使用R进行logistic回归分析

Alex / 2025-05-13 / free_learner@163.com / AlexBrain.cn

本文介绍在R中使用logistic回归模型预测二分类变量的基本方法。

一、背景

Logistic回归模型和线性回归模型在形式上很接近，区别在于对于线性模型，响应变量（response variable）是连续变量，而对于logistic回归，响应变量是二分类变量：

$$log(\frac{p}{1-p}) = \boldsymbol{X}\boldsymbol{\beta}$$

在上式中，p表示事件发生（比如患病）的概率，$1-p$表示事件未发生（比如不患病）的概率，$\frac{p}{1-p}$表示odds（一般翻译为“几率”），$log(\frac{p}{1-p})$表示对odds取自然对数，一般称为log odds或logit；$\boldsymbol{X}$和$\boldsymbol{\beta}$分别表示预测变量（predictor variable）构成的矩阵和对应的回归系数。在线性回归中，回归系数（假设用β_i表示）表示某个预测变量（假设用x_i表示）增加一个单位，响应变量变化β_i个单位；在logistic回归中，回归系数表示x_i增加一个单位，log odds变化β_i个单位。e^{β_i}表示odds ratio (OR，几率比)，表示x_i增加一个单位，事件发生的几率变化为原来的e^{β_i}倍。比如，e^{β_i}等于1.5，表示x_i增加一个单位，事件发生的几率变化为原来的1.5倍，即增加0.5倍。

Logistic回归模型既可以用于统计推断的目的，关注回归系数的方向和显著性，也可以用于预测，关注在新数据上的预测表现。

二、样例数据

我这里测试使用的样例数据为`mlbench`包里的`PimaIndiansDiabetes2`数据集，共包含9个变量。我们根据前8个变量来预测最后一个变量（是否患有糖尿病）。由于包含缺失值，因此先去除含有缺失值的样本，使用剩下的数据来构建模型。此外，将80%的数据作为训练集，20%的数据作为测试集。

```r
## Load and clean data
data("PimaIndiansDiabetes2", package = "mlbench")
in_idx <- complete.cases(PimaIndiansDiabetes2)
all_dat <- PimaIndiansDiabetes2[in_idx,]
## Split the whole dataset into training and test sets
N <- nrow(all_dat)
set.seed(100)
train_idx <- sample(c(1:N), round(0.8*N))
train_dat <- all_dat[train_idx, ]
test_dat <- all_dat[-train_idx, ]
```

三、训练模型

```r
## Train logistic regression model
logit_mod <- glm(diabetes ~ ., family = binomial(link = "logit"), data = train_dat)
summary(logit_mod)
## Calculate odds ratio
exp(coef(logit_mod))
```

在上面的代码中，使用R自带的`glm`函数进行模型拟合，对于响应变量的编码，可以将事件发生和未发生编码为数字1和0，也可以编码为一个因子变量，第一个水平表示事件未发生，第二个水平表示事件发生。模型拟合结果如下：

```
Call:
glm(formula = diabetes ~ ., family = binomial(link = "logit"),
    data = train_dat)

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -9.4668206  1.3343757  -7.095 1.30e-12 ***
pregnant     0.0847091  0.0599348   1.413  0.15755
glucose      0.0377368  0.0062986   5.991 2.08e-09 ***
pressure    -0.0105557  0.0132871  -0.794  0.42694
triceps      0.0031191  0.0190381   0.164  0.86986
insulin     -0.0008424  0.0013783  -0.611  0.54108
mass         0.0894500  0.0297206   3.010  0.00262 **
pedigree     0.7272649  0.4615445   1.576  0.11509
age          0.0342975  0.0201135   1.705  0.08816 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 402.89  on 313  degrees of freedom
Residual deviance: 282.51  on 305  degrees of freedom
AIC: 300.51

Number of Fisher Scoring iterations: 5
```

从上面的结果可以看到，`glucose`和`mass`对于预测`diabetes`具有显著贡献。每个回归系数对应的OR值如下：

```
 (Intercept)     pregnant      glucose     pressure      triceps      insulin         mass
pedigree          age
7.737703e-05 1.088400e+00 1.038458e+00 9.894998e-01 1.003124e+00 9.991579e-01 1.093573e+00
2.069413e+00 1.034892e+00
```

从上面的结果可以看到，`glucose`每增加一个单位，患糖尿病的几率增加3.8%倍。

四、测试模型

```r
library(caret)
## Test model on unseen data
pred_prob <- predict(logit_mod, test_dat, type="response")
pred_class <- factor(ifelse(pred_prob > 0.5, 'pos', 'neg'), levels = c('neg', 'pos'))
caret::confusionMatrix(table(Predicted=pred_class, Real=test_dat$diabetes), positive = 'pos')
```

注意在使用`predict`函数时，需要加上`type="response"`，返回的数值表示事件发生的概率，一般选择概率大于0.5为阈值，将预测值划分为两类。使用`caret`包里`confusionMatrix`函数计算一些反映预测表现的指标，结果如下：

```
Confusion Matrix and Statistics

         Real
Predicted neg pos
      neg  48  10
      pos   7  13

               Accuracy : 0.7821
                 95% CI : (0.6741, 0.8676)
    No Information Rate : 0.7051
    P-Value [Acc > NIR] : 0.08327

                  Kappa : 0.4552

 Mcnemar's Test P-Value : 0.62763

            Sensitivity : 0.5652
            Specificity : 0.8727
         Pos Pred Value : 0.6500
         Neg Pred Value : 0.8276
             Prevalence : 0.2949
         Detection Rate : 0.1667
   Detection Prevalence : 0.2564
      Balanced Accuracy : 0.7190

       'Positive' Class : pos
```