Enhancing Breast Tumor Diagnosis: A Comprehensive Analysis

Data Science II

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.

|  |  |  |
| --- | --- | --- |

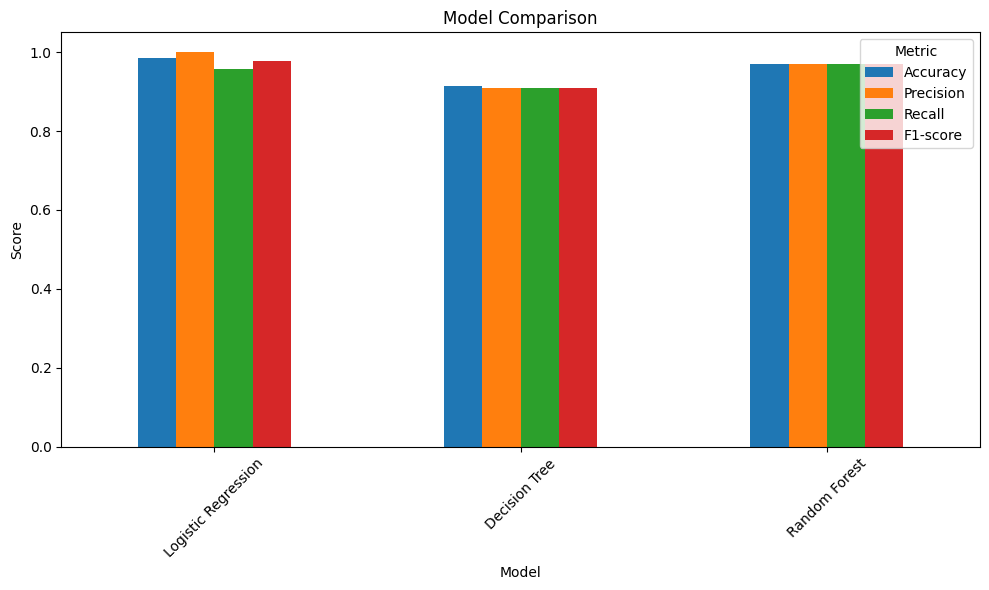
*Figure 4: Learning Curve for all three models*

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.