Predicting Diagnosis of Breast Cancer Tumor

Jianghui Lin jl5172, Zixu Wang zw2541, Jack Yan xy2395

Introduction

This dataset includes different features which were computed from a digitized image of a fine needle aspirate (FNA) of a breast mass. The FNA describes the characteristics of the cell nuclei present in the image. Ten real-valued features are computed for each cell nucleus. The features include radius, texture, perimeter, area, compactness, concavity, concave points, symmetry, and fractal dimension. In this project, we aim to use the lab results to determine which features contribute to a better diagnosis result of breast mass (M = malignant, B = benign). In the dataset, we have 357 benign observations and 212 malignant observations, so the two classes are balanced.

Dataset Partitioning

The original dataset contains 569 observations, among which 456 were randomly partitioned into the training set, and 113 into the test set. The training set was used to conduct model selection based on 10-fold cross-validation, and the test set was used to verify the model selection result.

Exploratory Data Analysis

1. K Means Clustering

K-means clustering was used to partition the observations into 2 clusters. In this dataset, the optimal number of clusters is 2, determined by average silhouette method using the the `fviz_nbclust` function. Interestingly, the number of optimized clusters exactly equals the number of response categories 'B' and 'M'. **(Figure 1)** shows the distribution of clusters on the first 2 principle components (PCs). Dots are labelled by their response categories, although response information was not used in clustering. The two clusters can be well separated by the first 2 PCs, and the two outcome classes are separated into each of the clusters.

2. Correlation plots

The correlation plot **(Figure 2)** shows that some strong correlation exist among a subset of covariates. Intuitively, the high correlation makes sense. For example, the correlation among perimeter, radius, and area are all related to the size of the tumor. However, we did not know which covariates are better, so we decided to keep all covariates at this point, and let the models decide which variables are more important.

3. One-to-one relation between classes and covariates

According to the feature plot **(Figure 3)** that shows the one-to-one relation between the response and covariates, The 'M' (malignant tutor) class generally has larger radius mean, texture mean, perimeter mean, area mean, compactness mean, smoothness mean and concavity mean compare to those of 'B' (benign) tumors. The two classes can be well separated on many individual covariates, such as radius, texture and concave points.

Models

All the 30 covariates were included in all the models. The 30 predictors are the means, standard errors and worst values of the 10 characteristics measured on the cell nuclei. Ordinary and regularized Logistic regression,

Linear and quadratic discriminant analysis, Naive Bayes, KNN, support vector machines, classification trees, random forests, and ensemble methods were used to fit the data. The area under the ROC curve (AUC) were calculated from repeated cross-validation and used to compare models. For models that require tuning, repeated cross-validation was used to decide the optimal hyperparameter(s) that corresponds to the largest AUC.

1. Logistic regression and regularized logistic regression (glmnet)

Both logistic regression and regularized logistic regression models (glmnet) were built to fit the 30 predictors on the response. Data were centered and scaled before fitting the glmnet model. Cross-validation showed that the logistic regression has AUC 0.9544, sensitivity 0.9821 and specificity 0.8856. The best parameter of glmnet is alpha 0.2 and lambda 0.014, with AUC 0.9961, which is better than the logistic regression model.

2. LDA

The assumption of LDA is that all the variables follow a multivariate normal distribution. Therefore, each covariate has its own mean and shares a common variance-covariance matrix. All 30 predictors were included in this model. By fitting the model with the cross-validation method, the optimal LDA model has AUC 0.9917, sensitivity 0.8958 and specificity 0.9930.

3. QDA

QDA assumes that all the variables are following a Gaussian distribution. Therefore, different from LDA, each covariate has its specific mean and different variance-covariance matrix. All 30 predictors were included in this QDA model and the AUC is 0.9910, sensitivity is 0.9718, specificity is 0.9425.

4. Naive Bayes

Naive Bayes assumes that the features are conditionally independent given the class instead of modeling their full conditional distribution given the class, and it is an approximation to the Bayes classifier. All 30 predictors were included in the model and the best AUC is 0.9873.

5. KNN

All 30 predictors were included in this KNN model, and the best tuning parameter is 33. This model has AUC 0.9925, sensitivity 0.8784 and specificity 0.9894.

6. Support Vector Machine (linear and radial kernel)

Support vector machines with both linear kernel and radial kernel were fit on the data. The optimal linear kernel has AUC 0.994 and the optimal radial kernel has AUC 0.996. The radial kernel is slightly better.

7. Classification Tree and random forests

A classification tree was built on the 30 predictors. Cp was tuned using cross-validation. The best cp turned out to be 0.0062, corresponding to 5 terminal nodes and AUC = 0.9412. In the random forest model, both interaction depth (mtry) and minimal node size were tuned by cross-validation. The best model random forest model had interaction depth 1 and minimal node size 1, with AUC 0.9931, which is much better than that of a single tree.

8. Boosting (AdaBoosting and binomial loss function)

Binomial loss boosting and AdaBoosting were tuned on 3 hyperparameters: number of trees, interaction depth, and shrinkage. Minimal node size was set to be 1. The best binomial loss boosting model and the best AdaBoosting model had AUC 0.9926 and 0.9930, respectively. Their AUC are comparable.

Model Selection

Model selection was based on repeated cross-validation. The mean AUC was summarized in **Table 1**. The support vector machine with radial kernal and glmnet have the same largest AUC 0.9957. Since the glmnet model can be easier interpreted, we decided to use it as our final model.

Final model

The final model is a mixture of ridge and lasso (alpha = 0.2, lambda = 0.014). Coefficients of the final regularized logistic regression (glmnet) are summarized in **Table 2**. Coefficients are arranged in descending order based on their absolute values. Among all the coefficients, SE of radius, worst value of radius, worst value of texture, worst value of concave points, and worst value of perimeter are the predictors with largest absolute coefficients. Since all the predictors are centered and scaled before fitting the model, the coefficients are on the same scale too. Therefore, their absolute values can reflect the importance of the predictors.

The train data ROC curve

APPENDIX

Figure 1 2-means clustering on the first two principle components

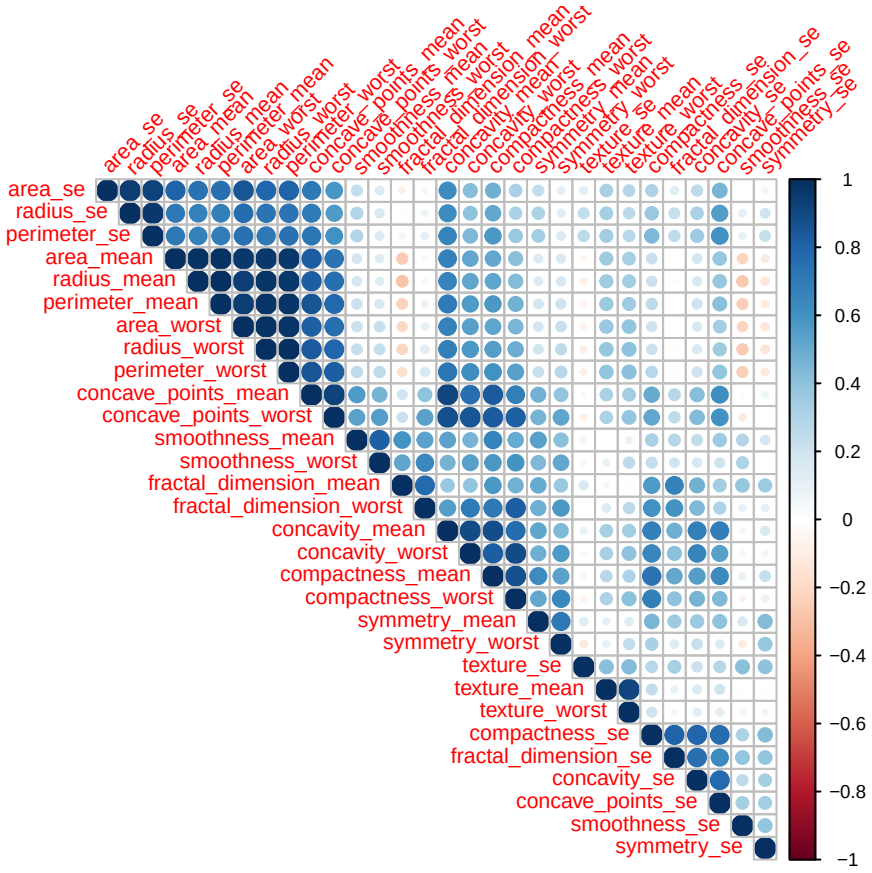

Figure 2 Correlation plots

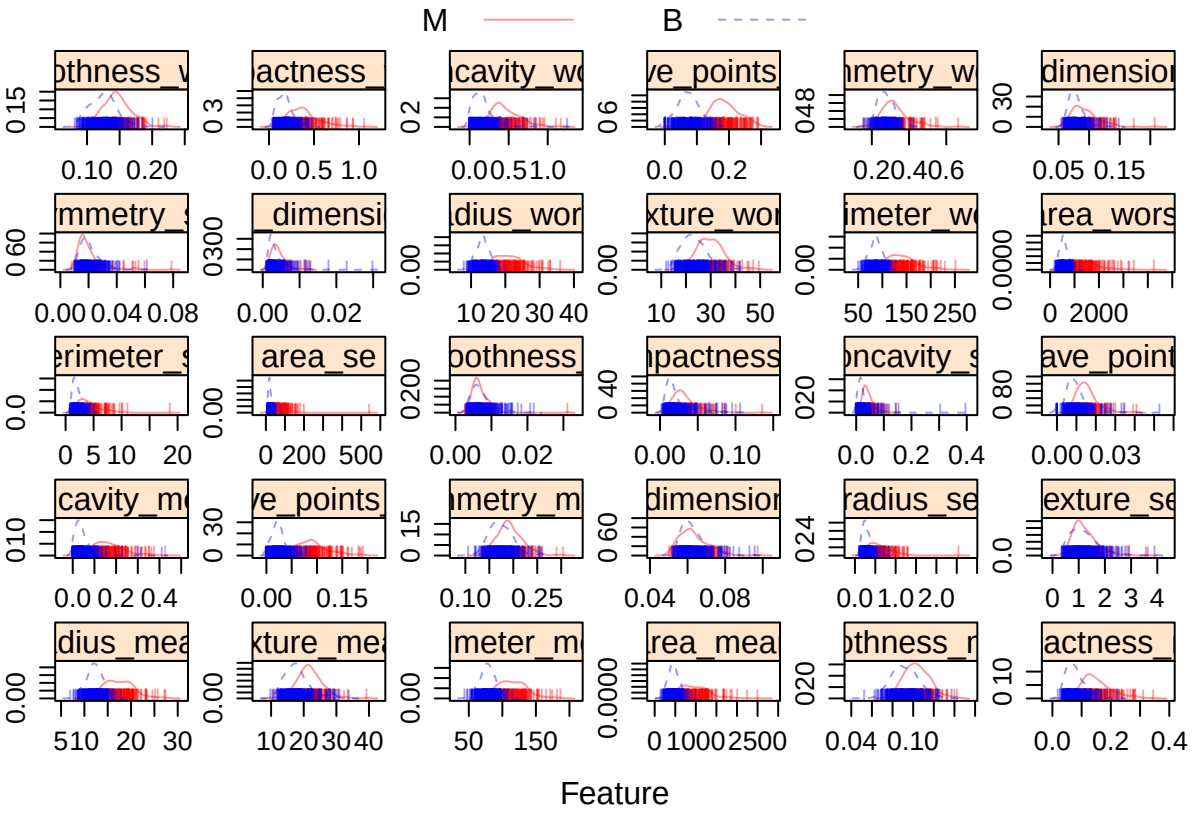

Figure 3 One-to-one relation between classes and covariates

```
## Warning: Setting row names on a tibble is deprecated.

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: Setting row names on a tibble is deprecated.
```

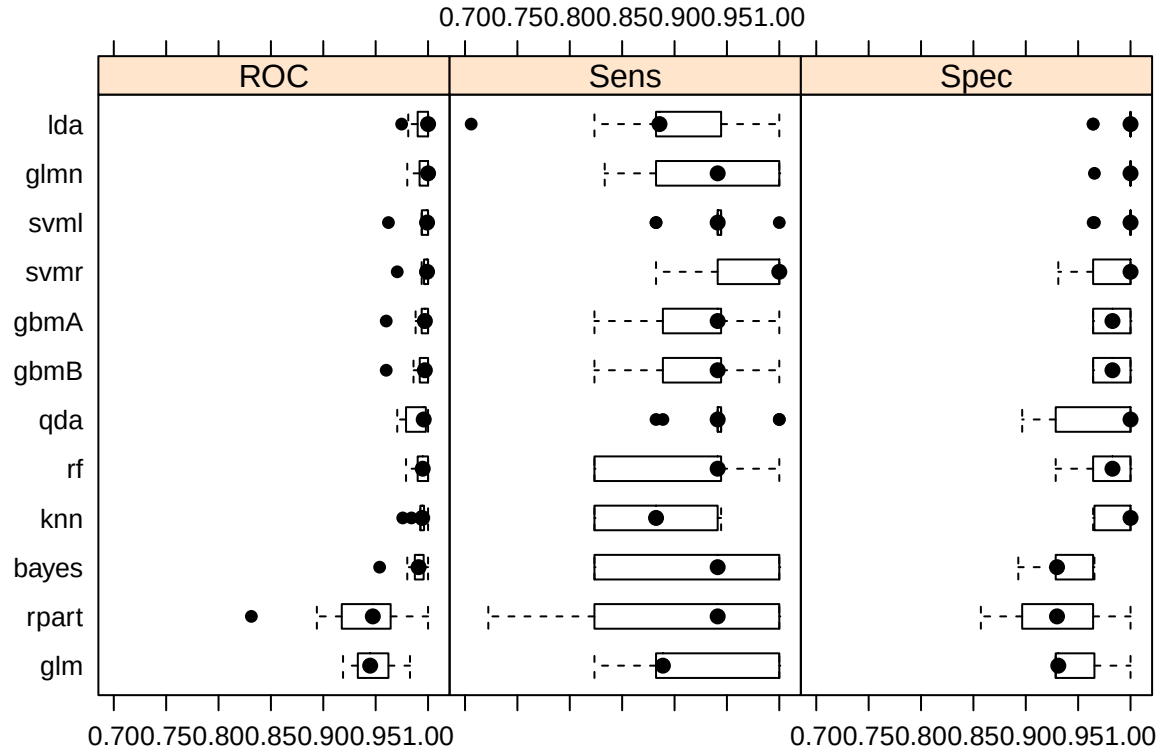

Figure Model comparision on AUC, sensitivity and specificity

Table 1: Mean of AUC, sensitivity and specificity of different models

model	AUC	Sensitivity	Specificity
svmr	0.9957	0.9650	0.9824
glmn	0.9957	0.9314	0.9966
svml	0.9944	0.9363	0.9930
lda	0.9938	0.8954	0.9929
rf	0.9933	0.9072	0.9788
gbmA	0.9930	0.9193	0.9824
gbmB	0.9926	0.9193	0.9824
knn	0.9925	0.8784	0.9894
qda	0.9899	0.9425	0.9718
bayes	0.9880	0.9186	0.9398
glm	0.9489	0.9190	0.9506
rpart	0.9363	0.9033	0.9329

Table 2: Non-zero Coefficients of the Final Model

```
# model.glmn$bestTune
formula3 <-
  diagnosis ~ radius_mean + texture_mean + perimeter_mean + area_mean +
  smoothness_mean + compactness_mean + concavity_mean + concave_points_mean +
  symmetry_mean + fractal_dimension_mean + radius_worst + texture_worst +
  perimeter_worst + area_worst + smoothness_worst + compactness_worst + concavity_worst +
  concave_points_worst + symmetry_worst + fractal_dimension_worst + radius_se + texture_se +
  perimeter_se + area_se + smoothness_se + compactness_se + concavity_se + concave_points_se +
  symmetry_se + fractal_dimension_se

train_scaled =
  scale(train_df[,-1]) %>% as.tibble() %>%
  mutate(diagnosis = train_df$diagnosis)
x <- model.matrix(formula3, train_scaled)[,-1]

y <- train_df$diagnosis %>% as.character()
y = if_else(y == "M", 1, 0)
glmn_best <- glmnet(x, y, family = "binomial",
                    alpha = 0.2, lambda = c(0.01407778,0.01407779))
coef = coef(glmn_best) %>% as.matrix %>% as.data.frame()
coef$term = rownames(coef)
# coeffients of the final model
coef =
  coef %>% as.tibble() %>%
  dplyr::select(term, s0) %>%
  rename(coef = s0) %>%
  filter(coef != 0, term != "(Intercept)") %>%
  arrange(desc(abs(coef)))
coef %>% knitr::kable(digits=4)
```

term	coef
radius_se	0.5796
radius_worst	0.5759
texture_worst	0.5550
concave_points_worst	0.5364

term	coef
perimeter_worst	0.5242
smoothness_worst	0.4886
area_worst	0.4783
concave_points_mean	0.4758
concavity_worst	0.4594
texture_mean	0.4250
area_se	0.4041
perimeter_se	0.3707
radius_mean	0.3549
symmetry_worst	0.3437
perimeter_mean	0.3436
concavity_mean	0.3422
area_mean	0.3185
fractal_dimension_se	-0.2383
compactness_se	-0.1616
symmetry_se	-0.1496
fractal_dimension_mean	-0.1288
smoothness_mean	0.1223
compactness_worst	0.1195
fractal_dimension_worst	0.0672
smoothness_se	0.0088

```
clst_plot = readRDS("nclust.rds")
clst_plot
```