US college data modeling and classification

Minh Chau

Context

U.S. News and World Report ranks colleges in the United States based on several factors, including graduation rates and retention rates. In this question, your goal is to classify colleges into a “high application” category based on several factors using logistic regression.

- Dataset required: uscollege.csv Load in the data for this question. There are 777 observations and the key variables are :
- CollegeName: Name of the US College
- Private: A factor indicating private or public university (Yes = private, No = public)
- Apps: Number of applications received
- Enroll: Number of new students enrolled
- Top10perc: Percentage of new students from top 10% of H.S. class
- Top25perc: Percentage of new students from top 25% of H.S. class
- Room.Board: Room and board costs
- Books: Estimated book costs
- Personal: Estimated personal spending
- perc.alumni: Percentage of alumni who donate
- Grad.Rate: Graduation rate

Part 1 ***Purpose: Understanding of key concepts such as quantitative independent variables and logistic regression function. Creating dummy variables and fit logistic regression models.***

1. Create a new binary variable `high_application` . This variable should be 1 if the college’s App is more than 10,000 applications, and 0 otherwise. Add it to the data frame as a new column.

```
##   high_application == "1"   n
## 1                   FALSE 732
## 2                    TRUE  45
```

We have 732 university with the number of applications smaller than 10000 and 45 universities that have the number of application greater or equal to 10000.

2. Since `Private` is a qualitative variable, convert the `Private` variable into a factor, setting “No” as the reference group. Provide the R code.

- List all unique values of the ‘ `Private` ’ and the categorical frequency count.
- Check and write down the indicator variable coding scheme used by R for ‘ `Private` ’.

```
uscollege$Private <- factor(uscollege$Private, levels = c("No", "Yes"), labels = c("N
o", "Yes"))
uscollege$Private <- relevel(uscollege$Private, ref = "No")
unique(uscollege$Private)
```

```
## [1] No  Yes
## Levels: No Yes
```

```
uscollege %>% count(uscollege$Private)
```

```
##   uscollege$Private   n
## 1                No 212
## 2               Yes 565
```

```
contrasts(uscollege$Private)
```

```
##     Yes
## No    0
## Yes   1
```

Reference group = “No”

Private = { 1 if Private = “Yes” 0 otherwise }

3. Fit a logistic regression model, to predict `high_application` , using ‘Private’, ‘Top10perc’, ‘Grad.Rate’ and ‘Room.Board’ as predictors. Provide the summary. Save it as mod1. Write down the equation of the logistic regression model.

```
mod1 <- glm(high_application ~ Private + Top10perc + Grad.Rate + Room.Board, data = u
scollege, family = binomial)
summary(mod1)
```

```
##
## Call:
## glm(formula = high_application ~ Private + Top10perc + Grad.Rate +
##     Room.Board, family = binomial, data = uscollege)
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -8.3403980  1.2427677  -6.711 1.93e-11 ***
## PrivateYes  -4.5696343  0.6481399  -7.050 1.78e-12 ***
## Top10perc    0.0501083  0.0110129   4.550 5.37e-06 ***
## Grad.Rate    0.0233952  0.0157615   1.484    0.138
## Room.Board   0.0009422  0.0002354   4.002 6.29e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 343.73  on 776  degrees of freedom
## Residual deviance: 206.47  on 772  degrees of freedom
## AIC: 216.47
##
## Number of Fisher Scoring iterations: 7
```

log(P / (1 - P)) = β0 + β1 * PrivateYes + β2 * Top10perc + β3 * Grad.Rate + β4 * Room.Board => The models predict log-odds (logit) of High application occurring: log-odds(P(high_application > 10000|X)) = -8.34 - 4.57 * PrivateYes + 0.05 * Top10perc + 0.023 * Grad.Rate + 0.0009 * Room.Board

Interpret the estimated parameters of the fitted logistic regression model. It’s important to distinguish this interpretation from that of a multivariate model.

4. Write down the regression coefficient estimate for `PrivateYes` in the logistic regression model.

- Explain the **odds ratio** of a private university having a high application rate (above 10,000 applications) **compared to public universities**.
- Assess its significance and explain what it implies. (3 marks)

The odds ratio (OR) tells us how much more likely private colleges are to have high applications compared to public colleges.

We have:

log-odds(P(high_application > 10000|X)) = -8.34 - 4.57 * PrivateYes + 0.05 * Top10perc + 0.023 * Grad.Rate + 0.0009 * Room.Board

Holding all other variables constant,

Being a private school leads to a decrease of 4.57 in the log-odds of having more than 10,000 applications compared to a public school.

In terms of odds: Being private decreases the odds of having a high number of applications by a factor of exp(-4.57) = 0.01036. This means that the odds of a private school having more than 10,000 applications is only 0.01036 times the odds for a public university.

Statistical Significance: The p-value for the PrivateYes coefficient is 1.78e-12, which is much smaller than the 0.001 significance level (1%). This indicates strong statistical evidence that there is a significant difference in the log-odds of having high applications between private and public universities.

Part 2

Purpose: understand computing the predicted probability and set a cut-off threshold to classify the binary outcomes. Understand how to use the confusion matrix to evaluate the performance of the logistic regression model.

1. Use your logistic regression model to predict the probability of `high_application` using predict (glm_object ,type='response').

- Define a cut-off : 1 for predicted probabilities >= 0.35, otherwise 0.
- Save the binary predictions in a variable 'predicted_high_application'
- Calculate the number of "positives" and "negatives" predictions made by the model.

```
predicted_probabilities = predict(mod1, type='response')

uscollege$predicted_high_application<- ifelse(predicted_probabilities >= 0.35, 1, 0)
#Number of positive
uscollege %>% count(uscollege$predicted_high_application)
```

```
##   uscollege$predicted_high_application   n
## 1                                    0 746
## 2                                    1  31
```

The model predicts 31 positives (universities with ≥ 10,000 applications) and 746 negatives (universities with < 10,000 applications). Since the number of universities with ≥ 10,000 applications is very small in the dataset, this causes the prediction to have a large false margin. Specifically, out of the 45 actual positives, only 31 are correctly predicted, leading to a false margin of:((45-31) /45 ≈ 31%)

This means that 31% of the actual positive cases are misclassified as negatives, resulting in a relatively high rate of false negatives.

2. Build a confusion matrix for the `high_application` dependent variable, where a "high application rate" (i.e., more than 10,000 applications) is the positive event. Report and explain the number of false positives and false negatives from the confusion matrix. Based on your findings, what conclusions can you draw about the performance of the binary classifier?

```
library(caret)
```

```
## Loading required package: ggplot2
```

```
## Loading required package: lattice
```

```
conf_matrix <- confusionMatrix(table(uscollege$predicted_high_application, uscollege
$high_application), positive = '1')
conf_matrix
```

```
## Confusion Matrix and Statistics
##
##
##       0   1
##   0 718  28
##   1  14  17
##
##                Accuracy : 0.9459
##                  95% CI : (0.9276, 0.9608)
##     No Information Rate : 0.9421
##     P-Value [Acc > NIR] : 0.35777
##
##                   Kappa : 0.42
##
##  Mcnemar's Test P-Value : 0.04486
##
##             Sensitivity : 0.37778
##             Specificity : 0.98087
##          Pos Pred Value : 0.54839
##          Neg Pred Value : 0.96247
##              Prevalence : 0.05792
##          Detection Rate : 0.02188
##    Detection Prevalence : 0.03990
##       Balanced Accuracy : 0.67933
##
##        'Positive' Class : 1
##
```

There are 28 false positives and 14 false negatives. The specificity is high (98.09%), meaning the model accurately predicts universities with fewer than 10,000 applications. However, the sensitivity is lower, at just 37.78%, indicating that the model only correctly predicts universities with more than 10,000 applications about one in three times.

This model performs better at identifying universities with fewer than 10,000 applications.