Nina Cunha

Loghan Hughes

Arya Shenoy

Predicting Tumor Malignancy

Abstract

Cancer remains a major focus of research due to its complexity and impact on human health. Part of cancer research is using new techniques and emerging technologies to determine cancer cells before they get too large to treat. This includes various data science and machine learning techniques to try to understand the intricacies of how cancer forms. Using data from the UC Irvine Machine Learning Repository, logistic regression, random forests, and k-nearest neighbors were applied to classify tumors more rapidly and accurately than manual diagnosis. Analysis revealed that a small number of cellular features strongly predict tumor malignancy, providing insight into early diagnostic criteria.

Background

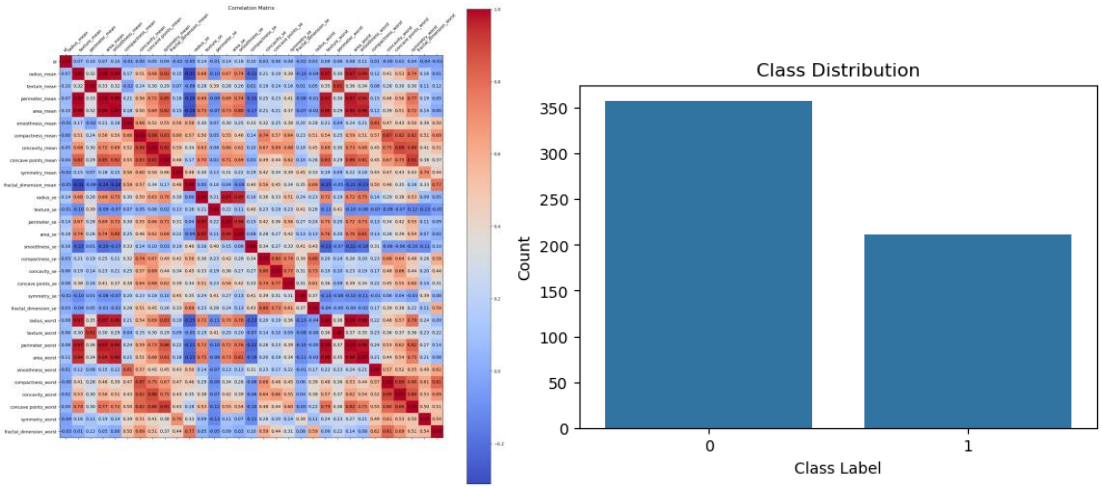
The Center for Disease Control and Prevention (CDC) lists cancer as the number two cause of death for the United States. (CDC) These statistics are staggering and have driven massive amounts of research into cancers. Not only are biologists working for a treatment, but data science can also be leveraged to attempt to predict the type, location, and even risk genes, based on genetic and physical properties of tissue samples. A tumor is classified in one of two categories when assessing its severity and if it is cancerous. If a tumor is benign, the cells within it are not considered to be cancerous. This means the tumor is not likely to spread to nearby tissue, and if it's growing, usually at a slow rate. Importantly, these tumors are not likely to regrow after removal and are usually considered less severe. However, if a tumor is classified as malignant, this means that the individual has cancerous cells. These cells are shaped differently, secrete hormones, invade nearby tissues, and partake in uncontrollable cell replication. Malignant tumors are much more dangerous as they can spread throughout the body when cancerous cells invade the basal membrane and move through the blood stream invading healthy tissue (Baptist Health). These key differences in tumor type make identification especially crucial and can help to get a jump on treatment if needed, saving lives before the cancer spreads. This project proposes ways to predict tumor malignancy based on physical properties of tissue samples.

Problem Statement

Our goal is to develop a model to accurately classify tumors as malignant or benign based on a series of physical features from samples of the tumor.

Data

This data set was taken from a Kaggle dataset taken from a UC Irvine study of breast cancer containing various characteristics of cell nuclei present in breast tissue, such as texture, smoothness, and symmetry. Features such as radius, texture, and perimeter have established relevance in tumor diagnosis, making this dataset appropriate for our classification task. Some of which are malignant tumors or benign. This dataset includes 1 categorical variable, the type of tumor, and 31 features. In the exploration of our dataset, the correlation table shows lots of highly correlated features, which makes sense since lots of the features in this dataset include different ways of quantifying the same measurement. However, this would need to be considered when choosing our variables for our final models. Another note about our data is that there is a slight class imbalance, which we addressed by adjusting the classification threshold from 0.5 to 0.4. This approach simultaneously reduces the effects of the imbalance while increasing the model's ability to correctly identify malignant cases, prioritizing sensitivity in cancer detection.



Preprocessing

In cleaning the dataset to be ready to use on the models, we dropped the 'id' feature since it serves as a unique identifier for each patient and does not carry any predictive information relevant to the diagnosis. In fact, because it is directly tied to the patient rather than their condition, keeping it could unintentionally bias the model if not handled properly. We also encoded the target variable such that malignant tumors were represented by a value of 1 and benign tumors by 0, enabling binary classification. After removing the 'id' feature, the dataset was left with 30 features that are more suitable for modeling and less likely to introduce noise or redundancy during training. Each model was then evaluated using an 80/20 train test split.

We used cross-validation to select features for each model, aiming to reduce the number of variables and avoid capturing noise in the dataset. Within each fold, the data was standardized via `StandardScaler` to ensure that the scale of the data is not the factor determining the importance of the feature. Standardization also helps ensure that the evaluation during cross-validation accurately reflects the model's generalization performance.

To better serve the objectives of our paper, the classification threshold for predicting malignant tumors was intentionally lowered. This adjustment prioritizes a reduction in false negatives, ensuring that potentially cancerous cases are less likely to be missed. While this approach may increase the number of false positives, leading to a potential slightly overfit mode, such outcomes are acceptable in the context of cancer detection, where the consequences of missing a malignant tumor far outweigh the inconvenience of a false alarm.

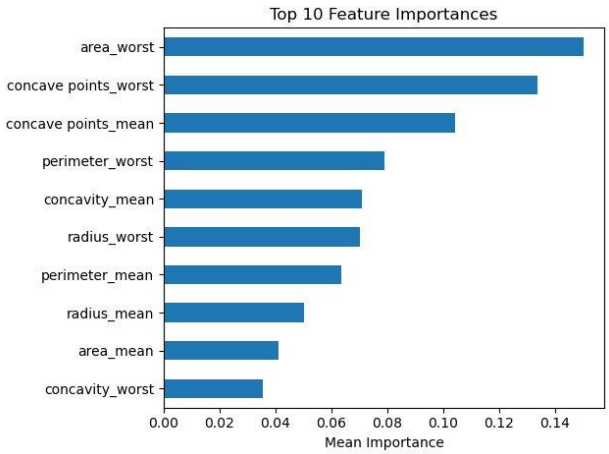
Modeling Techniques

Logistic Regression

The Logistic Regression model was chosen to act as a base model for comparison. This is due to the fact that it is widely used in binary classification tasks. It allows for easy interpretability and to act as a baseline and it requires relatively minimal computational resources. However, because many features were highly correlated, this posed a problem for the Logistic Regression model, which assumes feature independence. To address this, Principal Component Analysis (PCA) was performed within each fold of cross-validation to reduce dimensionality and decorrelate the features. The top 5 features that contributed most to the variance in the top Principal Components were selected based on their loading's values. It resulted in 'fractal_dimension_worst', 'fractal_dimension_se', 'compactness_worst', 'radius_se', and 'area_se' being the final chosen features. The top 5 were chosen since there was a significant decrease in loading values after the top five features. This ensures that the model retains the most influential and informative features while minimizing noise and reducing the risk of overfitting.

Random Forest

We chose to include a random forest model as it combines predictions from multiple decision trees, and it utilizes bootstrap aggregating, training each tree on a subset of the data, reducing reliance on any single feature and naturally decreasing correlation within the model. Within each fold of the 5-fold cross-validation used for feature selection, the feature importance scores were extracted. After completing the cross-validation the top 10 most important features were selected based on these mean values across the folds. 10 was chosen since the values of the average importance scores had a natural cutoff point at 10 variables. This approach reduces the number of features in a model and helps to reduce overfitting and makes the model more generally applicable and interpretable.



To optimize the performance of our Random Forest classifier, we used Grid search and cross validation to systematically search over a defined set of hyperparameter values. Specifically, we tuned the number of trees (n_estimators), the maximum depth of each tree (max_depth), and the minimum number of samples required to split an internal node (min_samples_split) and to be at a leaf node (min_samples_leaf). 5-fold cross-validation was also applied within the grid search to evaluate each hyperparameter combination and apply the highest average accuracy across folds. Tuning hyperparameters is essential for Random Forests because inappropriate settings can lead to underfitting or overfitting. Initially, we did not include a parameter to control for overfitting like min_samples_leaf, so training accuracy was 100%, accounting for all variability in the data. Selecting the best configuration and limiting the complexity of the model ensures the most accurate and stable predictions possible.

K-Nearest Neighbors

We implemented a K-Nearest Neighbors (KNN) Model to classify tumor malignancy based on similarity to other data points. KNN makes no assumptions about relationships in the data, unlike logistic regression, and with our dataset malignant and benign tend to cluster together due to their similar feature values, allowing for KNN to perform best. Within the cross-validation, forward selection combined with SelectKBest, was used to reduce dimensionality and overfitting while preserving interpretability. Grid search was also used to tune the number of neighbors (*k*), testing a range of odd values to avoid classification ties. Since KNN relies on distance calculations, we standardized all input features to ensure equal contribution across the selected attributes. The results of the cross-validation resulted in the optimal number of neighbors to be 6 along with the top 5 features chosen to be perimeter_mean, concave points_mean radius_worst, perimeter_worst, and concave points_worst. This is due to the fact that these features had the highest average feature importance, and 6 neighbors gave us the highest accuracy in our cross validation.

Results

To assess the performance of our final logistic regression, random forest, and K nearest neighbor's models, we used several evaluation metrics on both the training and test sets. After training the models we computed accuracy on the train and test sets. We then generated a confusion matrix for the test set to provide a more detailed breakdown of true positives, true negatives, false positives, and false negatives—helping us understand the types of errors the models make (Figure 1). All models had a similar misclassification rate and performed well on test data.

To further evaluate the model's classification performances, especially their ability to distinguish between classes, we plotted the Receiver Operating Characteristic (ROC) curve for the test set. The ROC curve shows the tradeoff between the true positive rate and the false positive rate at various classification thresholds. From this curve, we calculated the Area Under the Curve (AUC), which summarizes the model's ability to discriminate between the two classes: an AUC closer to 1 indicates better performance (Figure 2). Together, these evaluation methods give a comprehensive view of how well the model generalizes and performs on unseen data.

In general, we saw very little difference between our predictive model's accuracy. All three models showed accuracy around 96%. However, due to the sensitive context of our data, the best one would have the lowest rate of false negatives. Both KNN and logistic regression only falsely classified 1 malignant tumor as benign and thus both models had very low false negative rates (Figure 3).

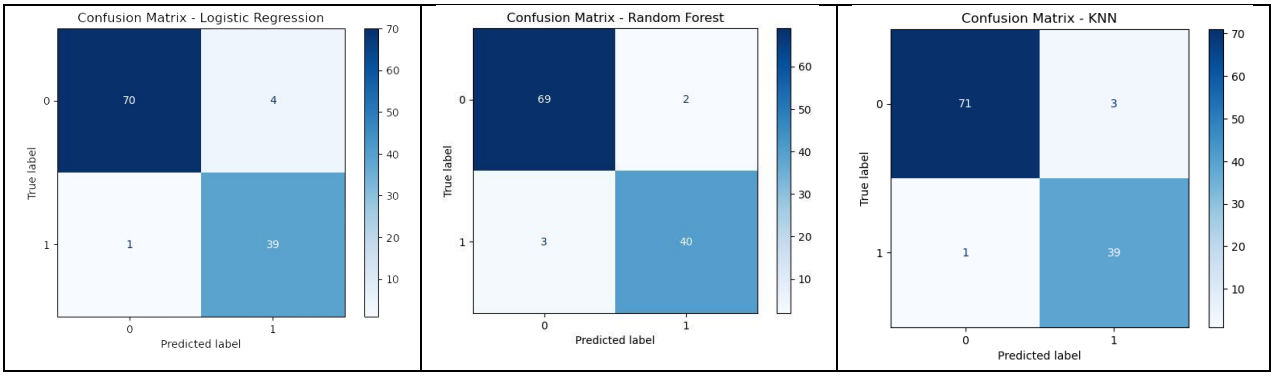
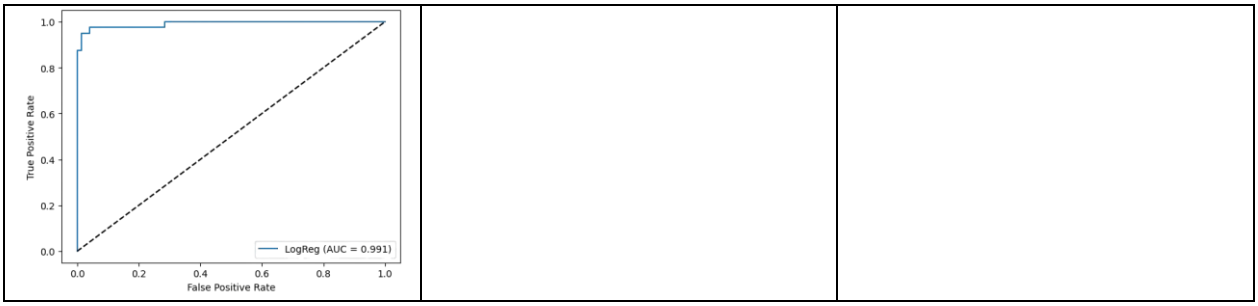


Figure 1: Confusion matrices between 3 models; logistic regression, random forest, and K nearest neighbors.



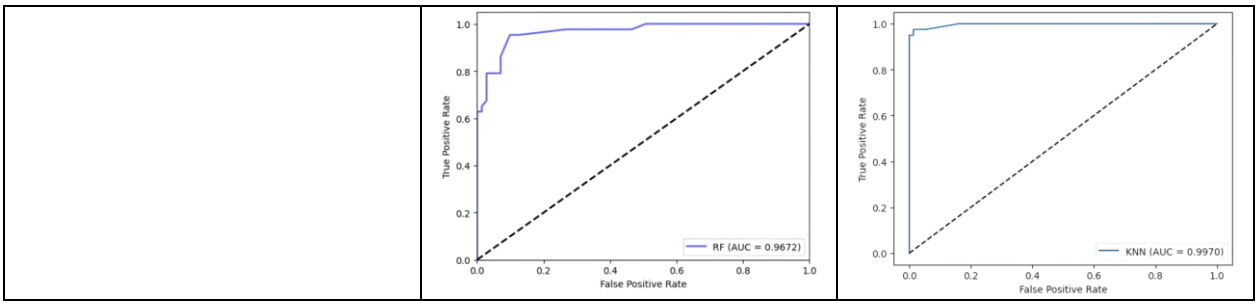


Figure 2: ROC curves including AUC for 3 models; logistic regression, random forest, k nearest neighbors.

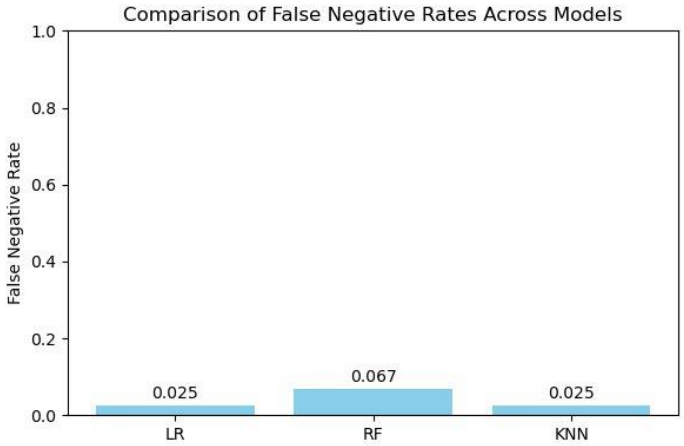


Figure 3: False negative rates across 3 models; logistic regression, random forest, and K nearest neighbors. Lower false negative rates are safer for real world applications

Discussion

Each of the models that were used performed well in detecting malignancy, showing the power that machine learning models have such that any of these models could be applied to hospitals and cancer centers throughout the world to improve timeliness of treatments. In addition to a doctor's opinion, inputting the information into a model would be a quick way for medical care providers to confirm their diagnosis, or take a closer look at the tumor if they disagree. In the future, a model like this could be linked to machines with cameras that could automatically measure the important features by just looking at a picture of the tumor or taking a scan in real time. One potential improvement if this problem were to be repeated would be to experiment with more complex models such as Gradient Boosting. The GBM could increase the number of accurately detected malignant tumors since it could capture some of the more intricacies of the data, along with input from a domain expert to confirm results.

If we were to apply these models in a real-world setting, we would choose the model with the highest recall for the malignant class, even if it leads to more false positives. In a sensitive field like medicine, it is better for a model to predict the worst case more, as long as it doesn't incorrectly predict the best case often or at all. This is to say, we would prefer the model to

classify more benign tumors as malignant as long as it doesn't classify many malignant tumors as benign. To account for this in the future when using classifying models, we could assign a higher penalty to false negatives in the model's loss function or during training. This might help to reduce the number of false negatives, increasing the safety of using the model, even if it may reduce accuracy.

References

Benign vs. malignant tumors. Baptist Health. (n.d.). https://www.baptisthealth.com/blog/cancer-care/benign-vs-malignant-tumors#:~:text=A%20tumor%20is%20an%20abnormal,and%20the%20tumor%20is%20malignant.

Centers for Disease Control and Prevention. (2024, October 25). *Leading Causes of Death*. Centers for Disease Control and Prevention. https://www.cdc.gov/nchs/fastats/leading-causes-of-death.htm

Simon, C. (2015, January 25). *Feature importance in random forests when features are correlated*. Mathemathinking. http://corysimon.github.io/articles/feature-importance-in-random-forests-when-features-are-correlated/