Statistics 452: Statistical Learning and Prediction

Chapter 4, Part 1: Introduction and Logistic Regression

Brad McNeney

Introduction

Classification Problems

- Instead of a quantitative response, we have a categorical (qualitative) response, such as yes/no.

```r
library(ISLR)
data(Default)
head(Default)
```

```
##   default student   balance    income
## 1      No      No  729.5265 44361.625
## 2      No     Yes  817.1804 12106.135
## 3      No      No 1073.5492 31767.139
## 4      No      No  529.2506 35704.494
## 5      No      No  785.6559 38463.496
## 6      No     Yes  919.5885  7491.559
```

Classification Problems, cont.

- Predicting a categorical response is called classifying.
  - We are assigning the observation to a category or class.
- Classification methods may be based on modelling the probability of class membership.
  - Assign to class with highest probability
  - Modelling class probabilities can be cast as a regression.
- Most popular classifiers are logistic regression, linear discriminant analysis and K-nearest neighbors.

Overview of Classification

- Will use a set of training observations, $\{(x_1, y_1), \ldots, (x_n, y_n)\}$ to build the classifier.
- Use the `Default` data to illustrate.
- The response is `default` on credit card payment (Yes/No), to be predicted by credit card `balance`, annual `income` and `student` status (Yes/No).

```r
summary(Default)
```

```
##  default     student        balance           income     
##  No :9667   No :7056   Min.   :   0.0   Min.   :  772  
##  Yes: 333   Yes:2944   1st Qu.: 481.7   1st Qu.:21340  
##                        Median : 823.6   Median :34553  
##                        Mean   : 835.4   Mean   :33517  
##                        3rd Qu.:1166.3   3rd Qu.:43808  
##                        Max.   :2654.3   Max.   :73554  
```

Default data

```r
dtab <- xtabs(~ default + student,data=Default)
dtab
```

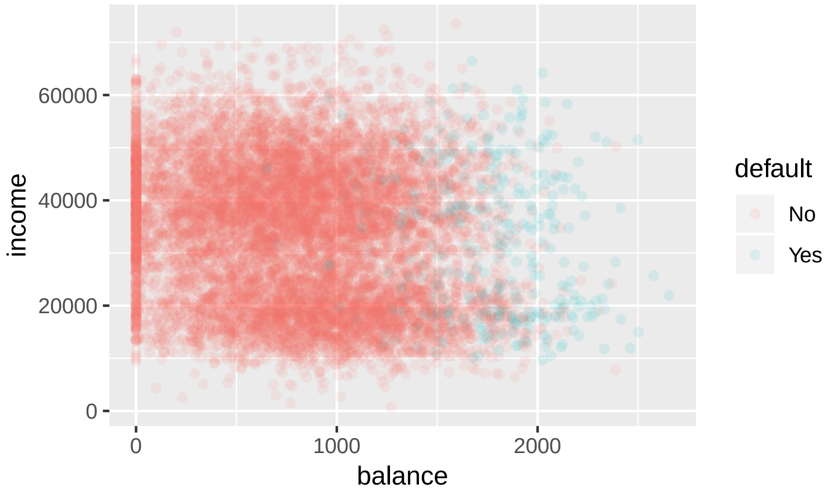
```
##         student
## default   No  Yes
##      No  6850 2817
##      Yes  206  127
```

```r
prop.table(dtab,margin=2)
```

```
##         student
## default          No        Yes
##      No  0.97080499 0.95686141
##      Yes 0.02919501 0.04313859
```

- Overplotting is a problem with a data set this large.

```r
library(ggplot2)
ggplot(Default,aes(x=balance,y=income,color=default)) + geom_point(alpha=0.1)
```



Why Not Linear Regression?

- Numerical codings of categorical variables have no real meaning.
  - Recall `origin` variable in `Auto` data.
- What does it mean for a one unit increase in X_i to be associated with a β_i increase in a categorical response?
- Binary response (success/failure) may be the exception.
  - Say we code success=1, failure=0.
  - Can show that a linear regression predicts the probability of success given X.
  - But probabilities not constrained to be between 0 and 1.

Logistic Regression

Notation and Use as Classifier

- Model $Pr(\mathrm{success}|X)$ as a function $p(X)$ of X.
  - E.G., for the `Default` data, model $Pr(\mathrm{default} = \mathrm{yes}|\mathrm{balance}) = p(\mathrm{balance})$.
- Predict response for new x_0 to be success if $p(x_0) > c$ for some constant c, such as $1/2$.

The Logistic Model.

- Model is linear in the log-odds (logit) of success:

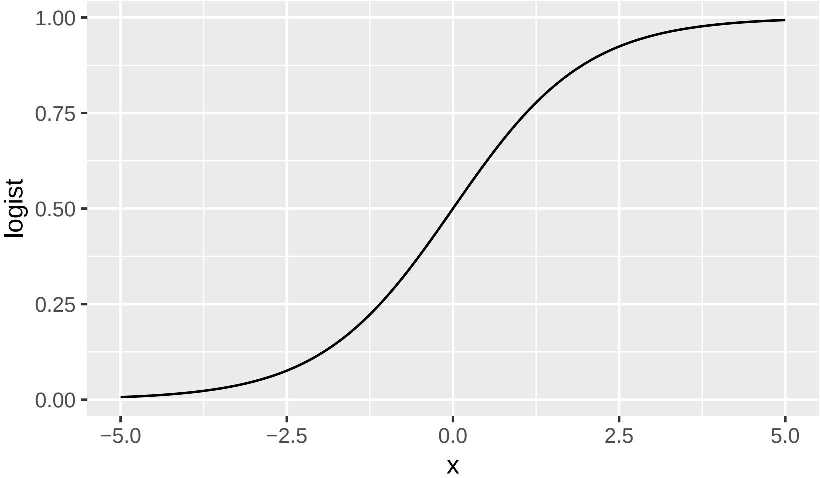
$$\log\left(\frac{p(X)}{1-p(X)}\right) = X\beta = \beta_0 + X_1\beta_1 + \ldots + X_p\beta_p.$$

  - A one unit increase in X_j holding all others fixed is associated with a β_j change in the log-odds.
- Can show that the logit model implies $p(\cdot)$ is the logistic function of $X\beta$:

$$p(X) = \frac{e^{X\beta}}{1+e^{X\beta}}.$$

The Logistic Function

```
seqLen <- 100
x <- seq(from=-5,to=5,length=seqLen)
dd <- data.frame(x=x,logist = exp(x)/(1+exp(x)))
ggplot(dd,aes(x=x,y=logist)) + geom_line()
```



Estimating β by Maximum Likelihood

- Choose $\hat{\beta}$ to maximize the likelihood.
- The likelihood is the probability of the observed data, viewed as a function of β.
- We assume independent observations, which means the probability of the data is the product of the probabilities of each observation.
- For the i^{th}, the probability of a success is $p(x_i)$ and the probability of a failure is $1 - p(x_i)$.
- Thus

$$L(\beta) = \prod_{i:y_i=1} p(x_i) \prod_{i:y_i=0} (1 - p(x_i)) = \prod_i p(x_i)_i^y (1 - p(x_i))^{1-y_i}$$

Maximizing the Log-Likelihood

- The maximizer of the likelihood is the same as the maximizer of the log-likelihood

$$l(\beta) = \log L(\beta) = \sum_i y_i \log p(x_i) + (1 - y_i) \log(1 - p(x_i)).$$

- In many problems, maximizing the log-likeihood is an easier optimization problem.

Fitting a Logistic Regression in R

- Use the `glm()` function (generalized linear models, with logistic as a special case).

```
dfit <- glm(default ~ balance + income + student,
            data=Default, family=binomial())
round(summary(dfit)$coefficients,4)
```

```
##             Estimate Std. Error  z value Pr(>|z|)
## (Intercept) -10.8690     0.4923 -22.0801   0.0000
## balance       0.0057     0.0002  24.7376   0.0000
## income        0.0000     0.0000   0.3698   0.7115
## studentYes   -0.6468     0.2363  -2.7376   0.0062
```

- Income does not predict default; no evidence of interaction between students status and balance (not shown)

Software notes

- `glm()` interface is very similar to `lm()`.
- New argument `family` specifies the type of GLM.
  - `binomial()` is for binary outcomes, or outcomes that are sums of binary variables.

Confounding

```r
dfitS <- glm(default ~ student,data=Default,family=binomial())
dfitSB <- glm(default ~ student+balance,data=Default,family=binomial())
round(coefficients(dfitS),5); round(coefficients(dfitSB),5)
```

```
## (Intercept)  studentYes
##    -3.50413     0.40489

## (Intercept)  studentYes     balance
##   -10.74950    -0.71488     0.00574
```

- Including balance reverses the effect of student:
  - Without balance, students look more likely to default (why?).
  - Adjusting for balance, students are less likely to default; i.e., given a student and a non-student with the same balance, the student is less likely to default.

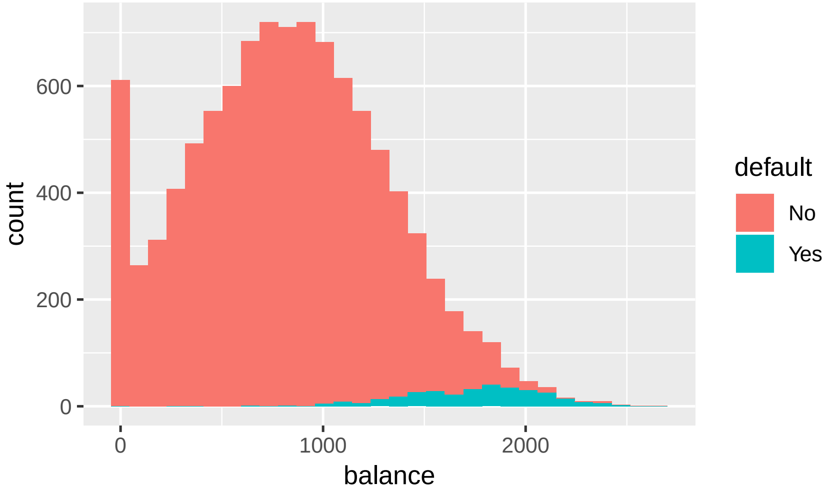
We Do Not Test for Confounding

- Find the percentage change in the coefficient with and without balance:
$$\frac{0.405 - (-0.715)}{|-0.715|} \times 100\% = 156.6$$
- If the change is more than some threshold (e.g., 10%) we say balance confounds the association between default and student.
- By contrast, we **do** test for interaction.

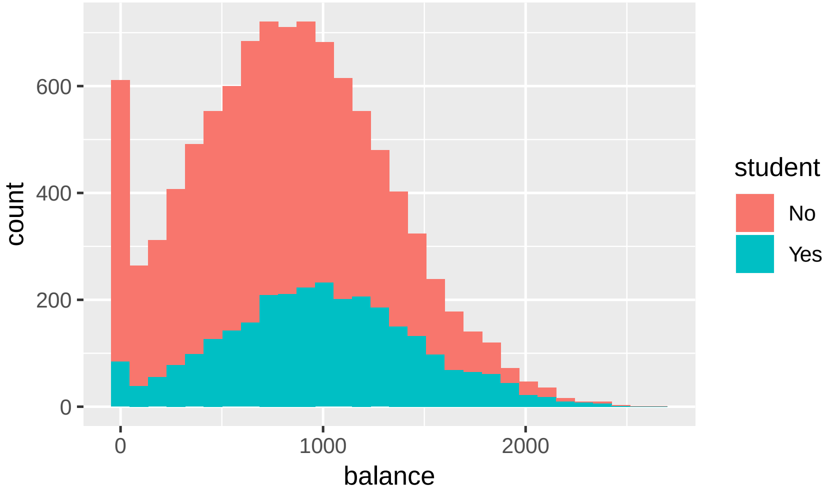
Cause of Confounding

- For a variable like balance to confound the association between default and student, it must be associated with both.
  - Higher balance, higher default rate.
  - Higher balance, more likely student.
  - Looks like students have higher default rate.

```r
ggplot(Default,aes(x=balance,fill=default)) + geom_histogram()
```



```r
ggplot(Default,aes(x=balance,fill=student)) + geom_histogram()
```



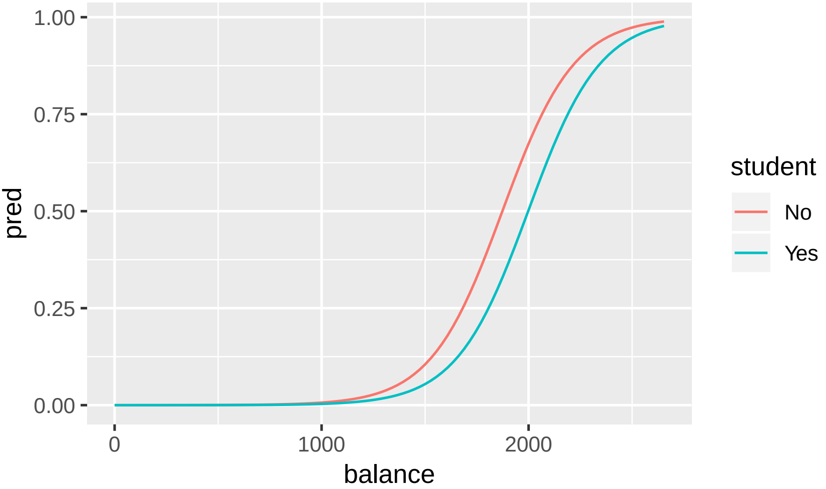
Predictions of $p(X)$.

- Plug in the values of a new x_0 into the fitted equation to get $\hat{p}(x_0)$.

```
bal <- seq(from=min(Default$balance),to=max(Default$balance),len=seqLen)
newdat <- data.frame(balance = c(bal,bal),
                     student = factor(c(rep("Yes",seqLen),
                                        rep("No",seqLen))))
pred <- data.frame(newdat,
                   pred = predict(dfitSB,newdata=newdat,type="response"))
head(pred)
```

```
##     balance student         pred
## 1   0.00000     Yes 1.049739e-05
## 2  26.81134     Yes 1.224320e-05
## 3  53.62268     Yes 1.427936e-05
## 4  80.43402     Yes 1.665415e-05
## 5 107.24536     Yes 1.942387e-05
## 6 134.05670     Yes 2.265421e-05
```

```
ggplot(pred,aes(x=balance,y=pred,color=student)) + geom_line()
```



Logistic Regression for > 2 Response Categories

- Instead of a logistic model we fit a polytomous or multinomial logistic regression.
- Functions such as `multinom()` in the `nnet` package can fit.
  - We won't go into details.
- Text suggests such models are less popular than discriminant analysis, to be discussed in the next set of notes.