ECE 595 Project: Breast Cancer Detection

Mohit Pandiya

December 12, 2023

1 Introduction

Breast cancer poses a significant health threat, underscoring the urgency for precise and timely detection. Digitized fine needle aspirate (FNA) images of breast mass are used to detect the presence of a tumor.

The objective of this project is to employ classification techniques covered in the course to distinguish between benign and malignant samples. Additionally, the project aims to assess and compare the performance of these classification methods.

2 Formulation

The above task is formulated as a **Classification** task. This task involves utilizing various features, including but not limited to radius_mean, texture_mean, perimeter_mean, area_mean, smoothness_mean, compactness_mean, concavity_mean, concave points_mean, symmetry_mean, fractal_dimension_mean, and many others. The primary goal is to classify the set of features as benign and malignant. Additionally, the project aims to compare different classification models and their performance on the whole set of features and also some reduced number of features.

3 Dataset

The dataset comprises features derived from a digitized image of a fine needle aspirate (FNA) of a breast mass, specifically detailing the attributes of cell nuclei in the image. With a total of 596 rows and 30 labels, the data has undergone preprocessing and contains no missing values. The only preprocessing required on the data is scaling. Since the labels span different ranges as shown in 1, scaling is required to bring all of them between $[0, 1]$

```
df.describe()
```
✓ 0.2s

	id	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean
count	5.690000e+02	569.000000	569.000000	569.000000	569.000000	569.000000	569.000000	569.000000
mean	3.037183e+07	14.127292	19.289649	91.969033	654.889104	0.096360	0.104341	0.088799
std	1.250206e+08	3.524049	4.301036	24.298981	351.914129	0.014064	0.052813	0.079720
min	8.670000e+03	6.981000	9.710000	43.790000	143.500000	0.052630	0.019380	0.000000
25%	8.692180e+05	11.700000	16.170000	75.170000	420.300000	0.086370	0.064920	0.029560
50%	9.060240e+05	13.370000	18.840000	86.240000	551.100000	0.095870	0.092630	0.061540
75%	8.813129e+06	15.780000	21.800000	104.100000	782.700000	0.105300	0.130400	0.130700
max	9.113205e+08	28.110000	39.280000	188.500000	2501.000000	0.163400	0.345400	0.426800

Figure 1: Statistics of some features

Given the abundance of features, the project necessitates some form of feature selection or dimension reduction. To facilitate project evaluation, the dataset has been partitioned into training and testing data, with 30% of the records reserved for testing purposes.

Figure 2 shows a histogram of all the features separated by color for different classes. From this, we can see that not all classes are equally important for classification. This adds another task for feature selection.

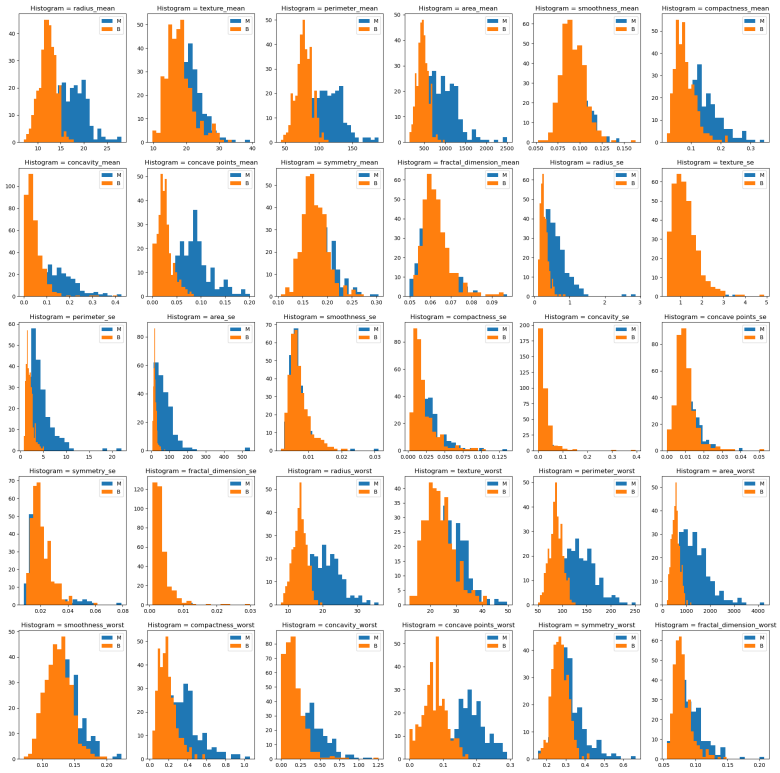

Figure 2: Distribution of all features separated by class

I select three features that show and compare classification results done on them with classification done on the whole data set and also 3 PCA features. The 3 features I select from this graph are *cocave_points_mean, radius_mean and area_mean*

The dataset is hosted on UC irvine machine learning repository. Link

4 Algorithm

I have applied 4 data mining algorithms on 3 features.

- All the features
- 3 features extracted manually by looking at the distribution
- 3 features extracted using PCA

The algorithms used to perform classification are -

- **Logistic Regression** -Logistic Regression is a statistical technique commonly used for binary classification tasks. It models the probability $P(Y = 1)$ of an event occurring as a function of one or more independent variables $X_1, X_2, \ldots, X_n$. The logistic function, or sigmoid function, ensures the output is bounded between 0 and 1. The logistic regression model is represented by the equation:

$$P(Y = 1) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_n X_n)}}$$

  In this equation, β_0 is the intercept, and $\beta_1, \beta_2, \ldots, \beta_n$ are the coefficients associated with the independent variables. The model is trained by maximizing the likelihood function, and the resulting coefficients are used to predict the probability of the event for new observations.

- **Random Forest** - Random Forest is an ensemble learning algorithm widely used for both classification and regression tasks. It operates by constructing a multitude of decision trees during training and outputs the class that is the mode of the classes (classification) or the mean prediction (regression) of the individual trees. Each tree in the Random Forest is trained on a random subset of the training data, and a random subset of features is considered at each split. The final prediction is a combination of the predictions from all the trees in the forest. For classification, the mode of the individual tree predictions is taken, and for regression, the mean of the individual tree predictions is used.

Let T represent the number of trees in the forest, and $h_i(\mathbf{x})$ denote the prediction of the i-th tree for input $\mathbf{x}$. The final prediction of the Random Forest for classification and regression is given by:

$$\text{Classification: } \hat{Y} = \text{mode}\{h_i(\mathbf{x})\}_{i=1}^{T}$$

Here, $\hat{Y}$ represents the predicted output, and the combination of multiple trees helps improve generalization and reduce overfitting.

- **Gaussian Naive Bayes** - Gaussian Naive Bayes is a probabilistic classification algorithm based on Bayes' theorem and the assumption of independence between features. It is particularly effective when dealing with continuous data. The model is named "naive" because it assumes that the features are conditionally independent given the class label. For a classification problem with C classes and n features, the probability of an observation $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ belonging to class c is calculated using Bayes' theorem as follows:

$$P(Y = c|\mathbf{x}) = \frac{P(Y = c) \cdot P(x_1|Y = c) \cdot P(x_2|Y = c) \cdot \ldots \cdot P(x_n|Y = c)}{P(\mathbf{x})}$$

Here, Y represents the class variable, and $P(Y = c)$ is the prior probability of class c. $P(x_i|Y = c)$ represents the likelihood of observing feature x_i given the class c, assuming a Gaussian (normal) distribution:

$$P(x_i|Y = c) = \frac{1}{\sqrt{2\pi\sigma_{c,i}^2}} e^{-\frac{(x_i - \mu_{c,i})^2}{2\sigma_{c,i}^2}}$$

In this equation, $\mu_{c,i}$ and $\sigma_{c,i}$ are the mean and standard deviation of feature x_i for class c, respectively. The final class prediction is the one with the highest probability.

5 Experiments

In the experiment, all classification models were trained on the training dataset, constituting 70% of the original data which is 398 rows. The model performance is evaluated on the remaining data. The training process was conducted three times, utilizing all features, manually selected features, and features reduced through PCA. The results of these models are presented in 1

Model	Features	Accuracy	F1-score	Time
Logistic Regression	All	88.89	0.8950	0.0040
	PCA	95.32	0.9643	0.0040
	Manually selected	84.21	0.8643	0.0046
Random Forest	All	92.40	0.9372	0.3062
	PCA	93.57	0.9479	0.2050
	Manually selected	87.13	0.8932	0.1556
Gaussian Naive Bayes	All	91.81	0.9314	0.0020
	PCA	90.64	0.9279	0.0010
	Manually selected	91.23	0.9268	0.0020

Table 1: Evaluation of different models

6 Comparison

Several insights can be gleaned from the performance of the various models. The Logistic Regression model achieved the highest accuracy and F1-score when utilizing the PCA features. This success can be attributed to the ability of PCA to capture the high variance axis of the dataset which is not something that the manually selected features have.

In the case of the Random Forest classifier, the model performed better when using all features compared to a subset of features. This behavior may be attributed to the nature of Random Forest, which consists of decision trees. Decision trees, being rule-based models, benefit from having more features to create better rules for classification. With only three features selected manually, the tree might have to create numerous nodes to capture the complexity of the data.

The Naive Bayes classifier demonstrated good performance on the dataset, with the advantage of fast fitting during training. Due to its probabilistic classification approach, much of the computation in Naive Bayes occurs during the inference phase rather than the training phase. This characteristic contributes to it having almost no training times recorded.

In summary, the success of each model is influenced by the feature selection strategy and the inherent characteristics of the algorithms. PCA, considering the highest variance features, proved effective for Logistic Regression. In contrast, Random Forest benefited from using all available features.

7 Challenges

The biggest challenge in using machine learning for medical use cases is that the False negative rate should be 0, i.e. the predicted value cannot be benign when

it is malignant, which could be bad for the patient. This could be tackled using more data and robust machine learning algorithms.

Project Files : link