Project 2: Breast Cancer Diagnosis

Shengzhi Luo, Xinran Sun, Haotian Wu, Lin Yang

3/31/2022

Objectives

A mammogram is an X-ray image of breast tissue. It can help save lives because it is easier to treat breast cancer in its early stages before the cancer is big enough to detect or cause symptoms. However, a wrong diagnosis can have a negative impact on patients. For example, if there is a false-positive test result, the doctor sees something that looks like cancer but is not. This could result in overtreatment that causes unnecessary side effects on patients. On the other hand, false-negative test result occurs when a doctor misses cancer tissues, which may delay the treatment. Therefore, building a model that gives an accurate classification of the tissue images is necessary to give proper treatment. In our study, we collected 569 images from both malignant and benign cancer tissues. Our goal is to build a predictive model to facilitate cancer diagnosis.

Dataset

Our data set consists of 569 rows, with 357 benign and 212 malignant. We denote 0 for benign and 1 for malignant. We also have 30 columns representing the features of the tissue images. They include the mean, standard deviation, and the largest values of the distributions of the following 10 features computed for the cell nuclei:

- radius (mean of distances from center to points on the perimeter)
- texture (standard deviation of gray-scale values)
- perimeter
- area
- smoothness (local variation in radius lengths)
- compactness ($perimeter^2/area$ - 1.0)
- concavity (severity of concave portions of the contour)
- concave points (number of concave portions of the contour)
- symmetry
- fractal dimension ("coastline approximation" - 1)

EDA

Before building the model, we want have a close look at the dataset. Therefore, we first examine the correlation between variables. The squares with dark color in the correlation plot has strong correlation with each other. We can see there is very strong correlation between radius, perimeter and area across mean, standard deviation, and the largest values. We decided to drop some variables with correlation larger than 0.7. The variables we dropped are perimeter_mean, area_mean, compactness_mean, concave_points_mean, perimeter_se, area_se, radius_worst, texture_worst, perimeter_worst, area_worst, and concavity_worst (11 variables).

After that, we built feature plot to analyze the relationship between variables after removing the variables with high correlation. From this plot, we can see that there are no strong relationship between variables after removing. We also found that the points for benign tissues are often locate at left-bottom side, which indicates the benign tissues usually have smaller feature values compared to malignant tissues.

We also calculated the mean of each variables to compare values between benign and malignant cases. According to the average values of the mean of each feature, we can find that benign tissues have smaller values compared to malignant tissues, except for fractal dimension. There is no general pattern for the average values of the standard deviations. Based the average values of the largest value of each feature, we can find that benign tissues have smaller largest values compared to malignant tissues.

To compare prediction performance of different models, the dataset is partitioned into the training data (0.8) and the test data (0.2).

Methods

Logistic Regression Model

Let y be the vector with 569 binary response variable, X be the 569×30 matrix with 30 numerical explanatory variables, and β be the vector with 30 corresponding coefficients. We also have β_0 as the intercepts.

For our logistic model, the probability of ith row be a malignant tissue is given by:

$$P(y_i = 1|X_i) = \frac{e^{\beta_0+\beta X_i}}{1 + e^{\beta_0+\beta X_i}}.$$

For likelihood function is:

$$L(\beta_0, \beta) = \prod_{i=1}^{n}[(\frac{e^{\beta_0+\beta X_i}}{1 + e^{\beta_0+\beta X_i}})^{y_i}(\frac{1}{1 + e^{\beta_0+\beta X_i}})^{1-y_i}].$$

Maximizing the likelihood is equivalent to maximizing the log likelihood:

$$f(\beta_0, \beta) = \sum_{i=1}^{n}[y_i(\beta_0 + \beta X_i) - \log(1 + e^{\beta_0+\beta X_i})].$$

The gradient of this function is:

$$\nabla f(\beta_0, \beta) = \begin{pmatrix} \sum_{i=1}^{n} y_i - p_i \\ \sum_{i=1}^{n} X_1(y_i - p_i) \\ ... \\ \sum_{i=1}^{n} X_n(y_i - p_i) \end{pmatrix} = X^T(y_i - p_i)$$

where $p_i = P(y_i = 1|X_i)$ as mentioned in previous probability function.

The Hessian is given by

$$\nabla^2 f(\beta_0, \beta) = -X^T W X$$

where $W = p_i(1 - p_i)$.

Newton-Raphson Algorithm

Newton-Raphson algorithm is a method to search for solutions to the system of equations $\nabla f(\beta_0, \beta) = 0$. At each step, given the current point $\boldsymbol{\beta}_0$, the gradient $\nabla f(\beta_0, \beta)$ for $\boldsymbol{\beta}$ near $\boldsymbol{\beta}_0$ may be approximated by

$$\nabla f(\beta_0, \beta) + \nabla^2 f(\beta_0, \beta)\,(\boldsymbol{\beta} - \boldsymbol{\beta}_0)$$

The next step in the algorithm is determined by solving the system of linear equations

$$\nabla f(\beta_0, \beta) + \nabla^2 f(\beta_0, \beta)\,(\boldsymbol{\beta} - \boldsymbol{\beta}_0) = \mathbf{0}$$

and the next "current point" is set to be the solution, which is a function of β_0:

$$\boldsymbol{\beta}_1 = \boldsymbol{\beta}_0 - [\nabla^2 f(\beta_0, \beta)]^{-1} \nabla f(\beta_0, \beta)$$

The ith step is given by a function of β_{i-1}:

$$\boldsymbol{\beta}_i = \boldsymbol{\beta}_{i-1} - [\nabla^2 f(\beta_{i-1}, \beta)]^{-1} \nabla f(\beta_{i-1}, \beta)$$

The Newton Raphson algorithm iterates through i beta values until the log-likelihood loss has converged. For this project, we uesd an additional half-stepping modification to the algorithm to control the number of iteration steps.

Path-wise Coordinate-wise Optimization Algorithm for Logistic-Lasso Model

To obtain a path of coefficients for a descending sequence of tuning parameter λ, we need to develop a coordinate-wise descent algorithm estimating coefficients for a specific lambda. The logistic-lasso can be written as a penalized weighted least-squares problem:

$$\min_{(\beta_0, \boldsymbol{\beta}_1)} L\left(\beta_0, \boldsymbol{\beta}_1, \lambda\right) = \left\{ -\ell\left(\beta_0, \boldsymbol{\beta}_1\right) + \lambda \sum_{j=0}^{p} |\beta_j| \right\}$$

When there are a large number of parameters, i.e., p is large, a coordinate-wise descent algorithm is required to optimize coefficients. The objective function is:

$$f\left(\beta_j\right) = \frac{1}{2} \sum_{i=1}^{n} \left(y_i - \sum_{k \neq j} x_{i,k} \widetilde{\beta}_k - x_{i,j} \beta_j \right)^2 + \gamma \sum_{k \neq j} \left| \widetilde{\beta}_k \right| + \gamma \left| \beta_j \right|$$

Minimizing $f\left(\beta_j\right)$ w.r.t β_j while having $\widetilde{\beta}_k$ fixed, we have weighted updates to update one coefficient at a time iteratively until the log-likelihood converges:

$$\widetilde{\beta}_j(\lambda) \leftarrow \frac{S\left(\sum_i w_i x_{i,j} \left(y_i - \tilde{y}_i^{(-j)} \right), \lambda\right)}{\sum_i w_i x_{i,j}^2}$$

where $\tilde{y}_i^{(-j)} = \sum_{k \neq j} x_{i,k} \widetilde{\beta}_k$.

If we apply Taylor expansion to the log-likelihood around "current estimates" $\left(\widetilde{\beta}_0, \tilde{\beta}_1\right)$, we have a quadratic approximation function $f\left(\beta_0, \boldsymbol{\beta}_1\right)$ to the log-likelihood:

$$f\left(\beta_0, \boldsymbol{\beta}_1\right) \approx \ell\left(\beta_0, \boldsymbol{\beta}_1\right) = -\frac{1}{2n}\sum_{i=1}^{n} w_i \left(z_i - \beta_0 - \mathbf{x}_i^T \boldsymbol{\beta}_1\right)^2 + C\left(\widetilde{\beta}_0, \widetilde{\boldsymbol{\beta}}_1\right)$$

where

$$z_i = \widetilde{\beta}_0 + \mathbf{x}_i^T \widetilde{\boldsymbol{\beta}}_1 + \frac{y_i - \widetilde{p}_i\left(\mathbf{x}_i\right)}{\widetilde{p}_i\left(\mathbf{x}_i\right)\left(1 - \widetilde{p}_i\left(\mathbf{x}_i\right)\right)}$$

$$w_i = \widetilde{p}_i\left(\mathbf{x}_i\right)\left(1 - \widetilde{p}_i\left(\mathbf{x}_i\right)\right)$$

$$\widetilde{p}_i = \frac{\exp\left(\widetilde{\beta}_0 + \mathbf{x}_i^T \widetilde{\boldsymbol{\beta}}\right)}{1 + \exp\left(\widetilde{\beta}_0 + \mathbf{x}_i^T \widetilde{\boldsymbol{\beta}}_1\right)}$$

w_i is the working weight, z_i is the working response, p_i is the probability of malignant case estimated at current coefficients. This quadratic approximation function is used in the coordinate-wise descent algorithm.

We then can develop a path-wise coordinate-wise optimization algorithm to get a path of solutions for a descending sequence of λ.

- Step 1: Find the smallest value λ for which all the estimated β are 0, defined as λ_{max}.
- Step 2: Define a fine sequence $\lambda_{max} \geq \lambda_1 \geq ...\lambda_{min} \geq 0$.
- Step 3: To estimate coefficients of the current λ_{k+1}, implement coordinate descent algorithm using the computed coefficients of the previous λ_k (warm start) as coefficient start values. ($\lambda_{k+1} < \lambda_k$)

Cross Validation

To select the best λ for the optimal model, a 5-fold cross-validation is performed.

- Step 1: Shuffle the original dataset randomly.
- Step 2: Split the shuffled dataset into 5 even groups.
- Step 3: Take one group as the test set and the remaining groups as the training set. Implement the path-wise coordinate-wise optimization algorithm based on the training data, and then calculate AUC scores for each λ using the test data.
- Step 4: Repeat this procedure until each of the 5 groups has been treated as test data, and mean AUC for each λ is computed.

Results

The coefficients of the full logistic regression model based on the training data using **glm()** are shown in **Table 1**. 5 predictors are found to be significant with p values less than 0.05, including **texture_mean**, **concavity_mean**, **radius_se**, **symmetry_worst**. For the logistic-lasso model, the λ_{max} is calculated to be 175.62, we then defined a descending sequence of 50 λ between λ_{max} and e^{-5}. Since the outcome is a binary variable, AUC is used as the evaluation metric to compare models. Through the 5-fold cross-validation, the best λ is found to be 0.981 with an average AUC of 0.997 (**Figure 3**). The coefficients for the best λ are shown in **Table 2**. As expected, the optimal logistic-lasso model shrinks some coefficients to

0. Predictors remained in the LASSO model are **radius_mean**, **texture_mean**, **concavity_mean**, **radius_se**, **compactness_se**, **fractal_dimension_se**, **smoothness_worst**, **concave_points_worst**, and **symmetry_worst**.

AUC scores of the full logistic regression model and the logistic-lasso model are computed based on the test data to compare prediction performance of the two models. The full model's AUC (0.977) is found to be less than the lasso model's AUC (0.996), thus the optimal lasso model slightly outperforms the full model.

Conclusions

Findings

The primary goal of our project is to build a model to predict whether a breast tissue sample is benign or malignant. After performing the exploratory data analysis, we drop 11 highly correlated variables, and 19 variables are used to fit a full logistic regression model. A Newton-Raphson algorithm is developed to estimate coefficients of the full model. We also compare the full model with a logistic-lasso model whose coefficients are estimated by a path-wise coordinate-wise optimization algorithm. The optimal LASSO model with the best λ is selected according to a 5-fold CV, and the optimal LASSO model is found to slightly outperform the full logistic model. Based on the optimal LASSO model, we realize some implications on breast cancer diagnosis. For example, breast tissue samples with higher mean radius tend to indicate malignant cases. On the other hand, the ones with lower compactness standard deviation tend to indicate benign cases.

Limitations

We found that Newton-Raphson algorithm is unstable during our work procedure. First, the convergence is not guaranteed, and it depends on the choice of starting values. If we set initial betas at large values, the algorithm would not work. Therefore, we have to carefully choose relatively small starting values. In addition, the number of lambdas in the lambda sequence is limited due to intensive computation of cross validation, thus we defined a sequence of 50 lambdas in the path-wise coordinate-wise optimization algorithm. If we include more lambdas, the selection of the best lambda would be more accurate and close to the truth.