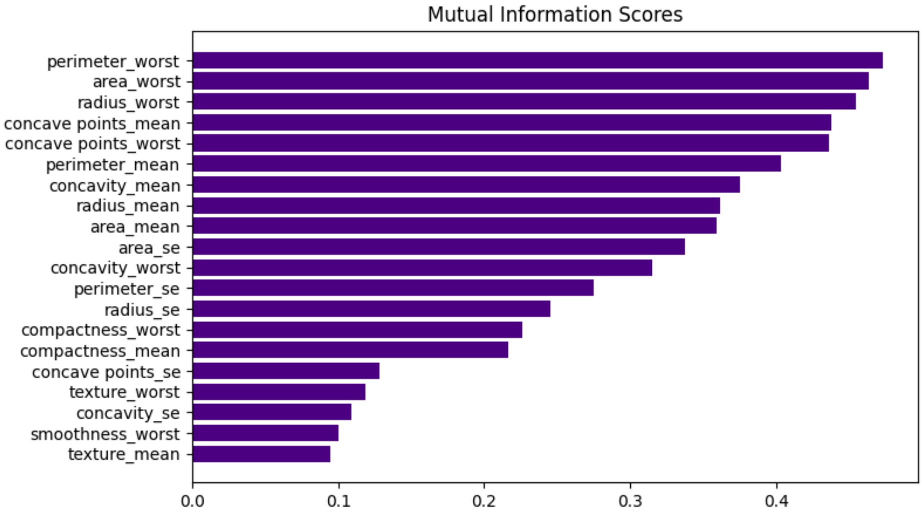


Figure 10: Bar plot displaying the mutual information scores of each feature with respect to ‘diagnosis’.

Prior to performing principal component analysis, we standardize the dataset using the ‘StandardScaler’ function to ensure that all features have a mean of 0 and a standard deviation of 1. Principal component analysis is then performed on the standardized feature matrix to transform the original features of the dataset into a set of uncorrelated principal components. The best number of components to retain is determined by the ‘plot_variance’ function, which visualizes the explained variance ratio of each principal component and the cumulative explained variance. **Figure 11** illustrates the stabilization of the explained variance ratio for every additional principal component after two. The cumulative variance graph also shows that two principal components capture a significant amount of variation in the data. Based on these insights, the choice of two principal components was deemed adequate for summarizing the dataset while retaining most of its variability.

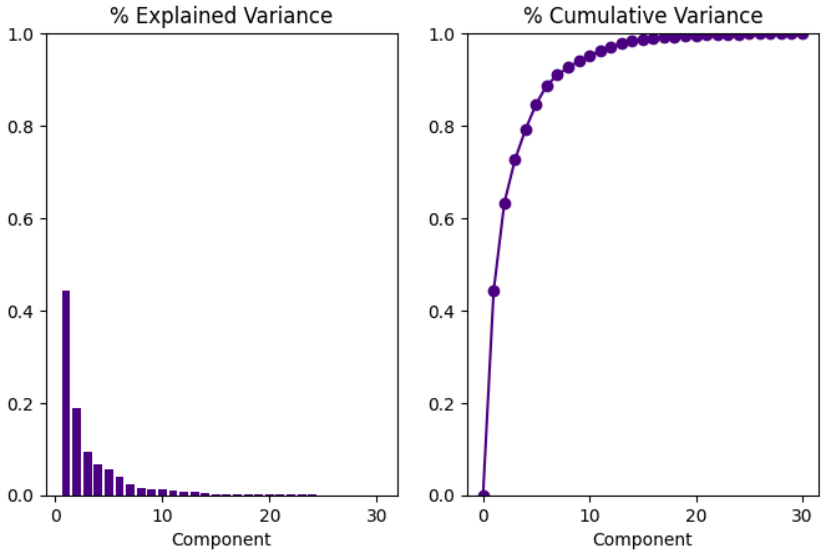


Figure 11: Explained Variance and Cumulative Variance Ratio Analysis plots.

The features selected for each principal component were determined by computing the absolute loading values obtained from PCA. The top three features were selected for each principal component, resulting in the following subsets:

- PC1: concave points_mean, concavity_mean, concave points_worst
- PC2: fractal_dimension_mean, fractal_dimension_se, fractal_dimension_worst

The subset of features is standardized and PCA is applied again, and the transformed data is then visualized using a scatter plot **(Figure 12)**, where each point represents a sample in the dataset and its position is determined by principal component values 1 and 2. The visualization of the PCA helps us understand the relationships between the samples based on their feature values. The explained variance ratios of 44% and 19% for PC1 and PC2 show the proportion of variance in the original data that is captured by each principal component. These values indicate that PC1 summarizes a larger portion of the 63% variability present in the dataset compared to PC2.

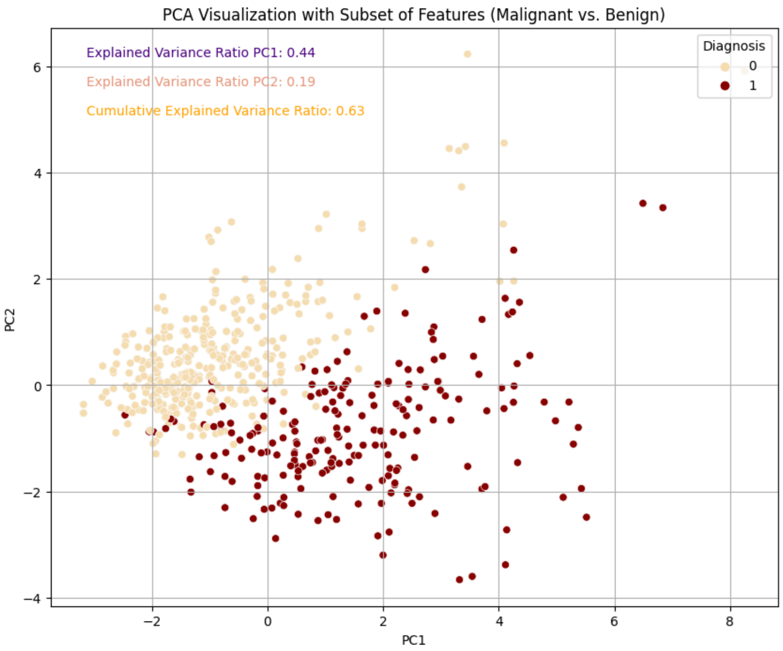


Figure 12: Scatter plot showing the visualization of the dataset after performing Principal Component Analysis (PCA) on the top three features for each component.

The dissimilarity between the top ten features computed using mutual information scores and the features chosen for each principal component capture different aspects of the data. While the top three features for each principal component focus on dimensionality reduction and on features that contribute most to the variability captured, the top ten features computed using mutual information scores prioritize features with the highest predictive power for diagnosing breast cancer, regardless of their contribution to dimensionality reduction. This emphasizes the importance of the multifaceted nature of feature selection in breast cancer diagnosis. Integrating these approaches allows us to take advantage of the strengths of each method while gaining a comprehensive understanding of the dataset.

Overall, the process of data cleaning and exploratory data analysis has improved the quality and understanding of the Breast Cancer Wisconsin data set. Cleaning processes and EDA gave us critical insights that provide a foundation for breast tumor diagnosis analyses. Moreover, feature engineering and the utilization of principal component analysis transformed the raw data into more informative features. Employing mutual information scores prioritized predictive features, while PCA reduced dimensionality. The integration of these approaches emphasizes a multifaceted feature selection strategy, ensuring a comprehensive understanding and facilitating the development of accurate data models in medical diagnostics.

Enhancing Breast Tumor Diagnosis: A Comprehensive Analysis

Data Science II: Deliverable 2

COSC 4337

Submitted to

Dr. Ricardo Vilalta

Ly Ha 1920058

Erica Hay 2049545

Khuong Ngo 1857210

In 1992, research conducted at the University of Wisconsin aimed to uncover correlations between cell characteristics and contextual features, with the ultimate goal of enhancing diagnostic accuracy by identifying traits linked to malignancy. The features analyzed describe various characteristics of cell nuclei present in the images. Leveraging machine learning techniques, our goal is to predict whether a given breast mass is benign or malignant based on these features. The importance of precise classification in breast cancer diagnosis cannot be overstated, given its pivotal role in treatment planning and patient outcomes. With a dataset consisting of 30 predictor variables, there is a likelihood of inter-variable correlation. Consequently, the selection of appropriate machine learning models becomes pivotal. Hence, in this report, we explore the effectiveness of multiple logistic regression, decision trees, and random forest classifiers in accurately classifying the binary target variable. Our objective here is to assess and compare the performance of these models in discerning the nature of breast masses with precision. Through evaluation and comparison of these models, we seek to provide insights into their relative strengths and weaknesses, aiding clinicians with valuable information for making well-informed decisions in breast cancer diagnosis and treatment planning.

Our first model, multiple logistic regression, is a statistical technique used for binary classification tasks, where the goal is to predict the probability of a binary outcome based on multiple predictor variables. Given the nature of our dataset, which contains 30 predictor variables, multiple logistic regression emerges as an obvious modeling approach. Its inherent interpretability allows us to easily interpret the relationship between each predictor variable and the probability of a breast mass being malignant or benign.

To optimize the performance of our logistic regression model, we employed GridSearchCV, a powerful method for hyperparameter tuning. GridSearchCV systematically explores a predefined grid of hyperparameters and evaluates the model's performance using cross-validation. In the case of logistic regression, the grid included two sets of hyperparameters: one for the penalty parameter ("l1" and "l2") and another for the C parameter (0.001, 0.01, 0.1, 1, 10, 100, 1000). By utilizing 5-fold cross-validation, GridSearchCV meticulously iterates through each combination of hyperparameters, training the model on a subset of the training data (training fold) and evaluating its performance on a distinct subset (validation fold). With 14 hyperparameter combinations fitted across 5 folds each, totaling 70 iterations, GridSearchCV calculated performance metrics such as accuracy, precision, recall, or F1-score for each validation fold. Our choice of performance metric was accuracy.

Upon assessing all hyperparameter combinations, GridSearchCV identified the set yielding the best performance metric across all folds. Ultimately, the combination of a C value of 0.1 with L2 regularization was deemed the optimal configuration for our logistic regression model.

Next, we decided to create a decision tree. Decision trees are ideal for capturing nonlinear relationships between features and the target variable, making them suitable for our classification problem. The model provides insights into the hierarchy of risk factors associated with breast cancer.

Much like in multiple logistic regression, we conducted hyperparameter tuning using GridSearchCV. For our decision tree model, we focused on tuning five main parameters:

- criterion: Determines the function to measure the quality of a split. → 'gini' or 'entropy'.
- max_depth: Controls the maximum depth of the tree, influencing the complexity and potential for overfitting. → range of 2 to 32
- min_samples_leaf: Specifies the minimum number of samples required to be at a leaf node, helping prevent overfitting by simplifying the model. → range of 2 to 9
- min_samples_split: Specifies the minimum number of samples required to split an internal node, encouraging the tree to only make splits when necessary.
- splitter: Determines how the algorithm selects features to split on at each node. → 'best' or 'random'.

After completing the hyperparameter tuning process, GridSearchCV identified the best combination of parameters for our decision tree model: {'criterion': 'gini', 'max_depth': 8, 'min_samples_leaf': 2, 'min_samples_split': 2, 'splitter': 'random'}. This optimal configuration was determined based on its performance on the validation folds during cross-validation.

Lastly, our third model is the random forest model. It is a powerful learning method which builds upon the decision tree algorithm and excels in handling complex data by aggregating predictions of multiple decision trees, reducing overfitting, and increasing accuracy. Breast cancer diagnosis often involves analyzing multiple complex factors, and random forests excel at handling complex interactions between the features and the target variable, allowing them to identify subtle patterns that may not be apparent using other models.

Unlike the first two models, we employed Bayesian optimization using BayesSearchCV, a sophisticated method for hyperparameter tuning. This approach efficiently explores the hyperparameter space to identify the combination that maximizes the model's performance. The key hyperparameters are similar to that of decision tree models, however there is an additional parameter:

- n_estimators: Represents the number of trees in the forest. Increasing the number of trees generally improves performance but also increases computational cost. → range of 10 to 400

For the other parameters we experimented with the following:

- criterion: 'gini' or 'entropy'.
- max_depth: range of 1 to 11
- min_samples_split: range of 2 to 20
- min_samples_leaf: range of 2 to 20

Using Bayesian optimization, BayesSearchCV iteratively explores the hyperparameter space based on the model's performance, gradually converging towards the optimal configuration. This process is guided by an acquisition function that balances exploration (searching different regions of the space) and exploitation (focusing on regions likely to yield improvements).

After completing the hyperparameter tuning process, BayesSearchCV identified the best combination of parameters for our random forest model: {'criterion': 'entropy', 'max_depth': 8, 'min_samples_leaf': 2, 'n_estimators': 251}. This optimal configuration was determined based on its performance on the validation folds during cross-validation.

Now that we have optimal parameters for each of the models, let's build and explore performance metrics to see which model performed the best on the dataset. In order to evaluate the performance of each model, we employed several key metrics: the confusion matrix, learning curves, and cross-validation.

First up, let's take a look at the confusion matrix produced by each model (**Figure 1**). The confusion matrix is a valuable tool in evaluating the performance of classification models, such as those used in our breast tumor diagnosis analysis. It provides a detailed breakdown of the model's predictions compared to the actual labels in the dataset. Specifically, the confusion matrix displays four important metrics:

- True Positives (TP): The number of instances correctly predicted as positive (malignant tumors in our case).
- False Positives (FP): The number of instances incorrectly predicted as positive when they are actually negative (benign tumors classified as malignant).
- True Negatives (TN): The number of instances correctly predicted as negative (benign tumors).
- False Negatives (FN): The number of instances incorrectly predicted as negative when they are actually positive (malignant tumors classified as benign).

By examining these metrics, we gain insight into how well the model is performing in terms of correctly classifying instances.

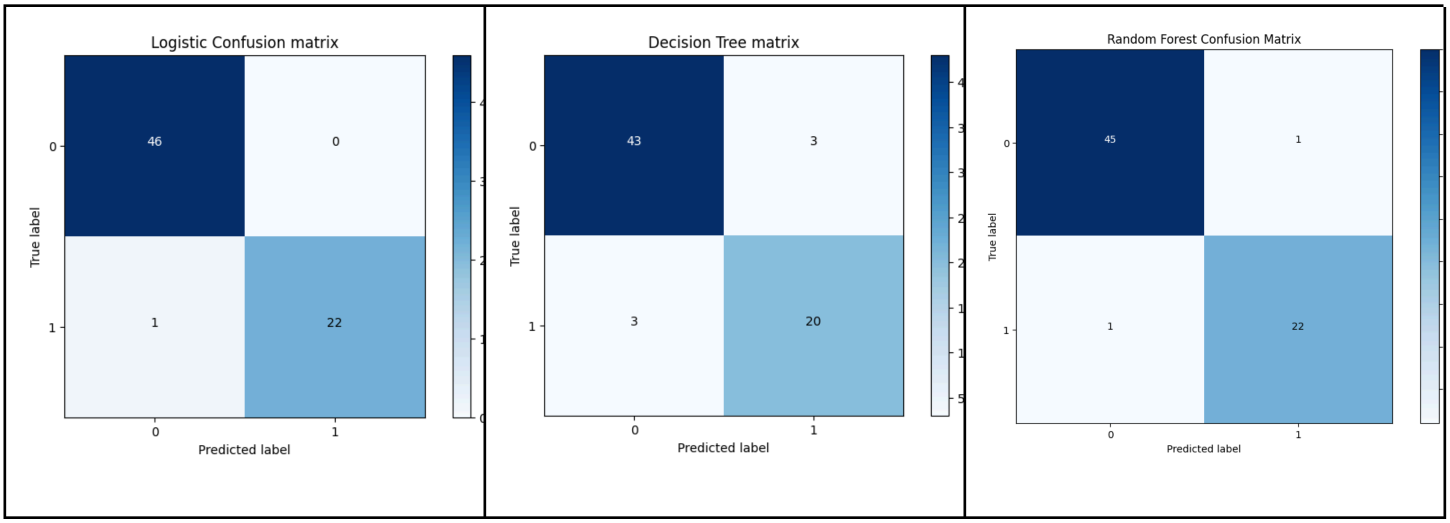


Figure 1: Confusion Matrix for all three models

	Logistic Regression	**Decision Tree**	**Random Forest**
True Positive	46	43	45
False Positive	0	3	1
False Negative	1	3	1
True Negative	22	20	22

Figure 2: Confusion Matrix Results

Based on our results from the confusion matrices (**see Figure 2**), we can see that all models achieved a relatively high number of true positives and true negatives, indicating that there is strong predictive performance across all models. However, there are differences in the number of false positives and false negatives, which impacts overall accuracy and reliability of each model. As multiple logistic regression does not have any false positives, this could mean that it does a better job at identifying malignant tumors. In addition, decision trees have 3 false positives and 3 false negatives meaning performance wise, it lacks in comparison to the other models as it cannot correctly classify the tumors. The random forest has 1 false positive, suggesting that the model is more adept at correctly identifying malignant tumors compared to the decision trees, while still having room for improvement when compared to multiple logistic regression.

Further analysis involved calculating additional performance metrics: accuracy, precision, recall, and F1-score. As seen in **Figure 3**, it's evident that the logistic regression model outperforms both the decision tree and random forest models across all metrics. It achieves the highest accuracy, precision, and F1-score, indicating superior performance in classifying the breast tumors as malignant or benign. While the decision tree and random forest models still demonstrate very high performances, they fall slightly short in comparison to that of the logistic regression model.

	Logistic Regression	**Decision Tree**	**Random Forest**
Accuracy	98.6%	91.3%	97.1%
Precision	100%	91%	97%
Recall	95.7%	91%	97%
F1-Score	97.8%	91%	97%

Figure 3: Classification Report of each model

Moving on to learning curves (**Figure 4**) for our models, the learning curves showcase the evolution of both training and cross-validation scores as the size of the training dataset increases. The training score curve allows us to assess whether the model is learning from the data effectively, whilst the validation score curve helps us assess how well the model generalizes unseen data. For the logistic model, initially both training and cross-validation scores increase with the addition of training examples, with cross-validation scores at a steeper increase. As the training set size continues to increase, the gap between the training and cross validation scores narrows, suggesting that the model's generalization performance improves with more data. Eventually, the curve plateaus around a training score of 0.98. As for the other two models, the random forest's learning curve is similar to that of the logistic model, where there is a steep increase in the cross-validation score initially as the training examples increase and stabilizes around 0.95. However, the training score seems to remain stable the entire time with a score of 1.00. As for the decision tree model, we can see that the learning curve is quite different from the other two models as the training and cross-validation score trends never converge. However, the two scores do follow the same trends as the other two models, whereas as the training examples increase, the cross-validation score increases. This means that for both the logistic model and the random forest model, we can observe effective learning and generalization. In contrast, the decision tree model also reflects effective learning and generalization but since it does not converge, we hit an issue with the model.

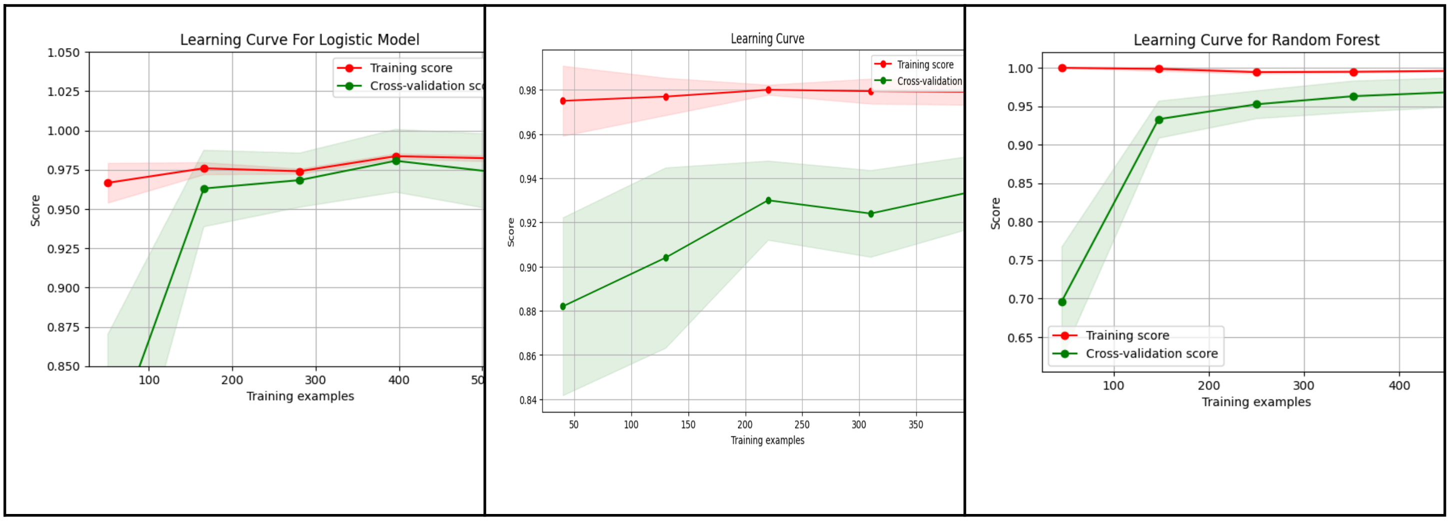


Figure 4: Learning Curve for all three models

In fact, a learning curve doesn't just give us information on the efficacy of learning and generalization of the model. It is also a bias-variance model. We can tell through the gap between the training and validation curves. In both the logistic model and the random forest model, we can see that since the two curves converge, there is not a large gap between the two curves. This is the ideal scenario as they are converging to a high value as the number of training examples increases. In contrast, in the decision tree model, there is a large gap between that of the training and validation scores, where the training score is significantly higher than the validation score. This means that the model has high variance, indicating overfitting.

Lastly, for performance evaluation, we took a look at the cross-validation scores of each model (**Figure 5**). The logistic regression and decision tree models both exhibit consistency in the cross-validation scores, indicating stability in the performance of the models across different folds of the data. However, the scores of the logistic regression model are higher across all folds of the data, signifying better performance than that of the decision tree model. In contrast, the random forest model has more variation in the cross-validation scores. Compared to the decision tree model, the random forest has higher scores for all folds, suggesting that the random forest approach effectively improved predictive accuracy in comparison to individual trees. The stability and the high scores across all folds of the logistic regression model leads us to believe that it performed best among the three models for cross-validation.

	Logistic Regression	**Decision Tree**	**Random Forest**
Fold 1	0.99	0.92	0.94
Fold 2	0.98	0.9	0.96
Fold 3	0.97	0.92	0.99
Fold 4	0.97	0.92	0.98
Fold 5	0.97	0.91	0.97
Mean	0.976	0.914	0.968

Figure 5: 5 Fold Cross-Validation Scores for all Models

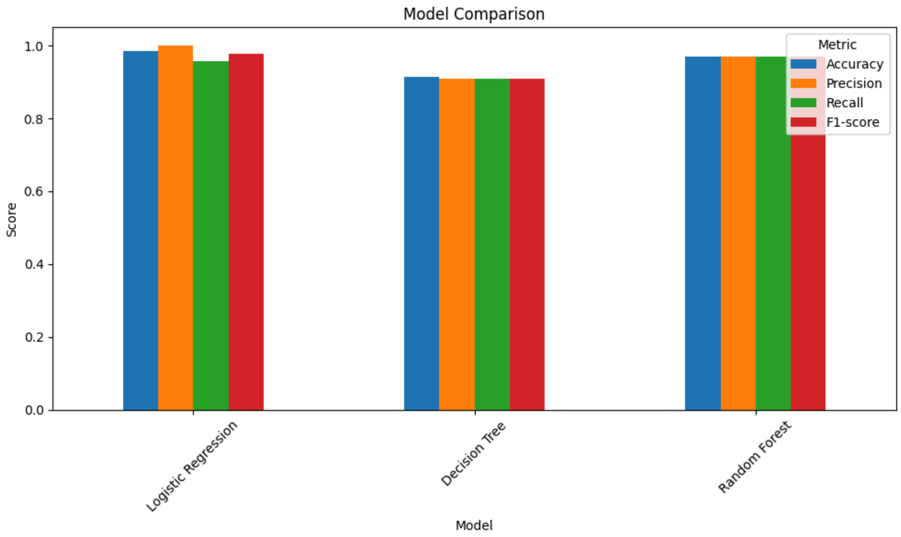


Figure 6: Visual Comparison of all Three Models and their Metrics

Based on our extensive analysis, we can conclude that the logistic regression model performed best overall (**as seen in Figure 6**), followed by the random forest model. In terms of accuracy, precision, recall, and F1-score, while also showing effective learning and minimal overfitting, the logistic regression model demonstrated superior performance in predicting breast tumor malignancy. Although slightly less accurate than logistic regression, the random forest model also showed great performance considering all metrics. The random forest model also demonstrated exceptional learning capabilities and robust performance. In summary, while logistic regression performed best amongst the three models, each model performed relatively well based on its predictive accuracy. All three models offer valuable insights into the classification of breast tumors, and can be considered useful tools when researching the most effective treatments and diagnosis.

Enhancing Breast Tumor Diagnosis: Insights & Implications

Data Science II: Deliverable 3

COSC 4337

Submitted to

Dr. Ricardo Vilalta

Ly Ha 1920058

Erica Hay 2049545

Khuong Ngo 1857210

In the medical field, traditional tumor diagnosis has heavily relied on full biopsies. While laboratory tests and imaging techniques serve as complementary diagnostic tools, the definitive confirmation for many medical practitioners often requires a biopsy, despite its invasiveness. Nonetheless, the invasive nature and inherent complexities of biopsies emphasize why exploring alternative diagnostic modalities is desirable. Due to the need for advancement in this area, a study was done at the University of Wisconsin in 1992, spearheaded by collaborative efforts between the Departments of Computer Science and Surgery. This study aimed to discern correlations among tumor features and malignancy by using interactive image processing techniques and linear programming-based inductive classification. Building on the insights gained from this research, our application of logistic regression, a supervised learning technique, seeks to enhance tumor diagnostics by identifying correlations among tumor features and malignancy. This approach shows promises to advance the diagnostic process by elucidating malignant features and improving diagnostic accuracy.

Using the dataset from this study, we analyzed the data from 569 images of fine needle aspirations of breast masses in order to examine the characteristics of the cells and contextual features. When taking a look into what the dataset consists of, we are given twelve main features. In the table below, we are able to see each feature, its data type, and its description. The dataset holds three different data types:

1. ‘Int’ types: data that is presented as an integer (number without decimals)
2. ‘Str’ types: data that is presented as a string (used to represent text)
3. ‘Float’ types: data that is presented as a float number (number with decimal points)

Column	Data Type	Description
ID	int	Unique identification number for each cell nucleus
Diagnosis	str	Malignant (M) or Benign (B) tumor diagnosis
Radius	float	Mean distance from the center to points on the perimeter
Texture	float	Standard deviation of gray-scale values
Perimeter	float	Perimeter of the cell nucleus
Area	float	Area of the cell nucleus

Smoothness	float	Local variation in radius lengths
Compactness	float	$Perimeter^2 \div area - 1.0$
Concavity	float	Severity of concave portions of the contour
Symmetry	float	Symmetry of the cell nucleus
Fractal Dimension	float	$coastline\ approximation - 1$

Furthermore, the data set includes important numbers from measurements of each tumor, giving us a ‘mean’, ‘standard error’, and ‘worst’ (the mean of the three largest values) for each of the features in the table. This gives us a comprehensive representation of the distribution of values within each cell feature. The ‘mean’ provides the measure of central tendency. ‘Standard error’ reflects the variability or uncertainty associated with the mean estimate, while ‘worst’ gives us insight on the rand and potential significance of observed values.

Our analysis begins by leveraging logistic regression, a statistical technique, that models the probability of a breast mass being malignant based on these features. Logistic regression helps us predict the likelihood of an event happening. In medical settings, it is often used to predict the probability of a certain outcome based on different factors. In our case, clinicians may need to figure out whether a patient with certain symptoms has breast cancer. Logistic regression helps us analyze how different factors, such as age, gender, or test results, contribute to the likelihood of the patient having that condition. The model calculates coefficients for each symptom or test result, allowing us to identify which features exert the most influence in predicting malignancy. Positive coefficients are indicative of features associated with an increased likelihood of malignancy, while negative coefficients suggest the opposite effect. Additionally, the results of logistic regression models are often presented as odds ratios. An odds ratio greater than 1 suggests that a factor increases the likelihood of the outcome, while an odds ratio less than 1 suggests it decreases the likelihood. Our predictive analysis yields valuable insights into the early detection of breast cancer. Here are the results generated by our model:

	Coefficient	**Odds Ratio**
Mean Radius	0.36365307094544297	1.4385750440947966

Mean Texture	0.4074393127528309	1.5029642320045415
Mean Perimeter	0.3529376427600434	1.423242391199154
Mean Area	0.3794043268232813	1.4614138051047514
Mean Smoothness	0.12682831311848605	1.1352220983075227
Mean Compactness	-0.029350417935999563	0.9710761223486043
Mean Concavity	0.38782449793216595	1.4737711117124
Mean Concave Points	0.4695575981337264	1.5992865093521276
Mean Symmetry	0.08404965184017221	1.087682897925519
Mean Fractal Dimension	-0.2165399524055209	0.8053003605933824
Standard Error Radius	0.515401121391353	1.6743099686628509
Standard Error Texture	-0.025325117178481602	0.9749928735617074
Standard Error Perimeter	0.3702632609240722	1.44811579678877
Standard Error Area	0.4127113372849948	1.5109088199367307
Standard Error Smoothness	0.10052371499309481	1.1057498642435377
Standard Error Compactness	-0.26630051330797094	0.7662088369478764
Standard Error Concavity	-0.0687305907073329	0.9335781608330143
Standard Error Concave Points	0.10021742829318461	1.1054112396274478
Standard Error Symmetry	-0.13301222278020353	0.8754543915567699
Standard Error Fractal Dimension	-0.25309044898746136	0.7763976542660842
Worst Radius	0.512019597615853	1.6686578115340118
Worst Texture	0.5777597694624887	1.7820417713070849
Worst Perimeter	0.4553482454062391	1.57672237376027
Worst Area	0.4897350121871681	1.6318837333547729
Worst Smoothness	0.4155369982041489	1.5151841734466325
Worst Compactness	0.1528319885011118	1.1651292073927302
Worst Concavity	0.4216434214270429	1.5244648362593824
Worst Concave Points	0.4608214440780806	1.5853757477781627

Worst Symmetry	0.48030003103106744	1.6165593474077335
Worst Fractal Dimension	0.10958302194253271	1.115812704027996

Figure 1: Coefficients and Odds Ratios of Each Feature

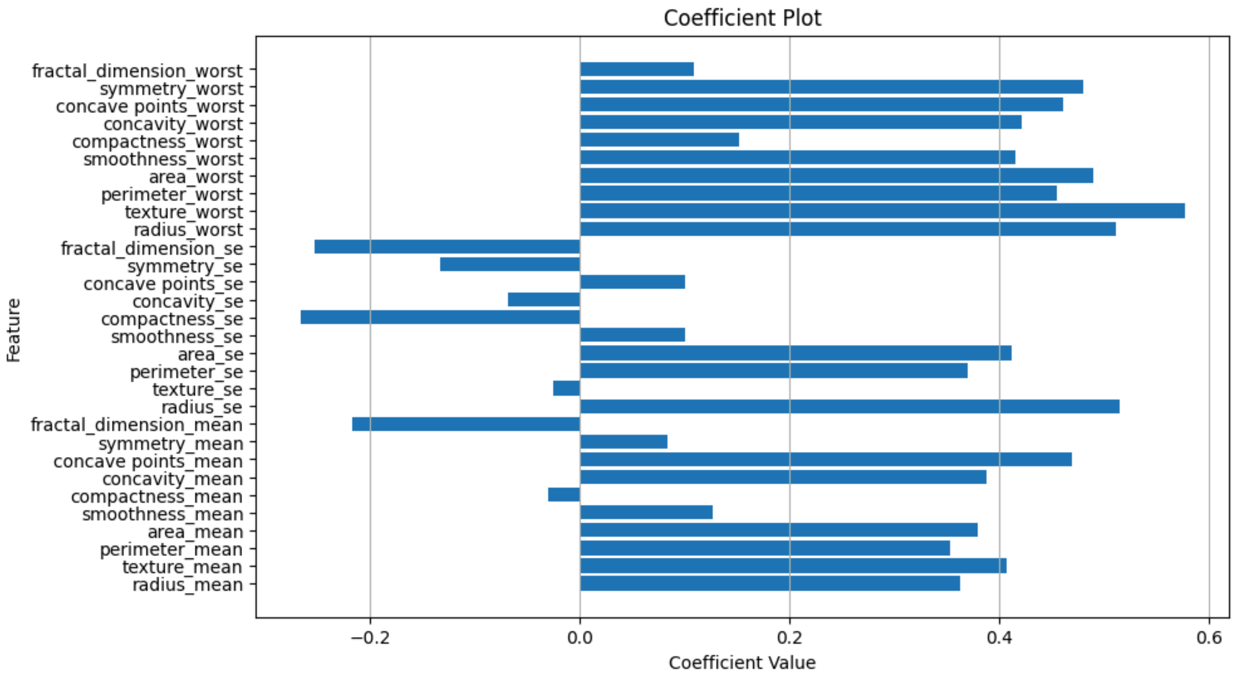


Figure 2: Coefficient Plot

As seen in **Figure 2**, features such as concave points, radius, perimeter, and area emerge as significant predictors of malignancy, with higher values associated with an increased likelihood of a tumor being malignant since they have a coefficient value closer to 1. These findings align with established medical knowledge, where irregularities in cell morphology, size, and shape are indicative of cancerous growth. Conversely, features like smoothness, compactness, and fractal dimension demonstrate a negative correlation with malignancy, suggesting that smoother, more uniform cell characteristics are often indicative of benign tumors. We can also take a closer look into each feature's logistic regression curve, in **Figure 3**, we are able to observe that as the S curve gets closer to 1, or M in this case, the observed cells that exhibit more concave points have a higher correlation to malignancy.

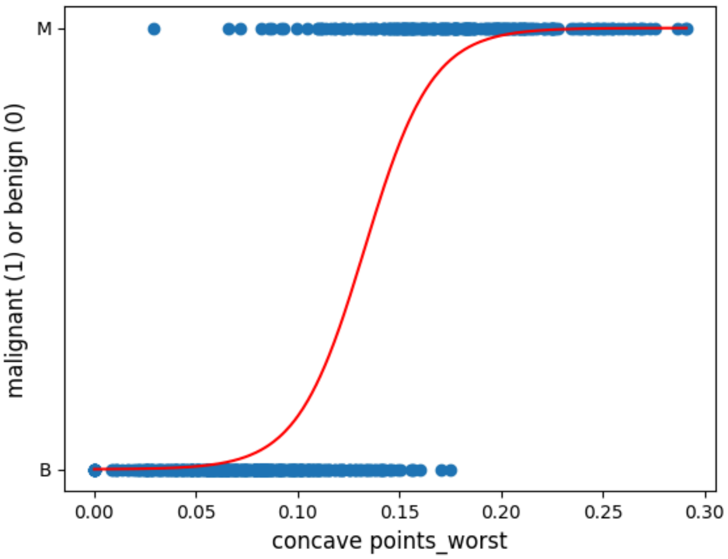
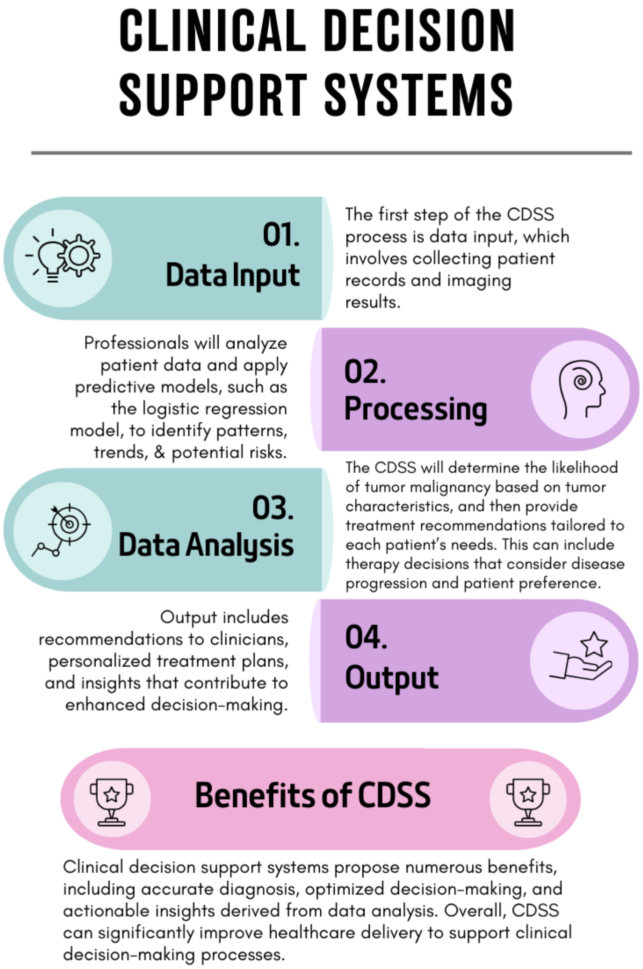


Figure 3: Logistic Regression curve of Concave Points

The implications of these predictions for decision-making in clinical practice are profound. Medical professionals can leverage the predictive model's insights to prioritize diagnostic evaluations for patients with breast masses exhibiting characteristics associated with malignancy. For instance, patients with higher values of concave points and larger radii may warrant further diagnostic procedures, such as biopsies or additional imaging tests, to confirm the presence of cancerous growths. Whereas patients with smoother, more regular cell characteristics may be reassured of the likelihood of a benign tumor, potentially avoiding unnecessary invasive procedures.

The insights gained from our analysis of breast tumor classification has immense potential for transforming clinical practices and healthcare management. Healthcare professionals and organizations can enhance diagnostic accuracy and treatment strategies by leveraging the machine learning techniques and data-driven insights.

A primary application of the findings of our analysis involves the development of clinical decision support systems (CDSS), a type of software that supports the decision-making of clinicians or healthcare professionals using analytical data. Clinical decision support systems can utilize the predictive models used for the breast cancer data set to assist professionals in diagnosing tumors more accurately and efficiently. CDSS can analyze patient data, including imaging results and clinical parameters, to provide recommendations and insights to doctors. The recommendations may include the likelihood of malignancy based on various tumor characteristics, treatment options tailored to each individual patient, and assessments to guide therapy decisions. Integrating machine learning techniques and predictive analytics into clinical workflows can help CDSS enhance decision-making, leading to more accurate diagnosis and personalized treatment plans, improving patient outcomes.



CLINICAL DECISION SUPPORT SYSTEMS

01. Data Input

The first step of the CDSS process is data input, which involves collecting patient records and imaging results.

02. Processing

Professionals will analyze patient data and apply predictive models, such as the logistic regression model, to identify patterns, trends, & potential risks.

03. Data Analysis

The CDSS will determine the likelihood of tumor malignancy based on tumor characteristics, and then provide treatment recommendations tailored to each patient's needs. This can include therapy decisions that consider disease progression and patient preference.

04. Output

Output includes recommendations to clinicians, personalized treatment plans, and insights that contribute to enhanced decision-making.

Benefits of CDSS

Clinical decision support systems propose numerous benefits, including accurate diagnosis, optimized decision-making, and actionable insights derived from data analysis. Overall, CDSS can significantly improve healthcare delivery to support clinical decision-making processes.

Figure 4: Example of how CDSS could be applied to tumor classification

The predictive models used in our analysis can play a crucial role in the development of personalized treatment plans for breast cancer patients. Considering multiple variables that may indicate tumor malignancy, the models can provide professionals with valuable insights into the specifics of each tumor, allowing them to tailor to the specific needs of the patient. Professionals will be able to analyze various tumor features such as size, shape, and cellular characteristics to assess how aggressive and malignant the tumor is. In addition, these models can factor in additional patient information such as age, health status, and genetic conditions. This information can help doctors in determining the best treatment plan for the patient, such as surgery, chemotherapy, or a combination of treatments, while considering medical history and also minimizing potential side effects or complications. The models can also assist in monitoring the effectiveness of treatment plans by analyzing the changes in tumors and patient health, allowing professionals to make adjustments to the treatment plan. Overall, the integration of predictive modeling into treatment planning for breast cancer patients holds promise for improving clinical outcomes while enhancing patient care.

Our analysis of breast tumor diagnosis using machine learning techniques provides valuable insights into the classification of tumors and its implications for clinical decision-making and healthcare management. Through the application of logistic regression modeling on the comprehensive breast cancer dataset, we have achieved strong performance in accurately classifying tumors based on various factors. This predictive capability has promise for improving patient care and treatment strategies, underscoring the potential of machine learning in diagnostic accuracy and decision-making for a wider range of cancer. In summary, our analysis represents a significant step forward in utilizing machine learning for medical diagnoses and improving healthcare practices.