Logistic Regression: Breast Cancer Classification

Adam Quek

2025-06-17

R Packages required: “mlbench”, “dplyr”, “broom”, “forestplot”

1. Introduction

This analysis demonstrates how to apply logistic regression to a classification problem in medical diagnosis—specifically, to predict whether a breast tumor is benign or malignant.

The dataset used is the BreastCancer dataset from the mlbench package, which contains anonymized diagnostic data collected from fine needle aspirate (FNA) tests of breast masses. Each observation describes a set of cytological features of a tumor (e.g., clump thickness, cell size, cell shape) along with a known diagnosis outcome.

Objectives:

- Clinical goal: Build a predictive model to assist early identification of malignant tumors using routine cytological features.
- Statistical goal:
    1. Identify significant predictors of malignancy among the variables.
    2. Train a logistic regression model to classify new cases with high accuracy.
    3. Evaluate the model using prediction performance metrics (e.g., accuracy, confusion matrix).
    4. Present the findings using odds ratios and a forest plot to communicate the strength and significance of predictors.

This serves as a foundational example of how statistical learning techniques can support clinical decision-making in a health data context.

2. Load and Explore the Data

BreastCancer dataset is available in R package “mlbench” and contains clinical records of 699 breast cancer patients, including patients’ ID, 9 clinical features (variables) and medical diagnosis outcome (Class: benign or malignant).

```
data("BreastCancer", package = "mlbench")
str(BreastCancer)
```

```
## 'data.frame':    699 obs. of  11 variables:
##  $ Id             : chr  "1000025" "1002945" "1015425" "1016277" ...
##  $ Cl.thickness   : Ord.factor w/ 10 levels "1"<"2"<"3"<"4"<..: 5 5 3 6 4 8 1 2 2 4 ...
##  $ Cell.size      : Ord.factor w/ 10 levels "1"<"2"<"3"<"4"<..: 1 4 1 8 1 10 1 1 1 2 ...
```

```
##  $ Cell.shape     : Ord.factor w/ 10 levels "1"<"2"<"3"<"4"<..: 1 4 1 8 1 10 1 2 1 1 ...
##  $ Marg.adhesion  : Ord.factor w/ 10 levels "1"<"2"<"3"<"4"<..: 1 5 1 1 3 8 1 1 1 1 ...
##  $ Epith.c.size   : Ord.factor w/ 10 levels "1"<"2"<"3"<"4"<..: 2 7 2 3 2 7 2 2 2 2 ...
##  $ Bare.nuclei    : Factor w/ 10 levels "1","2","3","4",..: 1 10 2 4 1 10 10 1 1 1 ...
##  $ Bl.cromatin    : Factor w/ 10 levels "1","2","3","4",..: 3 3 3 3 3 9 3 3 1 2 ...
##  $ Normal.nucleoli: Factor w/ 10 levels "1","2","3","4",..: 1 2 1 7 1 7 1 1 1 1 ...
##  $ Mitoses        : Factor w/ 9 levels "1","2","3","4",..: 1 1 1 1 1 1 1 1 5 1 ...
##  $ Class          : Factor w/ 2 levels "benign","malignant": 1 1 1 1 1 2 1 1 1 1 ...
```

```
summary(BreastCancer)
```

```
##       Id             Cl.thickness   Cell.size     Cell.shape  Marg.adhesion
##  Length:699         1      :145   1      :384   1      :353   1      :407
##  Class :character   5      :130   10     : 67   2      : 59   2      : 58
##  Mode  :character   3      :108   3      : 52   10     : 58   3      : 58
##                     4      : 80   2      : 45   3      : 56   10     : 55
##                     10     : 69   4      : 40   4      : 44   4      : 33
##                     2      : 50   5      : 30   5      : 34   8      : 25
##                     (Other):117   (Other): 81   (Other): 95   (Other): 63
##   Epith.c.size  Bare.nuclei   Bl.cromatin  Normal.nucleoli    Mitoses
##  2      :386   1      :402   2      :166   1      :443     1      :579
##  3      : 72   10     :132   3      :165   10     : 61     2      : 35
##  4      : 48   2      : 30   1      :152   3      : 44     3      : 33
##  1      : 47   5      : 30   7      : 73   2      : 36     10     : 14
##  6      : 41   3      : 28   4      : 40   8      : 24     4      : 12
##  5      : 39   (Other): 61   5      : 34   6      : 22     7      :  9
##  (Other): 66   NA's   : 16   (Other): 69   (Other): 69     (Other): 17
##        Class
##  benign   :458
##  malignant:241
##
##
##
##
##
```

3. Data Cleaning

Based on the background knowledge of the relevant study, the first 3 variables about cell features Cl.thickness (Clump thickness assessing if cells are mono- or multi-layered), Cell.size and Cell.shape are considered in the logistic model. As they were recorded as categorical variables in the dataset, we need to convert them into numerical ones. There are 16 missing values.

```
# Remove rows with missing data
bc <- BreastCancer[complete.cases(BreastCancer), ]

# Convert selected ordinal/categorical columns to numeric
bc[, 2:4] <- sapply(bc[, 2:4], as.numeric)

# Convert outcome variable to binary factor
bc <- bc %>%
  mutate(y = factor(ifelse(Class == "malignant", 1, 0))) %>%
```

```
  select(Cl.thickness:Cell.shape, y)

str(bc)
```

```
## 'data.frame':    683 obs. of  4 variables:
##  $ Cl.thickness: num  5 5 3 6 4 8 1 2 2 4 ...
##  $ Cell.size   : num  1 4 1 8 1 10 1 1 1 2 ...
##  $ Cell.shape  : num  1 4 1 8 1 10 1 2 1 1 ...
##  $ y           : Factor w/ 2 levels "0","1": 1 1 1 1 1 2 1 1 1 1 ...
```

4. Train-Test Split

To evaluate how well our logistic regression model generalizes to unseen data, we split the dataset into two parts:

- Training set (80%): used to fit the model.
- Test set (20%): used to assess model performance on new data.

This separation helps prevent overfitting and provides an unbiased estimate of prediction accuracy.

```
set.seed(100)
training.idx <- sample(1:nrow(bc), size = 0.8 * nrow(bc))
train.data <- bc[training.idx, ]
test.data <- bc[-training.idx, ]
```

5. Fit Logistic Regression Model

We fit a logistic regression model using the training data to estimate the probability that a tumor is malignant based on three key features: clump thickness, cell size, and cell shape.

Logistic regression is appropriate here because the outcome variable is binary (malignant = 1, benign = 0), and it models the log-odds of the outcome as a linear combination of the predictors.

```
mlogit <- glm(y ~ Cl.thickness + Cell.size + Cell.shape,
              data = train.data,
              family = "binomial")

summary(mlogit)
```

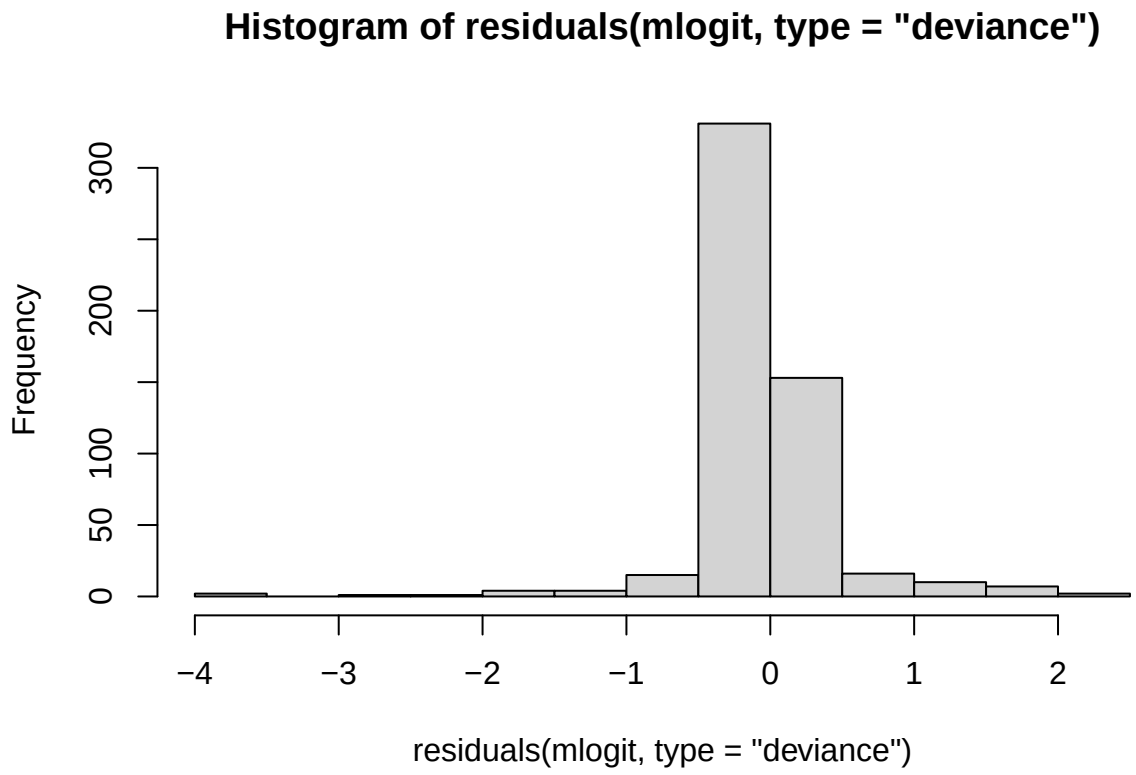
```
##
## Call:
## glm(formula = y ~ Cl.thickness + Cell.size + Cell.shape, family = "binomial",
##     data = train.data)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -7.7184     0.7699 -10.025  < 2e-16 ***
## Cl.thickness   0.5195     0.1121   4.634 3.59e-06 ***
## Cell.size      0.7081     0.2106   3.362 0.000773 ***
## Cell.shape     0.7679     0.1971   3.895 9.80e-05 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 703.10  on 545  degrees of freedom
## Residual deviance: 138.33  on 542  degrees of freedom
## AIC: 146.33
##
## Number of Fisher Scoring iterations: 7
```

```
summary(residuals(mlogit, type = "deviance"))
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -3.79676 -0.17562 -0.08081 -0.06123  0.01436  2.08919
```

```
hist(residuals(mlogit, type = "deviance"))
```



Model goodness-of-fit is assessed using *deviance residuals* and *AIC*:

- Deviance residuals are approximately normally distributed if the model is correctly specified.
- A lower **AIC** (Akaike Information Criterion) indicates a better model fit.
- In our model, residual deviance decreased by 564.8 with a loss of 3 degrees of freedom, indicating a statistically significant regression effect (compared with $\chi^2_{3,\alpha} = 7.81$).

Summary table of estimated coefficients includes

- The estimated log-odds coefficients, Standard errors, Wald z-statistics, and associated p-values.
- All three predictors (Cl.thickness, Cell.size, Cell.shape) are statistically significant ($p < 0.001$).

```
# Extract odds ratio estimate with CIs and P-value
or_df <- tidy(mlogit, conf.int = TRUE, exponentiate = TRUE)
or_df <- or_df[or_df$term != "(Intercept)", ]
print(or_df)
```

```
## # A tibble: 3 x 7
##   term         estimate std.error statistic    p.value conf.low conf.high
##   <chr>           <dbl>     <dbl>     <dbl>      <dbl>    <dbl>     <dbl>
## 1 Cl.thickness     1.68     0.112      4.63 0.00000359     1.37      2.13
## 2 Cell.size        2.03     0.211      3.36 0.000773       1.39      3.16
## 3 Cell.shape       2.16     0.197      3.90 0.0000980      1.49      3.23
```

Odds Ratio Interpretation

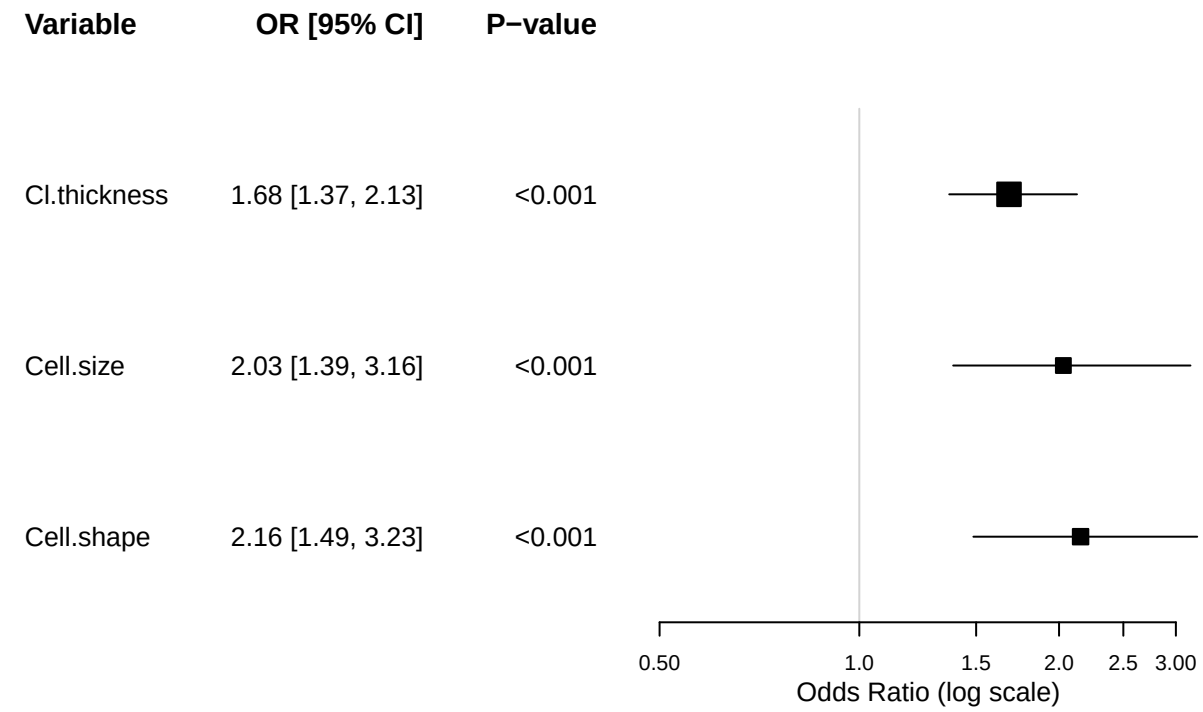
- For every one-unit increase in `Cl.thickness`, the log-odds of being malignant increases by 0.5195.
- This corresponds to an odds ratio of $\exp(0.5195) = 1.68$, meaning the odds of being malignant are 1.68 times higher.
- Similarly, with one unit increase in `Cell.size` (or `Cell.shape`), the odds of being malignant increases twice since $e^{0.7081} = 2.03$ (or more than twice, $e^{0.7679} = 2.15$).

```
# Create a label matrix `Variable`, `OR [95% CI]` and `P-value`
label_matrix <- cbind(
  Variable = or_df$term,
  OR_CI = sprintf("%.2f [%.2f, %.2f]", or_df$estimate, or_df$conf.low, or_df$conf.high),
  P = ifelse(or_df$p.value < 0.001, "<0.001", sprintf("%.3f", or_df$p.value))
)

# Add header row
label_matrix <- rbind(c("Variable", "OR [95% CI]", "P-value"), label_matrix)

# Plot with forestplot()
forestplot(
  labeltext = label_matrix[-1, ],
  mean  = or_df$estimate,
  lower = or_df$conf.low,
  upper = or_df$conf.high,
  xlog = TRUE,
  zero = 1,
  clip = c(0.5, 5),
  xticks = c(0.5, 1, 1.5, 2, 2.5, 3),
  xlab = "Odds Ratio (log scale)"
) |>
  fp_add_header(
    Variable = "Variable",
    OR_CI = "OR [95% CI]",
    P = "P-value"
  ) |>
```

```
  fp_set_style(
    box = "black",
    line = "black",
    summary = "black",
    align = c("l", "r", "r"),
    txt_gp = fpTxtGp(
      label = gpar(cex = 0.8),
      ticks = gpar(cex = 0.7),
      xlab  = gpar(cex = 0.8),
      title = gpar(cex = 0.9)
    )
  )
```



7. Model Prediction and Evaluation

By predicting the probability $\mathbb{P}(Y = 1)$ based on given features, the logistic regression can be applied to determine whether individual patients belong to a particular class or not. To obtain predictions and optionally estimate standard errors of those predictions from a fitted generalized linear model object, we use the function `predict()` with syntax below:

```
predict(object, newdata = NULL,
        type = c("link", "response", "terms"),
        se.fit = FALSE, dispersion = NULL, ...)
```

where the `type` of prediction is required. The default is on the scale of the linear predictors; the alternative `"response"` is on the scale of the response variable. Thus for a default logistic model the default predictions are of log-odds (probabilities on logit scale), and `type = "response"` gives the predicted probabilities $\mathbb{P}(Y = 1)$.

```
# Predict probabilities on test set
pred.p <- predict(mlogit, newdata = test.data, type = "response")
```

The output of `pred.p` is the probability $\mathbb{P}(Y = 1)$. We can select a threshold value (say 0.5). If the probability is greater than this threshold value, the event $Y = 1$ is predicted to happen and the patient is classified into the malignant class; otherwise $Y = 0$ is predicted to happen and the patient is classified into the benign class.

```
# Convert probabilities to binary prediction (threshold = 0.5)
y_pred <- factor(ifelse(pred.p > 0.5, 1, 0), levels = c(0, 1))

# Accuracy
accuracy <- mean(y_pred == test.data$y)

tab <- table(y_pred, test.data$y)
tab
```

```
##
## y_pred  0  1
##      0 82  4
##      1  4 47
```

Confusion matrix is a table used to **describe the performance of a classification model** on a set of test data for which the true outcomes are known. It compares the actual outcomes in the test data set to the predicted outcomes. In the confusion matrix above, two diagonal entries 82 and 47 are the numbers of true negative and true positive cases. 4 cases are false positive (predicted class=1 but the true class=0), while another 4 cases are false negative (the truth=1 but predicted as 0).

Accuracy: (82+47)/(82+47+8)=94.16%, describing overall how often the classification is correct.