Breast Cancer Diagnosis Classification

Yiling Peng (yp387)
Shuhan Ding (sd925)

I. INTRODUCTION

Cancer is one of the major killers threatening human health and life, and the breast cancer has the highest incidence and mortality rate worldwide [1]. Breast Cancer occurs as a result of abnormal growth of cells in the breast tissue, commonly referred to as a Tumor. A tumor does not mean cancer - tumors can be benign (not cancerous), or malignant (cancerous). It is of great significance to identify novel and effective strategies for breast cancer diagnosis. In recent years, artificial intelligence (AI) technologies have played significant roles in the clinical care of breast cancer, providing new approaches for clinicians to identify high-risk patients. This project aims to observe which features are most helpful in predicting malignant or benign cancer and to see general trends that may aid us in model selection and hyper parameter selection. The goal is to classify whether the breast cancer is benign or malignant, assisting clinicians in diagnosis.

The dataset is from the Wisconsin Breast Cancer Diagnostic Dataset [2]. Medical personnel acquired digital images of patients' breast lumps after fine needle aspiration (FNA) and extracted features from these digital images that describe the presentation of cell nuclei. Tumors can be classified as benign or malignant.

The dataset includes a total of 32 attributes, representing as following：

1) ID number;

2) Diagnosis (M = malignant, B = benign)

Ten real-valued features are computed for each cell nucleus:

a) radius; b) texture; c) perimeter; d) area; e) smoothness; f) compactness; g) concavity; h) concave points; i) symmetry; j) fractal dimension.

The mean, standard deviation (SE), and maximum values-worst (mean of the three largest values) of these ten features are calculated for each sample separately.

Therefore, before data cleaning, we have an entire 569*32-dimensional data. Diagnosis result is treated as our perdition label and the rest of the features, which we analyzed and processed, are used to predict whether the tumor is benign or malignant. Our goal is to generate the final feature vectors by performing PCA, and then test and compare the models, expecting to achieve great prediction performance.

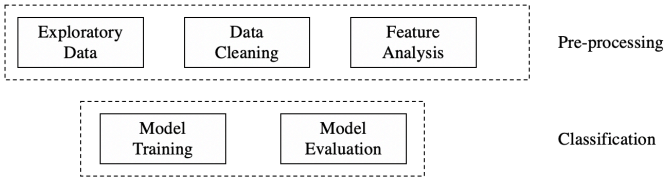

Figure 1. General structure of this project.

The structure of the project is as follows:

The 1st chapter introduces the background and significance of the project research, and gives the main content and structure of this project.

The 2nd chapter is devoted to the pre-processing of the raw data, including checking missing values and mapping variables. A preliminary exploration and selection of data is also performed.

The 3rd chapter mainly implements the classification of benign-malignant breast cancer after feature selection. The classification algorithms used are Random Forest, Support Vector Machine, Logistic Regression, Light GBM and XG BOOST.

The 4th chapter summarizes and discusses the experimental results of this article. Improvements and deficiencies are identified, also we give suggestions for further research.

II. DATA PROCESSING

2.1 Check missing data

First, we filter out the attributes we need. Since “id” is a meaningless feature for our prediction, it is removed.

Next, we check if there are still missing values. We found that in this dataset, no missing values exist. Therefore, the final size of data is 569*31.

2.2 Map data label

We standardized non-numeric and numeric label values. In particular, in original dataset, malignant tumors were recorded as M and benign as B. For binary classification, we converted diagnostics parameters to values, 0 for benign and 1 for malignant. It is noted that in 569 patients, 357 were benign and 212 were malignant.

2.3 Data Analysis for the entire dataset

2.3.1 Explore the Mean/SE/Worst Data

We first use the type of data as a reference for classification, and we analyze the mean, SE and worst data for all features separately.

For mean and worst data (Figure 2 and 3), we can find that the features form more distinct clusters, which are worthy of further analysis. The values of cell radius, perimeter, area, compactness, depression and dimple can be used for cancer classification. Larger values of these parameters tend to show a correlation with malignancy. On the other hand, the values of texture, smoothness, symmetry or fracture dimension did not show a particular preference for one diagnosis. Moreover, area, radius and perimeter are strongly correlated and in subsequent processing we may need to select only one of these variables to participate in our classification. As for the SE data (Figure 4), we found that the clusters of benign and malignant almost overlap and are not quite separable, so further processing may be required.

Overall, almost perfectly linear patterns between the radius, perimeter, and area attributes are hinting at the presence of multicollinearity between these variables. Another set of variables that possibly imply multicollinearity are the concavity, concave_points, and compactness.

2.3.2 Data Analysis for Benign/Malignant Tumor

In addition to the analysis of each data category, we then explored the data separately for the different types of benign and malignant tumors.

First, we performed the analysis for benign tumors. We selected the records of the mean of all patients with benign tumors and computed the mean, minimum and maximum values. Similarly, we analyzed worst cases to obtain the actual and optimized values. Then, we perform the same operation for the malignant tumor data. As shown in the figure, we use the radar plot to compare the relationship between the optimized and actual values for malignant and benign under worse and mean records.

Under the mean record, we found that for patients with benign tumors, the data for all ten characteristics were smaller than the average, especially for concave_points and perimeter, and this observation was also confirmed in patients with malignant tumors. Therefore, concave_points and perimeter are the key influencing factors in this condition.

It is found that under worse cases, patients with benign tumors have higher values in symmetry than average, smoothness and texture are not significantly

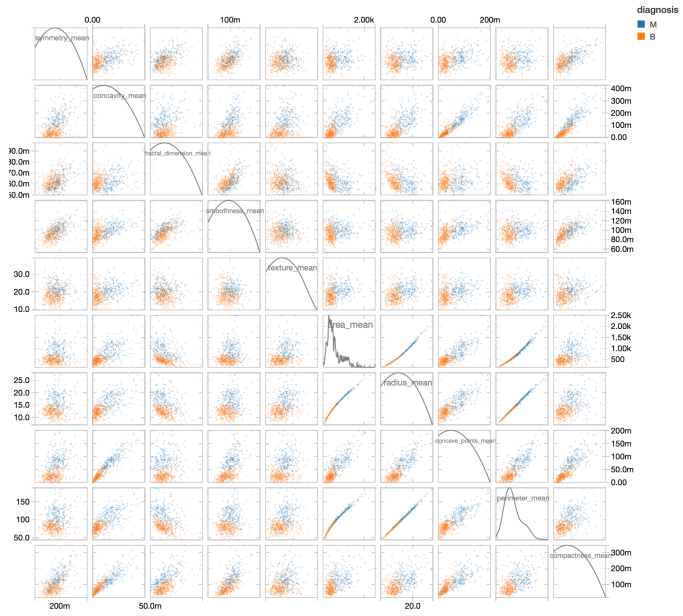

Figure 2. Plotpairs of mean data.

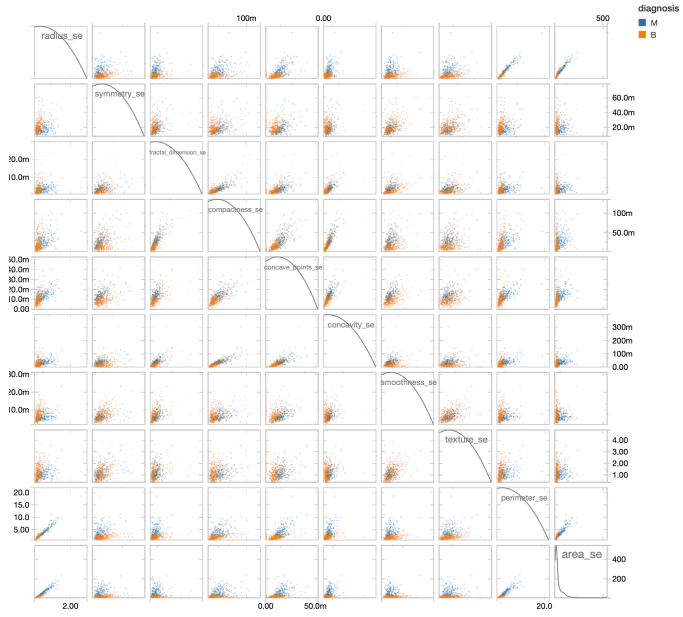

Figure 3. Plotpairs of SE data.

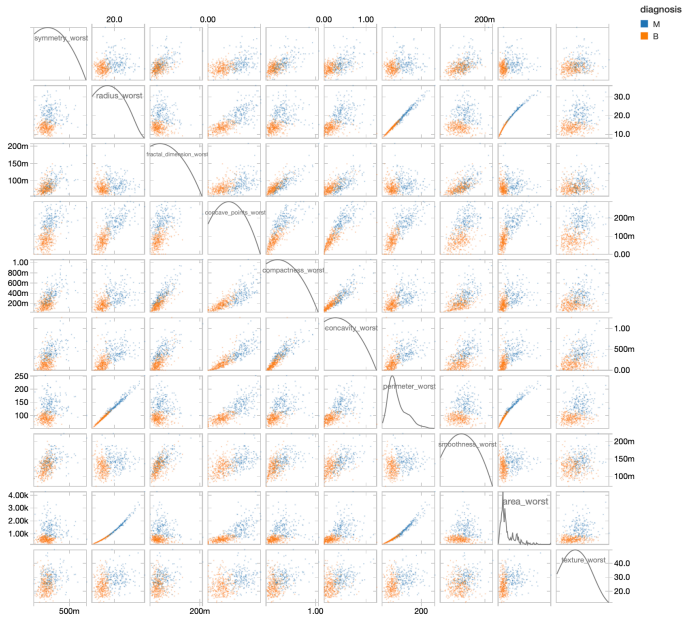

Figure 4. Plotpairs of worst data.

different from average, while the rest of the features have smaller values for patients with benign tumors. For patients with malignant tumors, the most significant feature is concave_points, which is twice higher than the average and is a potentially critical factor affecting malignancy.

2.4 Principal Component Analysis

The concept of principal component analysis (PCA) was introduced by Karl Parson in the study of regression analysis in 1901 [19], but only the case of non-random variables was studied at that time. In 1933 Hotelling extended this concept to random variables [20]. PCA retains the features with large variance contribution in the data set, while discarding features with little impact on the original data. This not only achieves dimension reduction, but also tries to avoid large errors when reconstructing the original data set. PCA did quite a decent job of visualizing our two target clusters. Although PCA is able to differentiate the classes very well, we found later that PCA technique also reduced some important features, which resulted in reduced accuracy. Therefore, we only used standard scaler on the dataset.

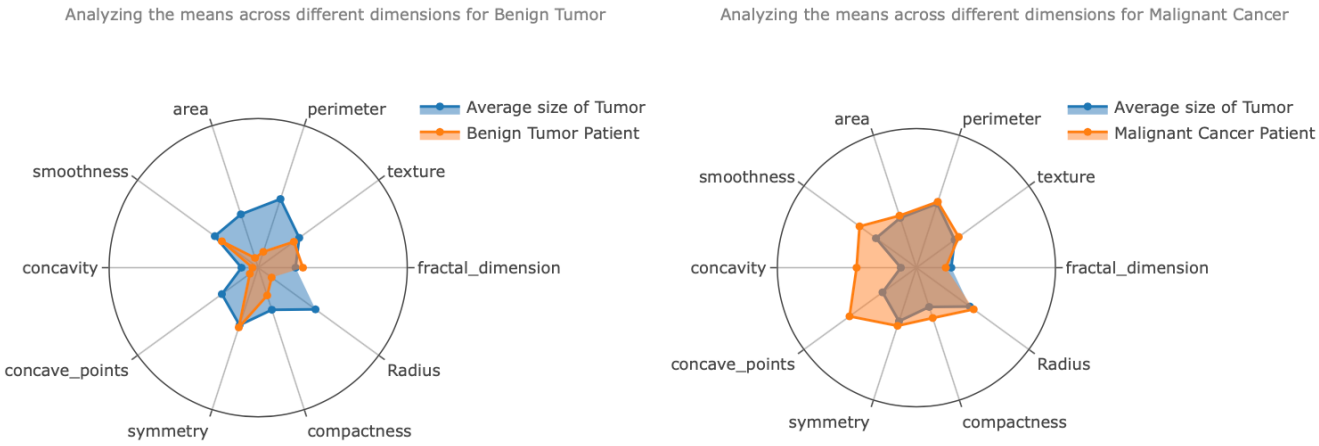

Figure 5. Radar graph to determine optimum vs actual cell type for means.

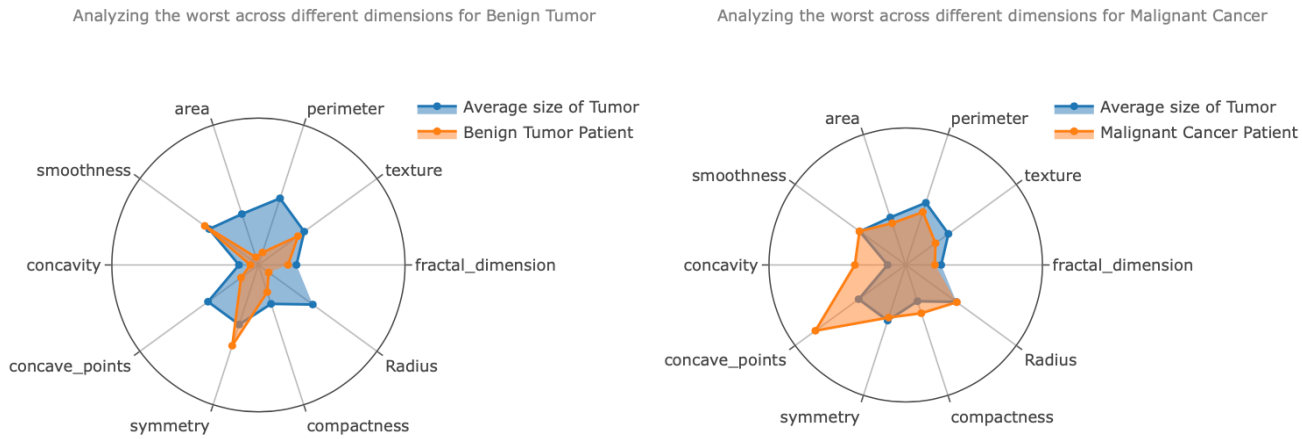

Figure 6. Radar graph to determine optimum vs actual cell type for worst attributes.

III. CLASSIFICATION

3.1 Splitting the data set

Since this dataset is not ordered, we make a simple 8:2 split of the dataset to create a training dataset and a test dataset. And we also 8:2 split the training dataset into a training and validation dataframe.

3.2 Classification model

We try five different classification models. The dataset used for training and testing these models is scaled to unit standard deviation. We choose the standard scaled dataset rather than the new features generated and selected by the PCA algorithm, to keep more important features.

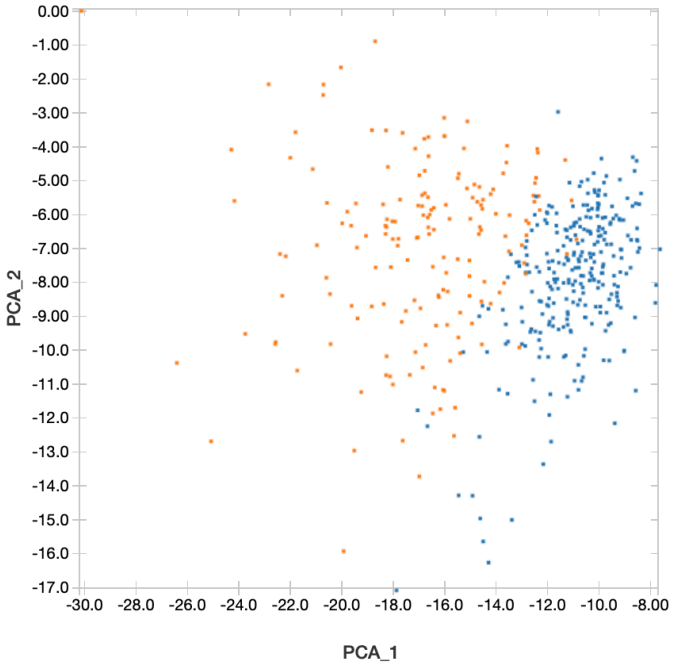

Figure 7. PCA of two clusters (yellow for Malignant and blue for Benign).

3.2.1 Random Forest

Random Forest is an ensemble learning method. It consists of many classification trees. When classifying a new object from an input vector, put the input vector down each of the trees in the forest. Each tree gives a classification result and the tree "votes" for that class. The forest chooses the classification having the most votes as the final result.

We use the “RandomForestClassificationModel” in the spark-ml library to realize the random forest [3]. In our model, 20 classification trees form a random forest. For each tree, the maximum depth is 5 and the criterion used for information gain calculation is Gini. We don’t limit the number of features to consider for splits at each tree node. Considering that the samples and features in the dataset are too limited, both the number of trees and the depth of the tree is small. If there are more trees than required, there will be duplicate trees. Whereas the complexity of the tree structure increases with depth, it’s more possible for deeper trees to overfit the small dataset.

3.2.2 Support Vector Classification

A support vector machine (SVM) is a supervised learning method [4]. When it is used for solving binary classification problems, it’s known as Support Vector Classification (SVC). SVC maps the feature vector of an instance to some points in space. The purpose of an SVC is to draw a line or hyperplane that "best" distinguishes between these two types of points so that if new points become available later, the line can also make a good classification. SVC is suitable for small-sized or medium-sized data samples, non-linear, high dimensional classification problems, and this is the kind of data set we need to classify.

We use the “LinearSVCModel” in the spark-ml library to realize the support vector classification [3]. When training the model, we set the maximum number of iterations to 100 without using regularization considering the size of the dataset.

3.2.3 Logistic Regression

The logistic model is used to model the probability of a sample belonging to a certain class [5]. Logistic regression is a statistical model that in its basic form uses a logistic function to model a binary dependent variable. Mathematically, a binary logistic model has a dependent variable with two possible values, such as yes/no which is represented by an indicator variable, where the two values are labeled "0" and "1". We also need to set a threshold to convert the probability to the label “0” or “1”.

We use the “LogisticRegressionModel” in the spark-ml library to realize the logistic regression [3]. During training, the loss function is L2 penalty and the threshold in binary classification prediction is set to 0.5, which is widely used. The maximum number of iterations is 100, which could prevent overfitting.

3.2.4 XG BOOST

XGBoost is an open-source software library which provides a regularizing gradient boosting framework [6]. The XG BOOST algorithm uses the decision tree ensembles. The random forest is also in the category of ensemble learning, but it’s bagging model where each tree is trained independently and the results of the previous

tree do not affect the training of the new tree. Gradient boosting is different in that it adds new weak learners to try to correct the residuals of all the previous weak learners, and eventually the sum of the multiple learners is used to make the final prediction with higher accuracy than one alone. It is called Gradient because of the gradient descent algorithm used to minimize the loss when adding new models.

The XGBoost has a number of advantages. Firstly, it adds a regular term to the cost function to control the complexity of the model. The regularity term contains the number of leaf nodes in the tree, the sum of squares of the L2 modes of the output scores at each leaf node. In terms of Bias-variance tradeoff, the regularity term reduces the variance of the model, making the learned model simpler and preventing overfitting, which is a feature that makes XGBoost better than traditional GBDT. In additions, the XGBoost allows cross-validation to be used in each boosting iteration. Thus, the optimal number of boosting iterations can be easily obtained. GBM, on the other hand, uses a grid search and can only detect a limited number of values.

We use the "XGBoostEstimator" in the spark-ml library to realize the XGBoost [3]. Here, the max depth of the tree is set to 6, which could prevent the overfitting. We apply L2 regularization term to make model more conservative.

3.2.5 Light GBM.

Light Gradient Boosting Machine (Light GBM) is a gradient boosting framework based on gradient boosting decision tree (GBDT) [7]. It works in a similar way to XGBoost. Light GBM is designed to be distributed. Compare to the XGBoost, it's efficient with the faster training speed, higher efficiency and better accuracy.

The shortcoming of the XGBoost is that calculating the information gain requires scanning all samples in order to find the optimal division point. Their efficiency and scalability are not satisfactory when dealing with large amounts of data or high feature dimensionality. A straightforward solution to this problem is to reduce the number of features and data without compromising accuracy. Light GBM, addresses these issues well, and it consists of two main algorithms. The first one is Gradient-based One-Side Sampling (GOSS). It excludes most samples with small gradients and uses only the remaining samples to calculate the information gain. Although GBDT has no data weights, each data sample has a different gradient, and by definition of calculating information gain, samples with larger gradients have a greater impact on information gain. Hence, we should try to keep samples with large gradients and randomly remove samples with small gradients when reducing the samples. The another one is Exclusive Feature Bundling (EFB). It improves computational efficiency by reducing the dimensionality of features through feature bundling. Usually, the bundled features are mutually exclusive (one has a zero value and one has a non-zero value), so that no information is lost when the two features are bundled. If the two features are not completely mutually exclusive (in some cases both features are non-zero), a metric can be used to measure the degree to which the features are not mutually exclusive, called the conflict ratio. When this value is small, we can choose to bundle two features that are not completely mutually exclusive, without affecting the final accuracy.

We use the "LightGBMClassificationModel" in the spark-ml library to realize the Light GBM [3]. In training, the learning rate is 0.1 and the maximum number of iterations is 100. In the LightGBM model, the number of trees constructed is decided by the number of iterations and the class. Here, the dataset contains two classes. Therefore, there are 200 trees in the final ensemble model.

3.3 Test results

We trained and tested several classifiers that are widely used in binary classification problems, including Random Forest, SVC, Logistic Regression, Light GBM and XG BOOST. And we summarized the results for these 5 models, in terms of accuracy on the training, validation and test sets shown in Table 1. It is found that

Table 1. The accuracy of training, validation, and test sets with different classifiers.

Accuracy	XG BOOST	Random Forest	SVC	Logistic Regression	Light GBM
Training	0.9751	0.9924	0.9706	0.9706	1.0
Validation	0.9555	0.9427	0.9521	0.9366	0.9521
Test	0.9010	0.9630	0.9937	0.9747	0.9688

the performance of all models was acceptable due to the organized data features in the previous part. SVM preformed best, even reaching the 99% accuracy in the test set.

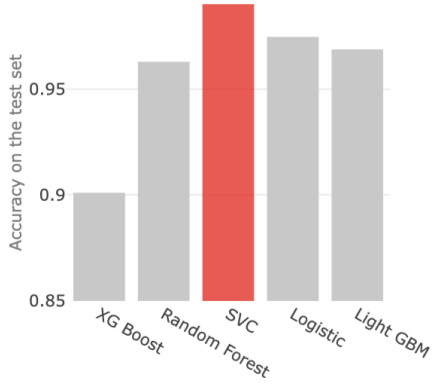

Figure 8. Ranking the top 5 models in terms of test set accuracy.

Table 2. The model evaluation of training, validation, and test sets with SVC.

SVC	TP	FP	FN	TN	Precision	Recall	AUC
Training	126	3	6	222	0.9767	0.9545	0.9706
Validation	36	1	3	52	0.9729	0.9231	0.9521
Test	41	1	0	78	0.9762	1	0.9937

IV. CONCLUSION

4.1 Summary

Overall, we have implemented pre-processing of the data and feature analysis. Through PCA analysis, we fouded that top 5 predictors are 'concave_points_mean', 'area_mean', 'radius_mean', 'perimeter_mean', 'concavity_mean'. Besides, we have trained and tested the Random Forest, Support Vector Machine, Logistic Regression, Light GBM and XG BOOST Classifiers, achieving 99% accuracy. The prediction is quite accurate, but for medical usage, the interpretability needs to be improved.

4.2 Discussion

In real life, predictive models, similar to our project, will be used only as an aid to diagnosis. The final diagnosis will be made by a doctor with extensive clinical experience. Therefore, the predictive models implemented in this project do not produce negative results, meaning that deviations in the model predictions do not mislead the diagnosis and treatment of patients. Secondly, the results of the model are measurable, as some of the tumor tissue collected during surgery will be tested to determine whether the tumor is benign or malignant. Most importantly, the results of each prediction do not provide feedback to the model. Each prediction is independent and the predicted outcome is not the final diagnosis. In summary, our model will not produce a Weapon of Math Destruction, provided it is used correctly.

In selecting and training our models, we take into account issues of fairness. Firstly, our training data are all pathological characteristics of the tumor and do not involve any protected attribute such as gender, race, age, region etc. Secondly, the prediction accuracy of our model is close for benign and malignant tumors and our model does not favor one category over the other. The recall of the model is 1, which represents the probability that the tumor is truly malignant and predicted to be malignant. And the specificity of the model is 0.99, which represents the probability of a true benign tumor being predicted to be benign. Hence, the model is fair.

4.3 Future Work

Although our prediction model has achieved a very high accuracy rate, it can only be used as a reference for doctors' diagnosis in clinical practice. Because the patient's situation is very complex, it cannot be judged by just one model constructed from data.

For the prediction of diagnosis of breast cancer, the AI technology needs to expand the application of the overall sample size and cross-crowd ethnographic database. A sufficiently large database is sufficient to support the prediction of diagnosis by AI technology, and it can help clinicians to find the factors that have the greatest impact on the cancer, so as to establish future prospective intervention research.

In addition, interpretability is an important consideration for AI applications in breast cancer. Although DL models showed excellent accuracy in the diagnosis, they are considered as "black boxes" owing to their lack of interpretability.

Reference:

[1] Sung, H., Ferlay, J., Siegel, R. L., Laversanne, M., Soerjomataram, I., Jemal, A., & Bray, F. (2021). Global cancer statistics 2020: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries. CA: A Cancer Journal for Clinicians, 0(0), 1–41. https://doi.org/10.3322/caac.21660

[2]https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Diagnostic%29

[3]https://spark.apache.org/docs/3.2.0/ml-migration-guide.html#breaking-changes-3

[4] Suthaharan, Shan. "Support vector machine." Machine learning models and algorithms for big data classification. Springer, Boston, MA, 2016. 207-235.

[5] Wright, Raymond E. "Logistic regression." (1995).

[6] Chen, Tianqi, and Carlos Guestrin. "Xgboost: A scalable tree boosting system." Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining. 2016.

[7] Ke, Guolin, et al. "Lightgbm: A highly efficient gradient boosting decision tree." Advances in neural information processing systems 30 (2017): 3146-3154.