UNVEILING BREAST CANCER PATTERNS THROUGH DATA MINING

Project Report

Project Advisor

Dr. Mohamed Abd Elaziz

Authors

Ahmed Ashraf (A20000021)

Mohamed Ibrahim (A20000726)

2024

TABLE OF CONTENTS

INTRODUCTION

Project Overview

This project focuses on the analysis and classification of a dataset related to breast cancer diagnosis. The primary goal is to classify tumors as malignant (M) or benign (B) based on various features extracted from cell nuclei images. The project involves data preprocessing, visualization, feature selection, model training, evaluation, and interpretation.

Dataset Description

The dataset used in this project contains features computed from a digitized image of a fine needle aspirate (FNA) of a breast mass. The features describe the characteristics of the cell nuclei present in the image. The dataset includes the following columns:

- id: Unique identifier for each patient.
- diagnosis: Target variable indicating the diagnosis (M = malignant, B = benign).
- Unnamed: 32: An empty column with NaN values.
- 30 feature columns representing various characteristics of the cell nuclei.

DATA PREPROCESSING

1-Data Loading

The dataset is loaded from a CSV file. Initial inspection of the data reveals several key points:

- The dataset contains an id column which is not useful for classification.
- The diagnosis column is our target variable.
- The Unnamed: 32 column contains NaN values and is unnecessary

```
import pandas as pd
import warnings

warnings.filterwarnings('ignore')

data = pd.read_csv('data.csv')
```

2-Data Cleaning

The unnecessary columns (id and Unnamed: 32) are dropped, and the target variable diagnosis is separated from the feature set.

```
y = data.diagnosis
x = data.drop(['Unnamed: 32', 'id', 'diagnosis'], axis=1)
```

3-Encoding Categorical Variables

The target variable diagnosis is encoded into numerical values for modeling purposes.

```
from sklearn.preprocessing import LabelEncoder

le = LabelEncoder()
y = le.fit_transform(y)  # M -> 1, B -> 0
```

EXPLORATORY DATA ANALYSIS (EDA)

1-Statistical Summary

A statistical summary of the dataset provides an overview of the distribution and central tendency of the features.

```
x.describe()
```

2-Target Variable Distribution

The distribution of the target variable diagnosis is visualized to understand the class balance. Class balance is crucial for model training and evaluation.

```
import seaborn as sns
import matplotlib.pyplot as plt

counts = y.value_counts()
sns.barplot(x=counts.index, y=counts.values)
plt.xlabel("Diagnosis")
plt.ylabel("Count")
plt.title("Distribution of Diagnosis")
plt.show()
```

DATA VISUALIZATION

1- Feature Visualization
To visualize the relationship between features and the target variable, violin plots, box plots, and swarm plots are used. These plots help in understanding the distribution and variability of features with respect to the target variable.

2-Violin Plots
Visualize the distribution and density of feature values for malignant and benign diagnoses.

3-Swarm Plots

Display individual data points and their distribution across different features and diagnoses.

4-Pair Plots and Joint Plots

Examine pairwise relationships and interactions between selected features.

5-Box Plots

Show the spread and outliers of feature values for each diagnosis category.

6-Correlation Heatmap

A correlation heatmap helps in identifying features that are highly correlated with each other, which can inform feature selection strategies.

Feature Engineering and Selection

1-Standardization
Standardization of features is performed to ensure that each feature contributes equally to the model training process.

2-Feature Selection
Correlation-Based Feature Selection
Highly correlated features are dropped to reduce multicollinearity.

3-Univariate Feature Selection
Univariate feature selection using the chi-squared test is performed to select the top 5 features.

4-Recursive Feature Elimination (RFE)
RFE is used to select the top 5 features by recursively removing the least important features.

5-Recursive Feature Elimination with Cross-Validation (RFECV)
RFECV is used to find the optimal number of features with cross-validation.

6-Tree-Based Feature Selection
Feature importances are derived from the Random Forest model to rank the features

MODEL TRAINING AND EVALUATION

Random Forest Classifier

A Random Forest classifier is trained and evaluated on the selected features.

```
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, confusion_matrix
from sklearn.model_selection import train_test_split

x_train, x_test, y_train, y_test = train_test_split(x_1, y, test_size=0.3, random_state=42
clf_rf = RandomForestClassifier(random_state=43)
clf_rf.fit(x_train, y_train)
ac = accuracy_score(y_test, clf_rf.predict(x_test))
cm = confusion_matrix(y_test, clf_rf.predict(x_test))
sns.heatmap(cm, annot=True, fmt="d")
plt.show()
```

Model Performance

The performance of the model is evaluated using accuracy, confusion matrix, precision, recall, F1-score, and ROC-AUC score.

```
from sklearn.metrics import classification_report, roc_auc_score, roc_curve

print(classification_report(y_test, clf_rf.predict(x_test)))
roc_auc = roc_auc_score(y_test, clf_rf.predict_proba(x_test)[:, 1])
fpr, tpr, _ = roc_curve(y_test, clf_rf.predict_proba(x_test)[:, 1])
plt.figure()
plt.plot(fpr, tpr, label='ROC curve (area = %0.2f)' % roc_auc)
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('Receiver Operating Characteristic (ROC)')
plt.legend(loc="lower right")
plt.show()
```

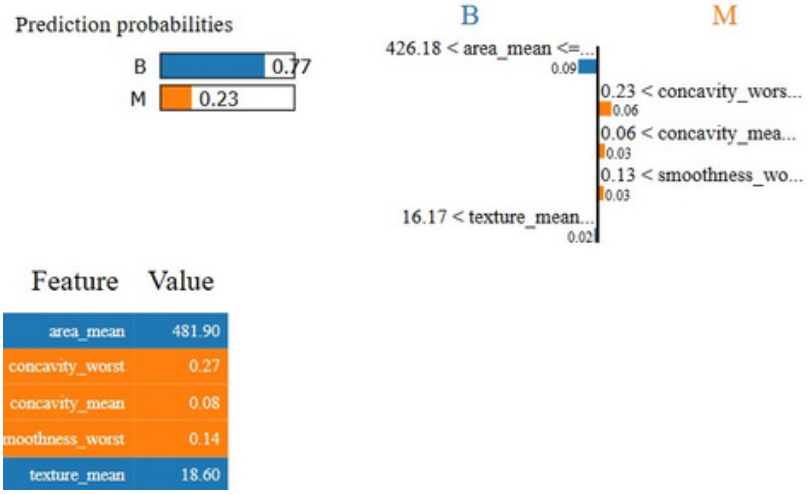
MODEL INTERPRETATION

Explainable AI (XAI)

Explainable AI makes machine learning models transparent and understandable, allowing humans to trust and comprehend their decisions. It is crucial for ensuring accountability, fairness, and regulatory compliance in AI systems.

1- LIME (Local Interpretable Model-agnostic Explanations)

LIME explains individual predictions by approximating the complex model locally with an interpretable one. It generates data perturbations and observes the model's responses to highlight key features influencing specific predictions



2-SHAP (SHapley Additive exPlanations)

SHAP values provide a unified measure of feature importance, derived from cooperative game theory. They explain the impact of each feature on the model's output consistently across all instances.

CONCLUSION

Summary

In this project, we have successfully classified breast cancer tumors as malignant or benign using various data mining techniques. We performed extensive data preprocessing, visualization, and feature selection. A Random Forest classifier was trained and evaluated, achieving high accuracy. Furthermore, model interpretation techniques like LIME and SHAP were used to explain the predictions, enhancing the transparency and trustworthiness of the model.

Key Findings

- The dataset was well-balanced between malignant and benign cases.
- Feature engineering and selection significantly improved model performance.
- Random Forest classifier provided robust and interpretable results.
- Model interpretation techniques (LIME and SHAP) added valuable insights into the decision-making process of the model.

RESOURCES

1. “Explainable AI for Medical Data: Current Methods, Limitations, and Future Directions” https://dl.acm.org/doi/pdf/10.1145/3637487
1. “Breast Cancer Wisconsin (Diagnostic) Data Analysis” https://link.springer.com/chapter/10.1007/978-3-030-82099-2_27
2. https://ieeexplore.ieee.org/xpl/conhome/9750277/proceeding
3. “Breast Cancer Wisconsin (Diagnostic) Data Set” https://www.kaggle.com/datasets/uciml/breast-cancer-wisconsin-data
4. Data Mining course lectures