Stats 101C Final Project

Predictive Analysis of Breast Cancer Diagnostic

Abstract

The goal of this kaggle project is to predict the diagnosis status of breast cancer using statistical learning models based on the given data, and this report provides a clear and detailed description on how we build our classification model from start to the end, including introduction, exploratory data analysis, data cleaning, feature selection, model construction, analyzing results, discussing limitations and recommendations.

The final model is based on logistic regression and uses variables texture_mean, area_mean, smoothness_mean, compactness_mean, concave.points_mean, symmetry_mean, fractal_dimension_mean, texture_se, area_worst, concave.points_worst. It has a Kaggle score of 0.66978, and our final rank is 3th.

1. Introduction

Breast cancer is one of the most common cancers among women and the major cause of death among women worldwide. Every year approximately 124 out of 100,000 women are diagnosed with breast cancer, and the estimation is that 23 out of the 124 women will die of this disease. When detected in its early stages, there is a 30% chance that the cancer can be treated effectively, but the late detection of advanced-stage tumors makes the treatment more difficult. Currently, the most used techniques to detect breast cancer in early stages are: mammography (63% to 97% correctness [5]), FNA (Fine Needle Aspiration) with visual interpretation (65% to 98% correctness [6]) and surgical biopsy (approximately 100% correctness). Therefore, mammography and FNA with visual interpretation correctness varies widely, and the surgical biopsy, although reliable, is invasive and costly. Thus, finding an efficient model to successfully detect breast cancer is significant and urgent.

In this kaggle project, the Wisconsin Breast Cancer Diagnostic Data set provided us with 40 variables describing different features of the cell nuclei. The training dataset provided contains 748 observations, and the testing dataset contains 321 observations. After having a quick glance at 40 variables, we found they are both numerical and categorical. The numerical variables show the 10 properties of the cell nuclei like radius, texture and symmetry, and we have mean, sd and worst measure of that particular cell. The categorical predictors provide more details of each cell nucleus, such as the position and the class. Our mission was to introduce a classification model to predict the target variable diagnosis "B(benign)" and "M(malignant) " in testing data by selecting key variables.

2. Data Analysis

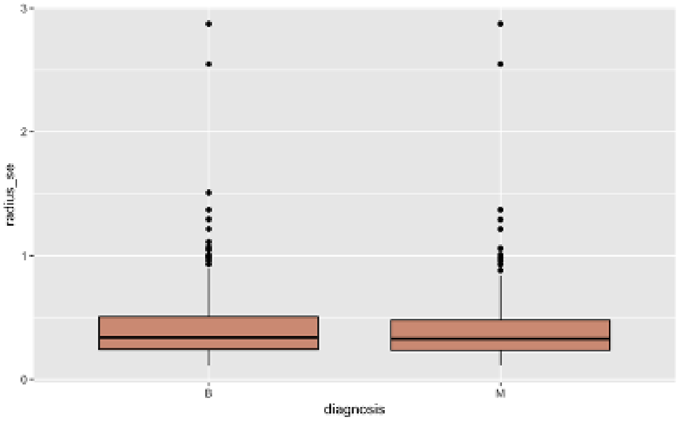
A. Variables exploratory data analysis

Before we started, we decided to perform some exploratory data analysis on variables as EDA helps us to investigate the dataset and better understand patterns within the data, as well as find interesting relations among the variables.

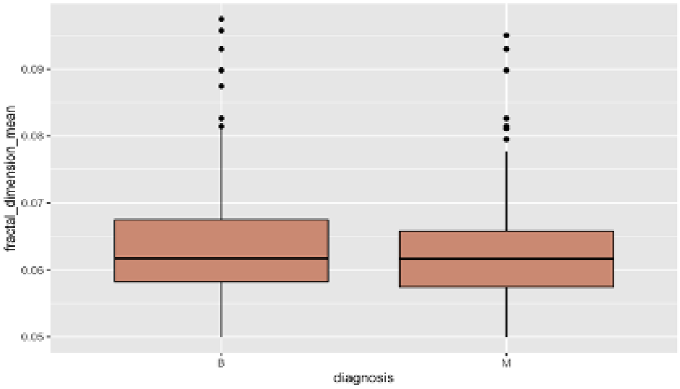
a. EDA - numerical

Since our target variable diagnosis is binary, the boxplot would be helpful. By plotting and analyzing the boxplot, we divided our numerical variables into 3 patterns.

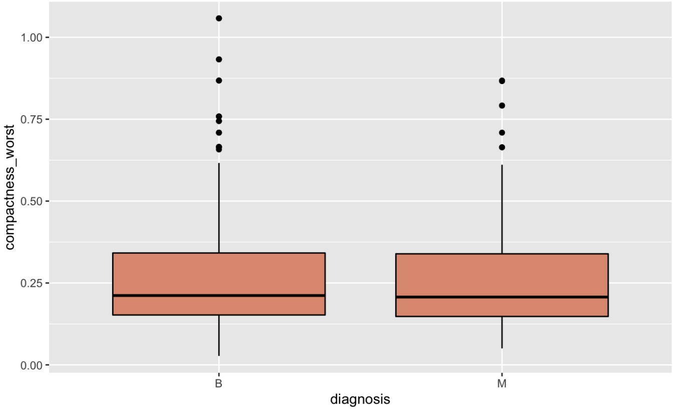
Pattern A are the variables with many outliers outside the boxplot such as Radius_se, perimeter_se and area_se:



Pattern B are the variables that have difference in range and median such as fractional_dimension_mean, concave.point_mean and smoothness_worst:



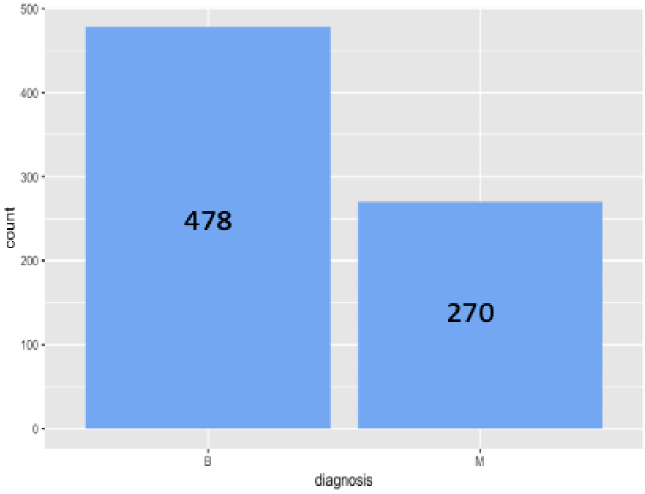
Pattern C are the variables that have negligible differences such as radius_se and compactness_worst:



From these three patterns, we can conclude that the variables in pattern B are potentially significant, and we will analyze those variables further in feature selection.

b. EDA - categorical

We did the same EDA to categorical variables. For our target variable diagnosis, there are 478 “B” and 279 “M”.



We also check the levels of other categorical variables:

Variables	Levels	Variables	Levels
age	6	class	2
menopause	3	breast	2
inv.nodes	7	breast.quad	5
node.caps	2	irradiat	2

From the above chart, we can see that different categorical variables have different levels. Variables with many levels would be eliminated and variables with fewer levels would be considered for our classification model.

B. Clean data -- finding and dealing with missing values

We noticed there are some missing values NAs in both training and testing data. The handling of missing data is very important during the preprocessing of the dataset as it will negatively affect our model's performance, and many machine learning algorithms do not support missing values. Thus, we decided to find and deal with these missing values.

In the training dataset, the missing values are the following:

node.caps	breast.quad
19	**2**

In the testing dataset, the missing values are the following:

node.caps	breast.quad
19	**2**

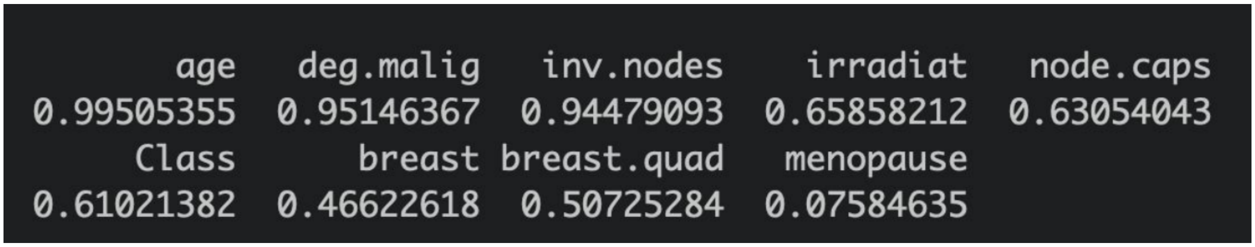
From the above tables, We noticed that all NAs are in categorical variables nodes.caps and breast.quad (We didn't impute NAs here since we will omit all categorical variables in the variable selection step). There are no missing values in numerical variables.

C. Variables Selection

After EDA and cleaning missing values, we started our feature selection process which helped us dig deeper into the data and choose the predictor more correctly .

a. categorical variables selection

For the categorical variables selection, we decided to use Chi-squared test to check whether any of the categorical variables would significantly affect the target variable - diagnosis. We built the test between each individual categorical variable and “diagnosis”, then checked the p-value of the Chi-squared test. If the p-value of the test result is smaller than 5% significance level, we would think that the factor could be a factor affecting the diagnosis result.



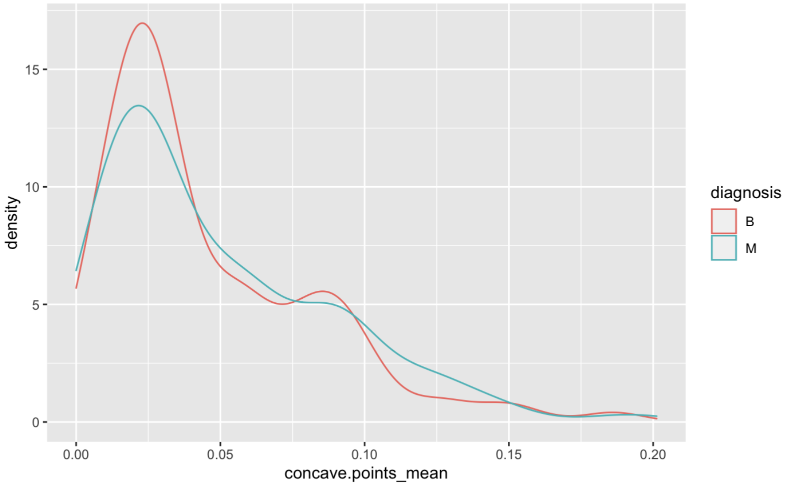
```
      age   deg.malig   inv.nodes    irradiat   node.caps
0.99505355  0.95146367  0.94479093  0.65858212  0.63054043
     Class      breast breast.quad   menopause
0.61021382  0.46622618  0.50725284  0.07584635
```

However, by using Chi-squared test to select the categorical variables, we find all the p-value of tests are greater than 5% significance level, which indicates that there is no strong causal relationship between diagnosis and all categorical variables. Even for the categorical variable - “menopause”, which has the greatest value of correlation with “diagnosis” among all the categorical variables, its p-value of the Chi-squared test here is still greater than the threshold of 5%. Thus, we would not include the categorical variables in our predict model and that’s why we didn’t impute NAs in the data cleaning step.

b. numerical variables selection

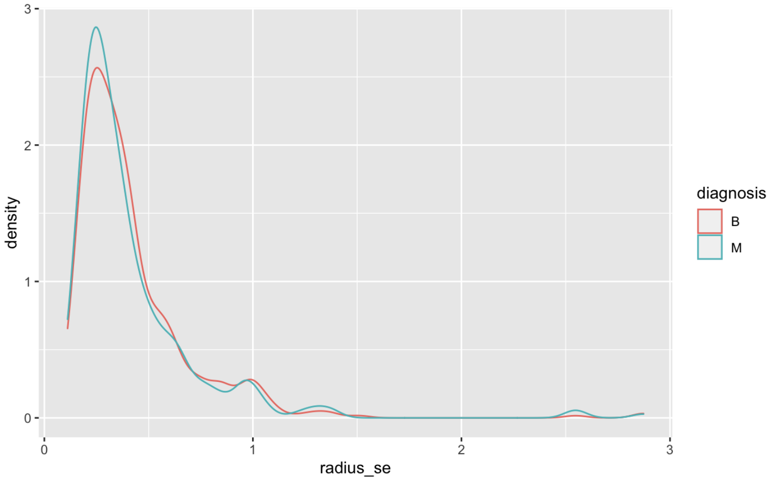
Since our response variable diagnosis is a binary categorical variable, we chose the quantitative variables based on density plots. We plotted the two density curves of each predictor over the two categories of diagnosis. If the density plot shows distinct differences in distribution, we can conclude that this predictor is potentially significant and keep them; we would discard predictors that have a lot of overlap.

i. Example for potentially helpful predictor:



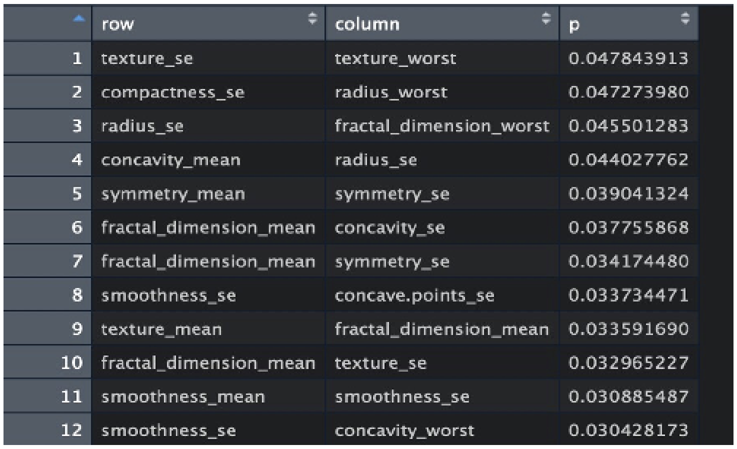
From the above graph, we can see that there are significant differences between the two density curves, indicating for each concave.point_mean value, the probabilities of getting benign and malignant breast cancer would be different, so the variable is helpful and we keep them.

ii. Example for less potentially helpful variable:



From the above density plot, we can see that density curves for B and M have large overlap, pointing out that for each radius_se value, the probabilities of getting benign and malignant breast cancer would be almost the same, so we discard the variable.

In addition to the density plots, we calculated the correlation and p-value between numerical variables. If the p-value of the test result is less than 5% significance level, we would conclude that these two variables are highly correlated, and we would choose the one which is more helpful.



	row	column	p
1	texture_se	texture_worst	0.047843913
2	compactness_se	radius_worst	0.047273980
3	radius_se	fractal_dimension_worst	0.045501283
4	concavity_mean	radius_se	0.044027762
5	symmetry_mean	symmetry_se	0.039041324
6	fractal_dimension_mean	concavity_se	0.037755868
7	fractal_dimension_mean	symmetry_se	0.034174480
8	smoothness_se	concave.points_se	0.033734471
9	texture_mean	fractal_dimension_mean	0.033591690
10	fractal_dimension_mean	texture_se	0.032965227
11	smoothness_mean	smoothness_se	0.030885487
12	smoothness_se	concavity_worst	0.030428173

Finally, we conclude that texture_mean, area_mean, smoothness_mean, compactness_mean, concave.points_mean, symmetry_mean, fractal_dimension_mean, texture_se, area_worst, concave.points_worst are potentially significant and we would use them in our classification model.

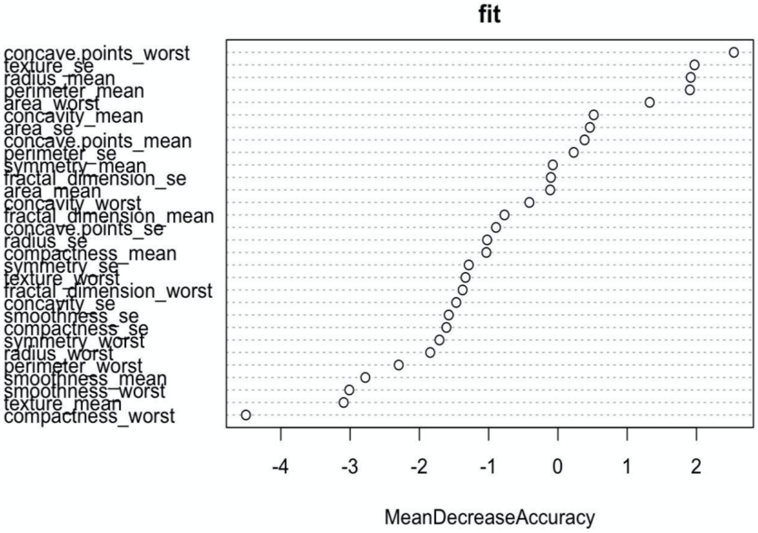
3. Methods & Models

Before we started to build our model, we used random sampling to split the training data set into 70% new training data and 30% new testing data. Then, we used new training data to construct a classification model and produce the predicted diagnosis based on new testing data. By comparing the predicted diagnosis and the true diagnosis from new testing data, we will get the accuracy rate. We would apply these steps to different models like Random Forest, logistic regression, lda and etc in order to find the highest accuracy and best-performing model.

A. Random forest

Random Forest is an ensemble learning method which is applicable for both classification and regression. It’s easy to find the relative importance of each feature and could add additional randomness to the model. Besides, the nonlinear relationship between parameters won’t affect the tree performance. It also reduces overfitting in each decision tree and helps to improve the accuracy. Considering these benefits, we decided to try Random Forest first.

We used the Variable Importance Plot to find the best size of the subset. We sorted the mean decrease accuracy of the predictors and chose the 8 best predictors (concave.points_worst, texture_se, radius_mean, perimeter_mean, area_worst, concavity_mean, area_se, and concave.points_mean) to include into our model. Unfortunately, the result we got from Random Forest was not as good as we expected. We firstly estimated the misclassification rate of the model with our testing data, getting the accuracy rate of 0.644850. After uploading our prediction on kaggle, the accuracy rate was only around 66.6%, which is not the best one compared to our other models.

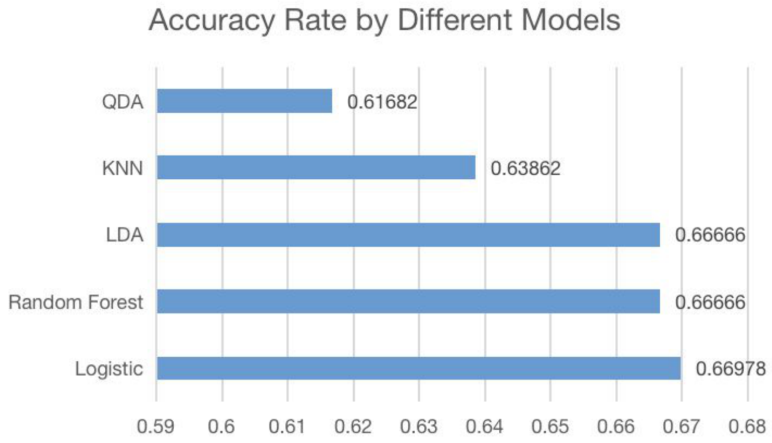


B. Logistic regression

Logistic regression is a simple yet very powerful algorithm to solve binary classification problems, and it is easier to implement and interpret. Besides, for our dataset, the number of observations is greater than the number of predictors; in this case, logistic regression would be very efficient. Thus, we used the predictors obtained from feature selection to run the logistic regression. The accuracy with our new testing dataset in R is 0.645743. After we submitted our prediction on kaggle, the accuracy was 0.66978, which is the highest score.

C. Others

Other than Random Forest and Logistic Regression, we also tried QDA, KNN, and LDA, and got the corresponding accuracy rate of 0.61682, 0.63862 and 0.66666, respectively. The performance of these methods were all not better than the Logistic Regression.



5. Discussion and Limitation

Although we got 3th in the leaderboard, we acknowledge that the accuracy is actually unsatisfactory and there are some limitations in our model.

Random Forest is computationally expensive. Also, it's hard to interpret the model as it works like a black box and we have little control on what each model does. Besides, the prediction is unstable when changing the variables and it's largely dependent on the seed setted.

Logistic Regression is limited to linearity. If the relationship between target variable and predictors are more complex, a different model or machine learning method will be much better.

LDA and QDA have the similar issue as Logistic Regression - they are also largely dependent on the relationship between the target variable and predictors. LDA works well in linear classification while QDA outperforms when the relationship is quadratic.

KNN does not work well in large datasets with high dimensions. It requires feature scaling before applying and the predicted results would be significantly affected by the selection of "K" value, which makes it difficult to find the best predictive model.

From the confusion matrix of prediction of training data, we found that our models here are much more accurate in classifying benign than malignant; however, in real world, detecting malignant is more significant and we would prefer the model which is more successful predicting malignant breast cancer.

6. Conclusion and Recommendation

From this project, we learned that the Logistic Regression outperforms when the number of observations is greater than the number of predictors and there is no complex interaction term. Thanks to the presentation and report, we realized that it is important to have a clear and organized process when we build our predictive model, including introduction, exploratory data analysis, data cleaning, feature selection, model construction and finally analyzing our results. We also learned that the limitations of different models will significantly affect the prediction accuracy. It's important to choose machine learning methods matching the characteristics of the datasets. Due to the time restriction, we fail to try more potential transformations of the variables to figure out a more precise model. If we could have more time to dive deeper into the project, we might probably group the variables by "mean", "se" and "worst", then figure out the interaction between the variables inside each group.

Overall, this project is a great experience for us to learn and practice our skill with real-world data, which would be beneficial for our future learning.

7. Acknowledgement

We would like to express our great gratitude to Professor Akram Mousa Almohalwas for teaching us Stats 101C this summer and all the help, suggestions and encouragement.

References

Kumar, Satyam. “Predict Missing Values in the Dataset.” *Towards Data Science,* Jul 26, 2020, https://towardsdatascience.com/predict-missing-values-in-the-dataset-897912a54b7b

IBM Cloud Learn Hub. “Exploratory Data Analysis.” Aug 25, 2020, https://www.ibm.com/cloud/learn/exploratory-data-analysis#:~:text=Exploratory%20data%20analysis%20(EDA)%20is

Almohalwas, Akram Mousa. “Using The Wisconsin Breast Cancer Diagnostic Data Set for Predictive Analysis”. Jun, 22, 2021.