

Project 2: Breast Cancer Diagnosis

Xinran Sun

3/17/2022

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

We built three feature plots to analyze the relationship between variables. The mean variables was plotted in first figure. From this plot, we can see that there are strong correlations between radius mean with perimeter mean and between radius mean with area mean. Perimeter mean and area mean also have strong correlation. Similarly, for the largest values, there are strong correlations between radius, perimeter and area. The correlations are reasonable because perimeter and area are calculated based on radius.

For benign cases, all of the 10 features' means and largest values are smaller than malignant cases.

Methods

Logistic 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 i th 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:

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

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 $W = p_i(1 - p_i)$.

Results

Conclusions

Appendix

data import and data clean

```

#load the data
breast_dat = read_csv("breast-cancer.csv") %>%
  janitor::clean_names() %>%
  select(-33) %>% #drop NA column
  add_row(id = 92751, diagnosis = "B", radius_mean = 7.76, texture_mean = 24.54,
          perimeter_mean = 47.92, area_mean = 181, smoothness_mean = 0.05263,
          compactness_mean = 0.04362, concavity_mean = 0,
          concave_points_mean = 0, symmetry_mean = 0.1587,
          fractal_dimension_mean = 0.05884, radius_se = 0.3857,
          texture_se = 1.428, perimeter_se = 2.548, area_se = 19.15,
          smoothness_se = 0.007189, compactness_se = 0.00466, concavity_se = 0,
          concave_points_se = 0, symmetry_se = 0.02676,
          fractal_dimension_se = 0.002783, radius_worst = 9.456,
          texture_worst = 30.37, perimeter_worst = 59.16, area_worst = 268.6,
          smoothness_worst = 0.08996, compactness_worst = 0.06444,
          concavity_worst = 0, concave_points_worst = 0,
          symmetry_worst = 0.2871, fractal_dimension_worst = 0.07039)
  #add missing row

head(breast_dat, 5)

## # A tibble: 5 x 32
##       id diagnosis radius_mean texture_mean perimeter_mean area_mean
##   <dbl> <chr>      <dbl>        <dbl>        <dbl>        <dbl>
## 1    842302 M         18.0       10.4       123.       1001
## 2    842517 M         20.6       17.8       133.       1326
## 3  84300903 M         19.7       21.2       130        1203
## 4  84348301 M         11.4       20.4       77.6       386.
## 5  84358402 M         20.3       14.3       135.       1297
## # ... with 26 more variables: smoothness_mean <dbl>, compactness_mean <dbl>,
## #   concavity_mean <dbl>, concave_points_mean <dbl>, symmetry_mean <dbl>,
## #   fractal_dimension_mean <dbl>, radius_se <dbl>, texture_se <dbl>,
## #   perimeter_se <dbl>, area_se <dbl>, smoothness_se <dbl>,
## #   compactness_se <dbl>, concavity_se <dbl>, concave_points_se <dbl>,
## #   symmetry_se <dbl>, fractal_dimension_se <dbl>, radius_worst <dbl>,
## #   texture_worst <dbl>, perimeter_worst <dbl>, area_worst <dbl>, ...

r = dim(breast_dat)[1] #row number
c = dim(breast_dat)[2] #column number

var_names = names(breast_dat)[-c(1,2)] #variable names

standardize = function(col) {
  mean = mean(col)
  sd = sd(col)
  return((col - mean)/sd)
}

stand_df = breast_dat %>%
  dplyr::select(radius_mean:fractal_dimension_worst) %>%
  map_df(.x = ., standardize) #standardize

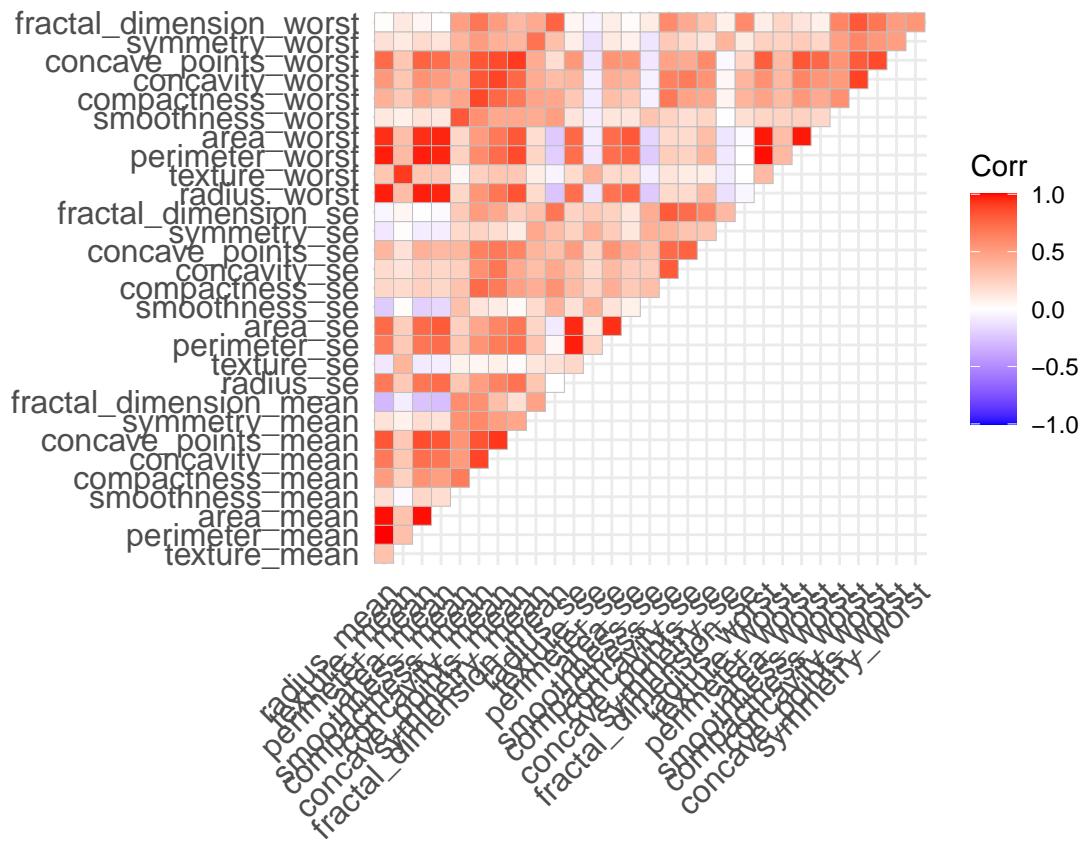
```

```
X = stand_df #predictors
y = as.vector(ifelse(breast_dat[,2] == "M", 1, 0))#response
```

check collinearity

```
corr = stand_df %>%
  cor()

ggcorrplot(corr, type = "upper")
```



```
#X = stand_df %>%
#select(radius_mean, texture_mean, smoothness_mean, compactness_mean,
#       #symmetry_mean, fractal_dimension_mean, radius_se, texture_se,
#       #smoothness_se, concavity_se, symmetry_se)

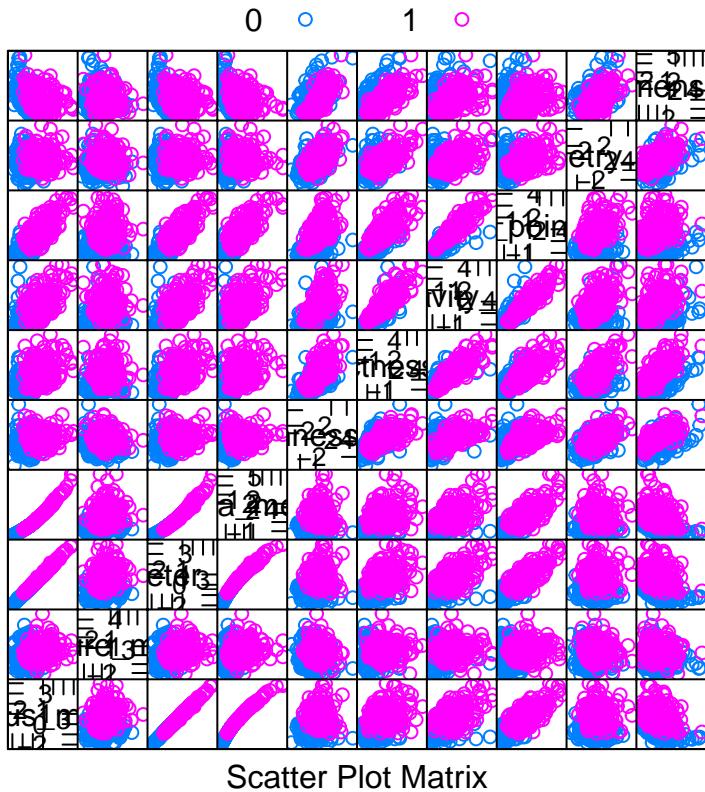
#corr_new = X %>% cor()
#corr_new

#ggcorrplot(corr_new, type = "upper", lab = TRUE)
```

feature plot

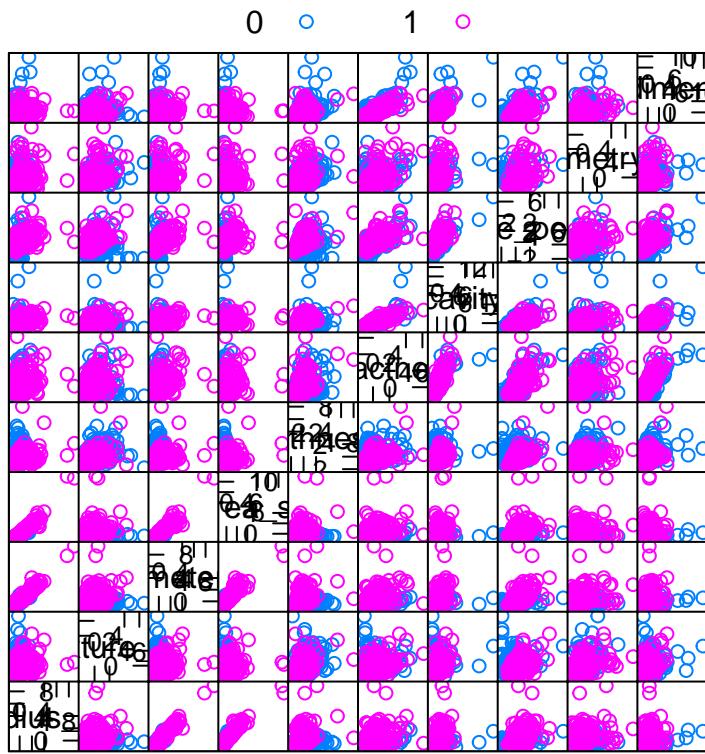
```
data = cbind(y,X)

featurePlot(x = data[, 2:11],
            y = factor(data$y),
            plot = "pairs",
            auto.key = list(columns = 2)
)
```



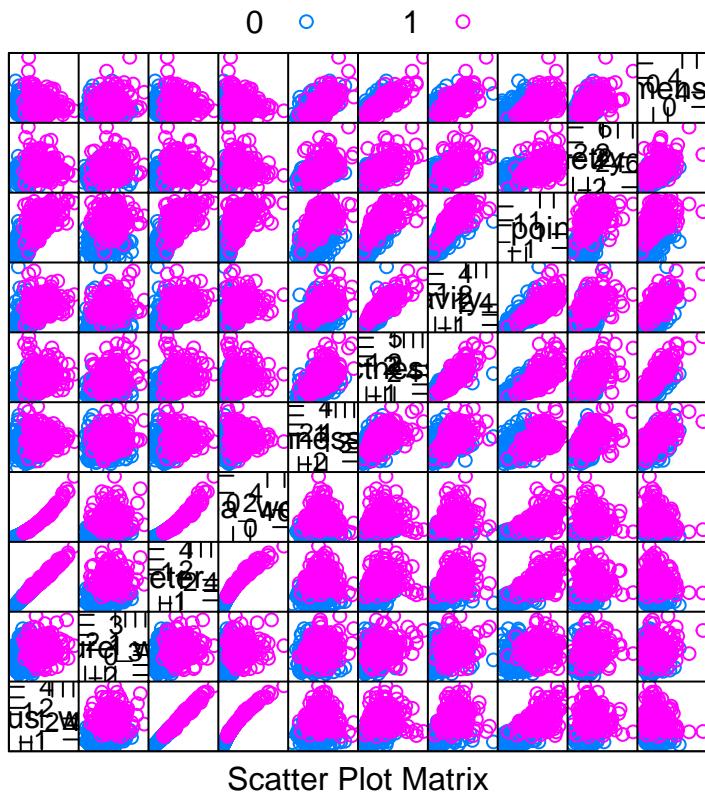
Scatter Plot Matrix

```
featurePlot(x = data[, 12:21],
            y = factor(data$y),
            plot = "pairs",
            auto.key = list(columns = 2)
)
```



Scatter Plot Matrix

```
featurePlot(x = data[, 22:31],
            y = factor(data$y),
            plot = "pairs",
            auto.key = list(columns = 2)
)
```



Scatter Plot Matrix

```
mean_data = breast_dat %>%
  group_by(diagnosis) %>%
  summarise(across(radius_mean: fractal_dimension_worst, ~ mean(.x, na.rm = TRUE)))
```

```
logdata = cbind.data.frame(y, X)
log_model = glm(y ~ ., family = binomial(link = "logit"), data = logdata)
summary(log_model)
```

```
##
## Call:
## glm(formula = y ~ ., family = binomial(link = "logit"), data = logdata)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max 
## -8.49   -8.49   -8.49    8.49    8.49 
##
## Coefficients:
##                               Estimate Std. Error z value Pr(>|z|)    
## (Intercept)                253916    23548  10.783 < 2e-16 ***
## radius_mean                 8552881    948876   9.014 < 2e-16 ***
## texture_mean                  842067    63252  13.313 < 2e-16 ***
## perimeter_mean                35796847    598698   59.791 < 2e-16 ***
## area_mean                   -45790271   1375034  -33.301 < 2e-16 ***
## smoothness_mean               -2144100    117586  -18.234 < 2e-16 ***
## compactness_mean                -339500    169667  -2.001  0.04540 *  
## concavity_mean                  83032     112278   0.740  0.45959  
## concave_points_mean            -665733    208830  -3.188  0.00143 **
```

```

## symmetry_mean          1109889      21306   52.093 < 2e-16 ***
## fractal_dimension_mean -298858       15312  -19.519 < 2e-16 ***
## radius_se               9230274      324119   28.478 < 2e-16 ***
## texture_se                3513102     110604   31.763 < 2e-16 ***
## perimeter_se              3438590      95432   36.032 < 2e-16 ***
## area_se                  -29084420    834804  -34.840 < 2e-16 ***
## smoothness_se             2249396      36747   61.213 < 2e-16 ***
## compactness_se            -3175247     102656  -30.931 < 2e-16 ***
## concavity_se               4614370     161208   28.624 < 2e-16 ***
## concave_points_se         -7773633     247582  -31.398 < 2e-16 ***
## symmetry_se                2389064      34103   70.054 < 2e-16 ***
## fractal_dimension_se        4001120     174560   22.921 < 2e-16 ***
## radius_worst              -29628795    1035752  -28.606 < 2e-16 ***
## texture_worst              -3584767     149772  -23.935 < 2e-16 ***
## perimeter_worst             -11889227    409644  -29.023 < 2e-16 ***
## area_worst                 50959831     1560436   32.657 < 2e-16 ***
## smoothness_worst            -493436      75304  -6.553  5.66e-11 ***
## compactness_worst           1413874      62922   22.470 < 2e-16 ***
## concavity_worst              -6316972     317828  -19.875 < 2e-16 ***
## concave_points_worst        9408268      359616   26.162 < 2e-16 ***
## symmetry_worst              -1530342      20986  -72.923 < 2e-16 ***
## fractal_dimension_worst      -667962      96441  -6.926  4.33e-12 ***
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance:  751.44  on 568  degrees of freedom
## Residual deviance: 32006.76  on 538  degrees of freedom
## AIC: 32069
##
## Number of Fisher Scoring iterations: 25

```