丁香园

丁香公开课
CLASS.DXY.CN

临床预测模型构建&机器学习(R语言进阶)

第3章 Logistic回归与判别分析R语言实现

周支瑞

CONTENT

背景知识

- 分类问题最好使用0和1之间的概率进行建模。有很多不同的函数可以实现这个功能，本章重点讨论Logistic回归模型与判别分析。
- Logistic模型以对数发生概率比为响应变量进行线性拟合，即*log(P(Y)/1-P(Y))=B0+B1*。这里的系数是通过极大似然估计得到，而不是通过OLS。极大似然的直观意义就是：找到一对B0和B1的估计值，使它们产生的对观测的预测概率尽可能接近Y的实际观测结果，这就是所谓的似然性。
- R语言可以实现极大似然估计，通过找到最优的B值组合来使似然性最大化。

案例分析

- [案例] 威斯康星大学威廉·沃尔伯格博士1990年发布了一个乳腺癌数据集，为明确肿瘤活检结果是良性还是恶性。研究者使用细针穿刺（FNA）技术收集样本，病理学家对活体组织进行检查，确定诊断结果（良性或恶性）。良性的乳腺肿瘤并不危险，恶性肿瘤必须进行干预。误诊为恶性（false positive,假阳性）会导致昂贵但不必要的治疗费用，还会使患者背负巨大的心理和生理负担。误诊为良性（false negative,假阴性）会使患者得不到应有的治疗，造成癌细胞扩散，引起过早死亡。乳腺癌患者的早期干预可大大提高存活率。我们的任务就是开发尽可能准确的机器学习诊断算法，帮助医疗团队确定肿瘤性质。

数据准备

- 数据集包含699个患者的组织样本，保存在有11个变量的数据框中，如下所示。

  ID：样本编码
  V1：细胞浓度
  V2：细胞大小均匀度
  V3：细胞形状均匀度
  V4：边缘黏着度
  V5：单上皮细胞大小
  V6：裸细胞核（16个观测值缺失）
  V7：平和染色质
  V8：正常核仁
  V9：有丝分裂状态
  class：肿瘤诊断结果，良性或恶性；这就是我们要预测的结果变量。

- 对9个特征进行了评分和编码，评分的范围是1~10。可以在R的MASS包中找到该数据框，名为biopsy。我们加载这个数据框，确认数据结构，将变量重命名为有意义的名称，还要删除有缺失项的观测，然后即可开始对数据进行可视化探索。代码如下：

数据准备 代码

```
library(MASS)
data(biopsy)
str(biopsy)
biopsy$ID = NULL
names(biopsy) = c("thick", "u.size", "u.shape", "adhsn",
                  "s.size", "nucl", "chrom", "n.nuc", "mit", "class")
names(biopsy)
biopsy.v2 <- na.omit(biopsy)
y <- ifelse(biopsy.v2$class == "malignant", 1, 0)
```

数据探索与可视化 代码


```
library(reshape2)
library(ggplot2)
biop.m <- melt(biopsy.v2, id.var = "class")
ggplot(data = biop.m, aes(x = class, y = value)) +
  geom_boxplot() +
  facet_wrap(~ variable, ncol = 3)
library(corrplot)
bc <- cor(biopsy.v2[ ,1:9]) #create an object of the features
corrplot.mixed(bc)
```

划分训练集和测试集 代码

```
set.seed(123) #random number generator
ind <- sample(2, nrow(biopsy.v2), replace = TRUE, prob = c(0.7, 0.3))
train <- biopsy.v2[ind==1, ] #the training data set
test <- biopsy.v2[ind==2, ] #the test data set
str(test) #confirm it worked
table(train$class)
table(test$class)
```

模型构建与模型评价

- 首先用所有输入变量建立一个Logistic回归模型，然后用最优子集法对特征进行削减。在此之后，我们会试验判别分析和多元自适应回归样条（MARS）方法。
- R中的glm()函数可以拟合广义线性模型，这一系列模型包括logistic回归模型。代码的语法和lm()函数相似，其中一个较大区别是我们必须在函数中使用family=binomial这个参数，该参数告诉R运行logistic回归，而不是其他的广义线性模型。
- 在训练数据集上使用所有特征建立一个模型，看看这个模型在测试数据集上运行的效果。代码如下：

模型构建 代码

```
full.fit <- glm(class ~ ., family = binomial, data = train)
summary(full.fit)
confint(full.fit)
exp(coef(full.fit))
library(car)
vif(full.fit)#共线性识别
train.probs <- predict(full.fit, type = "response")
train.probs[1:5] #inspect the first 5 predicted probabilities
contrasts(train$class)
```

模型评价

- 评价模型在训练集上执行的效果，然后再评价它在测试集上的拟合程度。快速实现评价的方法是生成一个混淆矩阵。在后面的章节中，我们使用的混淆矩阵是由caret包实现的，InformationValue包也可以实现混淆矩阵。这时，我们需要用0和1来表示结果。函数区别良性结果和恶性结果使用的默认值是0.50，也就是说，当概率大于等于0.50时，就认为这个结果是恶性的：

模型评价 代码

```
library(InformationValue)
trainY <- y[ind==1]
testY <- y[ind==2]
confusionMatrix(trainY, train.probs)
# optimalCutoff(trainY, train.probs)
misClassError(trainY, train.probs)
confusionMatrix(trainY, train.probs)
test.probs <- predict(full.fit, newdata = test, type = "response")
misClassError(testY, test.probs)
confusionMatrix(testY, test.probs)
```

模型进一步评价 -- 交叉验证

- 交叉验证目的是提高测试集上的预测正确率，以及尽可能避免过拟合。K折交叉验证的做法是将数据集分成K个相等的等份，每个等份称为一个K子集（K-set）。算法每次留出一个子集，使用其余K-1个子集拟合模型，然后用模型在留出的那个子集上做预测。将上面K次验证的结果进行平均，可以使误差最小化，并且获得合适的特征选择。你也可以使用留一交叉验证方法，这里的K等于N。模拟表明，LOOCV可以获得近乎无偏的估计，但会有很高的方差。所以，大多数机器学习专家都建议将K值定为5或10。
- bestglm包可以自动进行交叉验证，这个包依赖于我们在线性回归中使用过的leaps包。交叉验证的语法和数据格式存在注意事项，我们如下按部就班地进行：

交叉验证 代码

```
library(bestglm)
X <- train[, 1:9]
Xy <- data.frame(cbind(X, trainY))
bestglm(Xy = Xy, IC = "CV", CVArgs = list(Method = "HTF", K = 10, REP = 1),
          family=binomial)
reduce.fit <- glm(class ~ thick + u.size + nucl, family = binomial, data = train)

test.cv.probs = predict(reduce.fit, newdata = test, type = "response")
misClassError(testY, test.cv.probs)
confusionMatrix(testY, test.cv.probs)

bestglm(Xy = Xy, IC = "BIC", family = binomial)
bic.fit <- glm(class ~ thick + adhsn + nucl + n.nuc,
          family = binomial, data = train)
test.bic.probs = predict(bic.fit, newdata = test, type = "response")
misClassError(testY, test.bic.probs)
confusionMatrix(testY, test.bic.probs)
```

CONTENT

判别分析 背景知识

- 判别分析又也是一项常用分类技术。当分类很确定时，判别分析可以有效替代logistic回归。当分类结果很确定时， logistic回归的估计结果可能是不稳定的，即置信区间很宽，不同样本之间的估计值会有很大变化（James，2013）。判别分析会比logistic做得更好，泛化能力更强。反之，如果特征和结果变量之间具有错综复杂的关系，判别分析在分类任务上的表现就会非常差。
- 在乳腺癌这个例子中， logistic回归在训练集和测试集上的表现都非常好，分类的结果并不确定。出于同logistic回归进行比较的目的，我们研究一下判别分析，包括线性判别分析和二次判别分析。

判别分析 基本原理

- 判别分析使用贝叶斯定理确定每个观测属于某个类别的概率。如果你有两个类别，比如良性和恶性，判别分析会计算观测分别属于两个类别的概率，然后选择高概率的类别作为正确的类别。贝叶斯定理定义了在X已经发生的条件下Y发生的概率 -- 等于Y和X同时发生的概率除以X发生的概率，公式如下：

  Y/X的概率= P(X+Y)/P(X)

判别分析 基本原理

分子表示一个具有某些特征的观测属于某个分类水平的可能性，分母表示一个具有这些特征的观测属于所有分类水平的可能性。同样地，分类原则认为如果X和Y的联合分布已知，那么给定X后，决定观测属于哪个类别的最佳决策是选择那个有更大后验概率的类别。获得后验概率的过程如下所示：

1. 收集已知类别的数据。
2. 计算先验概率 -- 代表属于某个类别的样本的比例。
3. 按类别计算每个特征的均值。
4. 计算每个特征的方差　协方差矩阵。在线性判别分析中，这会是一个所有类别的混合矩阵，给出线性分类器；在二次判别分析中，会对每个分类建立一个方差　协方差矩阵。
5. 估计每个分类的正态分布（高斯密度）。
6. 计算discriminant函数，作为一个新对象的分类原则。
7. 根据discriminant函数，将观测分配到某个分类。

判别分析 优势与不足

- 尽管线性判别分析简单而又优雅，但具有局限性。线性判别分析假设每种类别中的观测服从多元正态分布，并且不同类别之间的具有相同的协方差。二次判别分析仍然假设观测服从正态分布，但假设每种类别都有自己的协方差。当你放宽相同协方差假设，就意味着允许二次项进入判别分数的计算，这在线性判别分析中是不可能的。重要的是，二次判别分析技术比logistic回归更灵活，同时还要牢记偏差/方差权衡的问题。使用更有灵活的技术可以得到偏差更小的结果，但很可能具有更高的方差。和很多灵活的技术一样，需要一个高鲁棒性的训练数据集来降低高分类方差。
- 线性判别分析（LDA）可以用MASS包实现，我们为了使用biopsy数据集已经加载了这个包。LDA的语法和lm()以及glm()函数非常相似。开始LDA模型拟合，R代码如下：

线性判别分析(LDA)代码


```
lda.fit <- lda(class ~ ., data = train)
lda.fit
plot(lda.fit, type="both")
train.lda.probs <- predict(lda.fit)$posterior[, 2]
misClassError(trainY, train.lda.probs)
confusionMatrix(trainY, train.lda.probs)

test.lda.probs <- predict(lda.fit, newdata = test)$posterior[, 2]
misClassError(testY, test.lda.probs)
confusionMatrix(testY, test.lda.probs)
```

二次判别分析(QDA)代码

```
qda.fit <- qda(class ~ ., data =train)
qda.fit
train.qda.probs <- predict(qda.fit)$posterior[, 2]
misClassError(trainY, train.qda.probs)
confusionMatrix(trainY, train.qda.probs)

test.qda.probs <- predict(qda.fit, newdata = test)$posterior[, 2]
misClassError(testY, test.qda.probs)
confusionMatrix(testY, test.qda.probs)
```

多元自适应回归样条方法 代码

```
library(earth)
set.seed(1)
earth.fit <- earth(class ~ ., data = train,
                   pmethod = "cv",
                   nfold = 5,
                   ncross = 3,
                   degree = 1,
                   minspan = -1,
                   glm=list(family=binomial)
                   )
summary(earth.fit)
plotmo(earth.fit)
plotd(earth.fit)
evimp(earth.fit)
test.earth.probs <- predict(earth.fit, newdata = test, type = "response")
misClassError(testY, test.earth.probs)
confusionMatrix(testY, test.earth.probs)
```

模型选择

- 我们从模型中计算混淆矩阵和错误率，为的就是有一个选择依据，但并不全面。对于分类模型的比较，受试者工作特征（ROC）曲线是一个很有效的工具。简言之，ROC基于分类器的性能对其进行可视化、组织和选择（Fawcett，2006）。在ROC曲线中，Y轴是真阳性率（TPR），X轴是假阳性率（FPR）。计算过程简单，如下所示：

  TPR = 正确分类的阳性样本数/所有阳性样本数

  FPR = 错误分类的阴性样本数/所有阴性样本数

ROC分析 代码

```
library(ROCR)
bad.fit <- glm(class ~ thick, family = binomial, data = train)
test.bad.probs <- predict(bad.fit, newdata = test, type = "response") #save probabilities
pred.full <- prediction(test.probs, test$class)
perf.full <- performance(pred.full, "tpr", "fpr")
plot(perf.full, main = "ROC", col = 1)
pred.bic <- prediction(test.bic.probs, test$class)
perf.bic <- performance(pred.bic, "tpr", "fpr")
plot(perf.bic, col = 2, add = TRUE)
```

ROC分析 代码续


```
pred.bad <- prediction(test.bad.probs, test$class)
perf.bad <- performance(pred.bad, "tpr", "fpr")
plot(perf.bad, col = 3, add = TRUE)
pred.earth <- prediction(test.earth.probs, test$class)
perf.earth <- performance(pred.earth, "tpr", "fpr")
plot(perf.earth, col = 4, add = TRUE)
legend(0.6, 0.6, c("FULL", "BIC", "BAD", "EARTH"), 1:4)
performance(pred.full, "auc")@y.values
performance(pred.bic, "auc")@y.values
performance(pred.bad, "auc")@y.values
performance(pred.earth, "auc")@y.values
```

小结

- 本章研究了如何使用基于概率的线性模型预测定性响应变量，介绍了三种计算方法：Logistic回归、判别分析和MARS。
- 除此之外还介绍了ROC分析，这是一种可视化的模型选择技术。我们还简要讨论了需要注意的模型选择问题和权衡问题。

请在此处输入小标题

感谢观看

THANKS

丁香园特邀讲师 周支瑞