CS 7641 Machine Learning (ML)

Predicting Breast Cancer Tumor Malignancy Using Machine Learning Algorithms

Final Report

Haocheng Yu
Tianyi Yu
Yuxin Sun
Zihan Chen
Ziyi Zhou

July 03, 2024

Summer 2024
Instructor: Dr. Max Mahdi Roozbahani

Introduction/Background

Literature Review

Breast cancer is one of the most prevalent cancers affecting women worldwide. Accurate prediction of tumor malignancy can significantly enhance diagnosis and treatment planning. Previous studies have leveraged machine learning (ML) algorithms to improve malignancy prediction, demonstrating the potential of ML to surpass traditional statistical methods in accuracy and reliability. For instance, a study by Lu and Liu [1] explored various ML models, such as K-Nearest Neighbors and Gradient Boosting Models, to predict breast cancer survivability. Their results highlighted the importance of integrating clinical features with molecular data to improve prediction accuracy.

Additionally, research has shown that molecular diagnostics can complement traditional clinical tools, potentially enhancing the accuracy of breast cancer prognosis [2]. However, there is still no consensus on the most effective computational methods for predicting breast cancer outcomes, emphasizing the need for further exploration in this area [3].

Dataset Description

The dataset used in this project is obtained from Kaggle and is originally from the SEER Program of the National Cancer Institute (NCI). It includes records of female patients diagnosed with infiltrating duct and lobular carcinoma breast cancer between 2006 and 2010. The dataset contains 4024 patients after excluding those with incomplete records or survival times of less than one month. Key features include age, race, marital status, T stage, N stage, differentiation, grade, A stage, and tumor size.

Problem Definition

Problem

The primary challenge is to accurately classify breast tumors as malignant or benign using patient medical records and tumor characteristics. This prediction is crucial for early diagnosis and effective treatment planning.

Motivation

Early and precise identification of malignant tumors can significantly improve patient outcomes, guiding timely and appropriate medical interventions. The development of an accurate predictive model can assist healthcare professionals in making informed decisions, ultimately enhancing patient care and survival rates.

Methods

Data Preprocessing Methods

- **Handling Missing Values:** Missing values in numeric columns were filled with the mean of the respective columns, while missing values in categorical columns were filled with the mode. This approach ensures that the dataset is complete and ready for analysis without introducing significant biases.
- **Outlier Removal:** Outliers were identified and capped using the Interquartile Range (IQR) method. This helps in reducing the impact of extreme values on the model's performance.
- **Feature Encoding:** Categorical features were encoded using factorization, converting them into numeric values that can be used by machine learning algorithms.
- **Feature Scaling:** MinMaxScaler was applied to normalize feature values. This scaling technique ensures that all features contribute equally to the model training process, avoiding biases due to varying scales.

- **Correlation Analysis:** Correlation analysis was performed to identify the strength and direction of relationships between features and the target variable (status). A heatmap was used to visualize the correlation matrix. This step helps in understanding which features are strongly correlated with the target variable and can provide insights into feature selection and engineering. Highly correlated features with the target variable can be more impactful for the prediction model, while highly correlated features among themselves can cause multicollinearity issues.

Code for Data Preprocessing

```
## Check for missing values in the DataFrame
print("Missing Values in Each Column:")
print(df.isna().sum())

## Outlier Analysis
k = 1
plt.figure(figsize=(15, 10))
plt.suptitle("Distribution of Outliers")

for i in num_list:
    plt.subplot(2, 3, k)
    sns.boxplot(x=df[i])
    plt.title(i)
    k += 1
    plt.tight_layout()
plt.show()

# Identify and cap outliers
df_out = df.copy()
for i in num_list:
    Q1 = df[i].quantile(0.15)
    Q3 = df[i].quantile(0.85)
    IQR = Q3 - Q1
    upper = Q3 + 1.5 * IQR
    lower = Q1 - 1.5 * IQR
    df_out.loc[df[i] > upper, i] = upper
    df_out.loc[df[i] < lower, i] = lower

print("Statistical Summary After Outlier Removal:")
print(df_out.describe([0.1, 0.25, 0.35, 0.5, 0.65, 0.75, 0.9, 0.95]).T)

## Encode categorical features
for i in df.select_dtypes("object"):
    print(df[i].value_counts())

df.select_dtypes("object").head()

for i in cat_list:
    df[i] = df[i].factorize()[0]

for i in cat_list:
    df_out[i] = df_out[i].factorize()[0]

print("DataFrame After Encoding:")
print(df.head())
print(df_out.head())

## Correlation Analysis
plt.figure(figsize=(12, 12))
sns.heatmap(df.corr(), linewidths=0.6, cmap="coolwarm", annot=True, fmt=".2f")
plt.title("Correlation Heatmap")
plt.show()

# Extract features with high correlation relationships
cor = df.corr()["status"].sort_values(ascending=False)
print("Correlation with Status:")
print(pd.DataFrame({"column": cor.index, "Correlation with status": cor.values}))
```

```
58
59 # Drop columns with low correlation or high multicollinearity, if they exist
60 columns_to_drop = ["grade"]  # Add any other columns you want to drop
61 for col in columns_to_drop:
62     if col in df.columns:
63         df.drop(col, axis=1, inplace=True)
64         df_out.drop(col, axis=1, inplace=True)
65
66 ## Feature Scaling:
67 from sklearn.preprocessing import MinMaxScaler
68
69 # Apply MinMaxScaler to normalize feature values
70 scaler = MinMaxScaler()
71 X_scaled = scaler.fit_transform(df.drop("status", axis=1))
72 y = df["status"]
73
74 X_out_scaled = scaler.fit_transform(df_out.drop("status", axis=1))
75 y_out = df_out["status"]
76
77 print("Feature Scaling Completed")
```

Listing 1: Data Preprocessing Code

ML Algorithms/Models Implemented

For this project, we implemented and evaluated two supervised machine learning algorithms to predict breast cancer tumor malignancy. The chosen models and their explanations are as follows:

- **Random Forest Classifier:** Random Forest is an ensemble learning method that constructs multiple decision trees during training and outputs the mode of the classes for classification tasks. It was chosen because of its ability to handle a large number of features and provide robust performance even with unbalanced data. The algorithm is also less prone to overfitting compared to individual decision trees.
- **Logistic Regression:** Logistic Regression is a simple yet effective algorithm for binary classification problems. It provides probabilities for classification outcomes and is interpretable, making it easier to understand the impact of each feature on the prediction. It was chosen as a baseline model to compare more complex algorithms against.
- **K-Means Clustering:** K-Means Clustering is a popular unsupervised learning algorithm used for partitioning a dataset into distinct groups, or clusters. It aims to minimize the variance within each cluster, making the data points in a cluster as similar as possible to each other. This algorithm is efficient for large datasets and is easy to implement, making it suitable for exploratory data analysis. It was chosen to identify hidden patterns and groupings in the dataset, providing insights into potential categories or segments within the data.

Explanation of Methods and Model Selection

The combination of these preprocessing methods and machine learning algorithms aimed to build a reliable and accurate predictive model for breast cancer tumor malignancy, leveraging the strengths of each approach to address the complexities of the dataset.

- **Data Preprocessing:** Data preprocessing steps were crucial in preparing the dataset for model training. Handling missing values ensured that no data points were lost, while outlier removal helped in maintaining the integrity of the statistical distribution of features. Feature encoding converted categorical variables into a numeric format suitable for machine learning algorithms. Finally, feature scaling was necessary to ensure that all features contributed equally during model training.
- **Model Selection:** The models were chosen based on their strengths and suitability for the dataset characteristics:

- **Random Forest** was selected for its robustness and ability to handle large datasets with numerous features.
- **Logistic Regression** was used as a baseline to provide a point of comparison for more complex models.
- **K-Means Clustering** was used to identify and analyze inherent groupings within the data, providing insights into natural clusters and patterns.

Results and Discussion

Dataset analysis with K-Means

First, Elbow Method graph is utilized to choose the optimal number of clusters, which is 4.

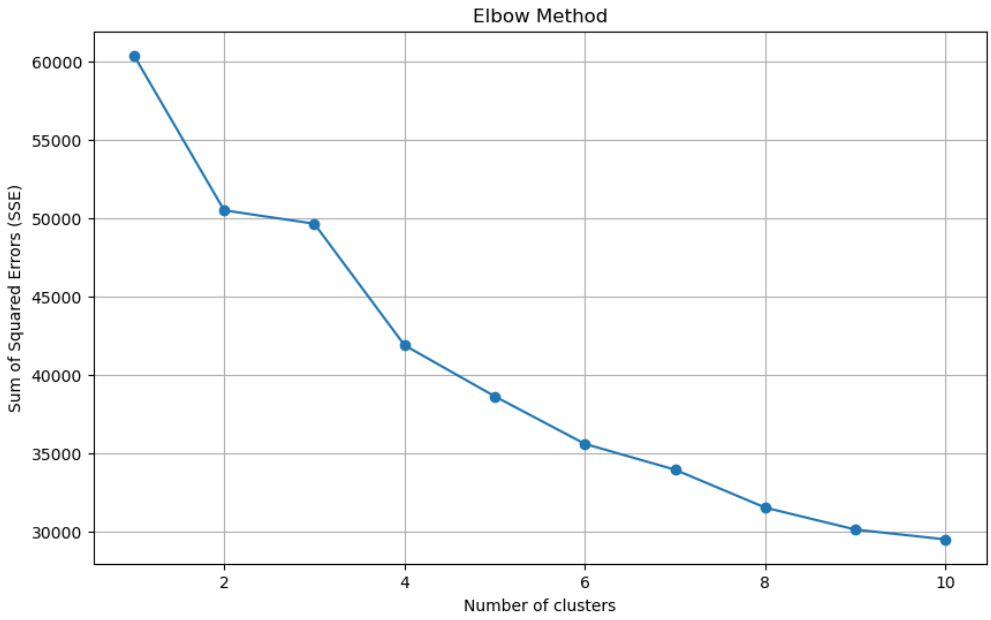

Figure 1: Elbow Method For Optimal Number of Clusters

Next, we applied K-Means clustering with 4 clusters to the full dataset, analyzed the characteristics of each cluster, and visualized the results using PCA with 2 principle components.

Then we analyzed the Characteristics for each cluster:

- **Cluster 0:**
  - Mean Age: 53.8
  - Higher proportion of well to moderately differentiated tumors.
  - High mean survival months (73.6).
  - Lower survival status (9.95% not survived).
- **Cluster 1:**
  - Mean Age: 51.9
  - Higher tumor stage and lymph node involvement.
  - Moderately differentiated tumors.
  - Lower mean survival months (60.2).

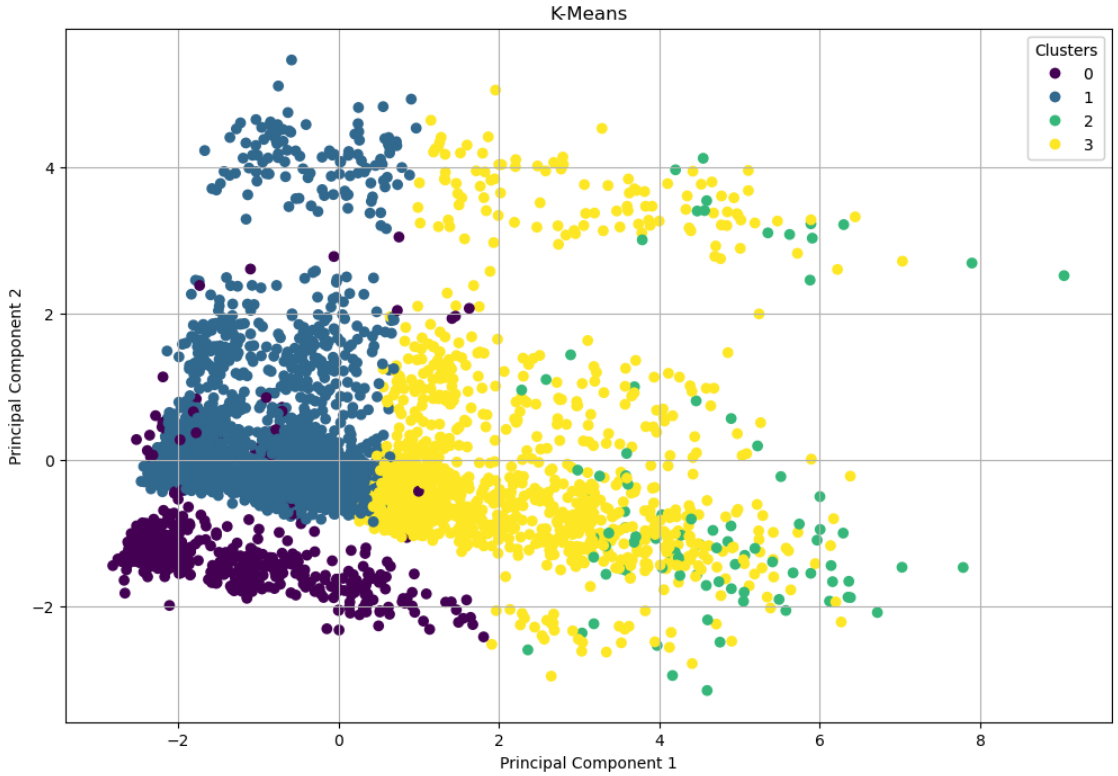

Figure 2: K-Means Clustering with 4 Clusters

  - Higher survival status (40.4% not survived).

- **Cluster 2:**
  - Mean Age: 55.2
  - Higher proportion of poorly differentiated tumors.
  - Moderately high mean survival months (73.9).
  - Higher survival status (5.7% not survived).

- **Cluster 3:**
  - Mean Age: 54.1
  - Higher tumor stage and lymph node involvement.
  - Moderately differentiated tumors.
  - Moderate mean survival months (68.5).
  - Higher survival status (23.5% not survived).

The clusters are distinct, with each cluster representing different patient profiles based on age, tumor characteristics, and survival outcomes. Clusters 1 and 3 have a higher proportion of patients with advanced tumor stages and lymph node involvement, resulting in lower survival rates. Clusters 0 and 2 show better differentiation and higher survival rates, indicating less aggressive cancer types.

Clusters with higher survival rates and better differentiation can be targeted for less aggressive treatments. On the other hand, clusters with lower survival rates and higher tumor stages may benefit from more intensive treatment and monitoring.

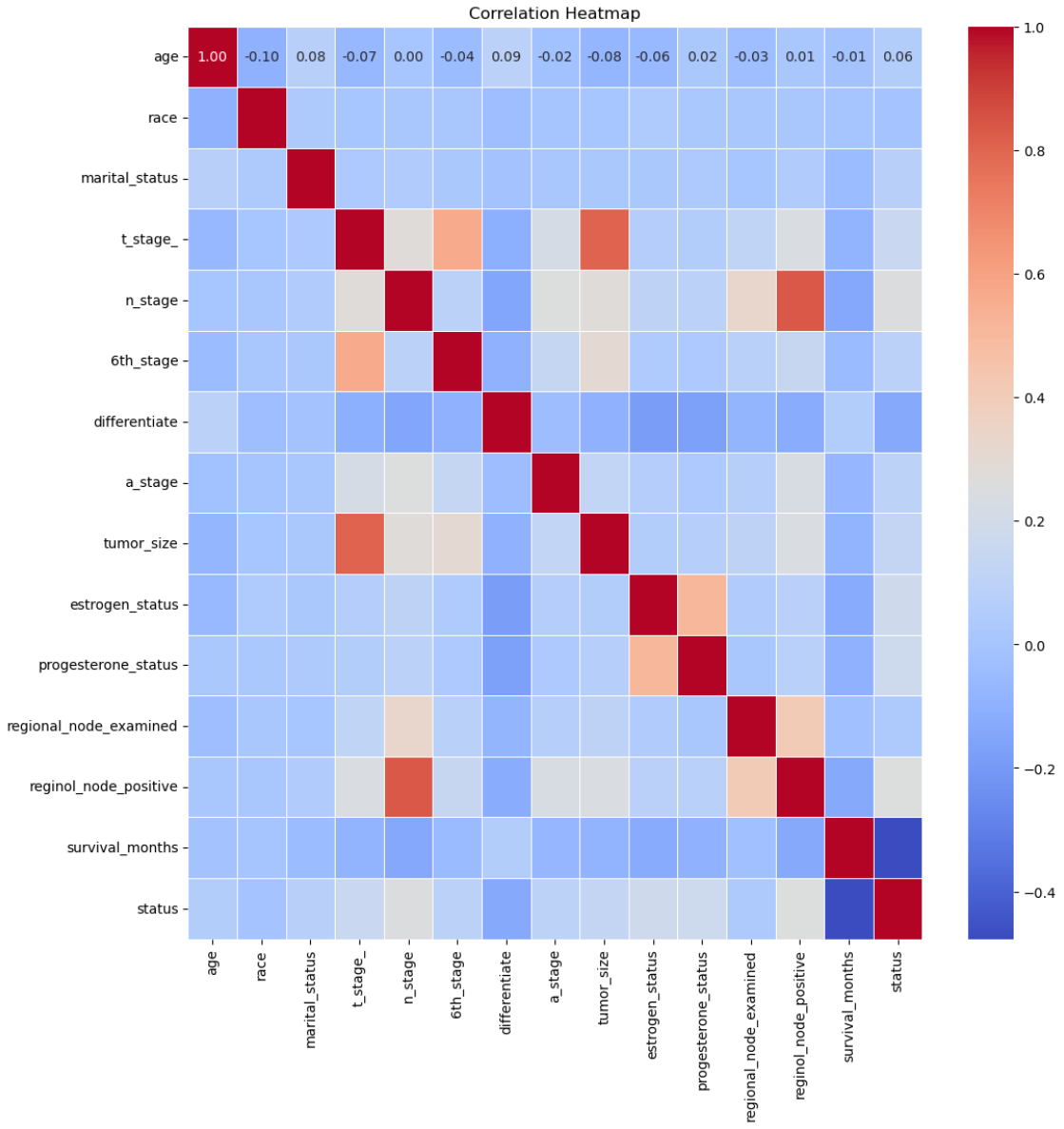

Figure 3: Confusion Matrix for Random Forest Classifier

Quantitative Metrics

- **Accuracy:** Measure of overall correctness of the model.
- **Precision:** Measure of the model's ability to correctly identify malignant tumors.
- **Recall:** Measure of the model's ability to detect all malignant tumors.
- **F1-Score:** Harmonic mean of precision and recall, providing a balance between the two.
- **ROC-AUC:** Area under the receiver operating characteristic curve, evaluating model performance across all classification thresholds.

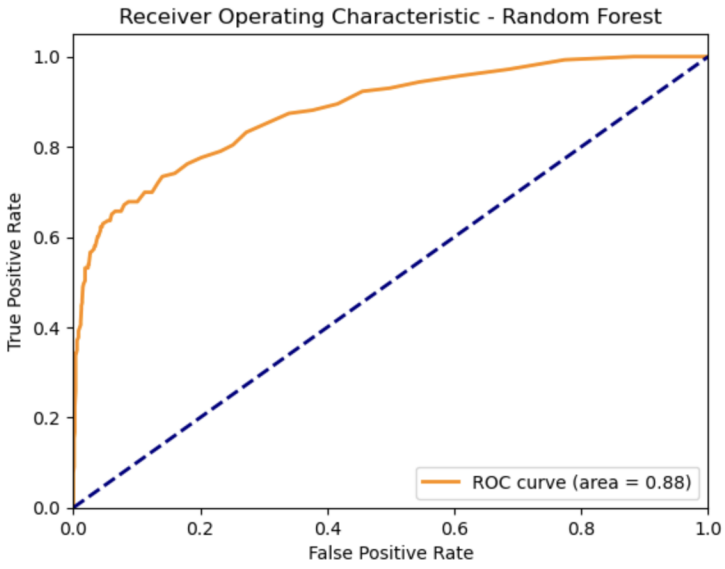

Figure 4: ROC-AUC Curve for Random Forest Classifier

Model Performance

The performance of various models on the test dataset is summarized below:

- **Logistic Regression:**
  - Accuracy: 0.9095
  - Precision: 0.8171
  - Recall: 0.4685
  - F1-Score: 0.5956
  - ROC-AUC: 0.7256
- **Random Forest:**
  - Accuracy: 0.9135
  - Precision: 0.7917
  - Recall: 0.5315
  - F1-Score: 0.6360
  - ROC-AUC: 0.7541

Visualizations

Analysis of Random Forest Classifier

The Random Forest Classifier showed the best overall performance among the models evaluated, with a high accuracy of 91.35%, a balanced F1-Score of 63.60%, and the highest ROC-AUC score of 75.41%. This indicates that the Random Forest model is robust and effective in predicting breast cancer tumor malignancy.

References

[1] C. Lu and J. Liu, "Breast Cancer Prognosis," Stanford University, 2012. Available: `https://cs229.stanford.edu/projects2012.html`.

[2] C. Curtis et al., "The genomic and transcriptomic architecture of 2,000 breast tumours reveals novel subgroups," Nature, 2012.

[3] S. Paik et al., "A multigene assay to predict recurrence of tamoxifen-treated, node-negative breast cancer," N. Engl. J. Med., vol. 351, no. 27, pp. 2817-2826, Dec. 2004.

Gantt Chart

TASK TITLE	TASK OWNER	START DATE	DUE DATE
Project Team Composition	All	6/20/2024	7/3/2024
Midterm Contributions			

Contribution Table

Name	Final Report
Haocheng Yu	
Tianyi Yu	
Yuxin Sun	
Zihan Chen	
Ziyi Zhou	