Chapter 4: Classification

"regression"

The linear model in Ch. 3 assumes the response variable Y is quantitiative. But in many situations, the response is categorical.

e.g. eye color
cancer diagnosis
whether a car's hwy mpg is above or below the median

In this chapter we will look at approaches for predicting categorical responses, a process known as *classification*.

Classification problems occur often, perhaps even more so than regression problems. Some examples include

1. A person arrives in the emergency room with a set of symptoms that could possibly be attributed to one of three medical conditions. Which of the three conditions does the person have?

2. An online banking system must be able to determine whether or not a transaction is fraudulent, based on user's IP address, past transaction history, etc.

3. Something is in the street in front of the self-driving car that you are riding in. The car must determine if it is a human or another car.

fit a model

As with regression, in the classification setting we have a set of training observations $(x_1, y_1), \dots, (x_n, y_n)$ that we can use to "build a classifier." We want our classifier to perform well on the training data and also on data not used to fit the model (**test data**).

more important.

We will use the `Default` data set in the `ISLR` package for illustrative purposes. We are interested in predicting whether a person will default on their credit card payment on the basis of annual income and credit card balance.

yes or no $\Rightarrow$ categorical.

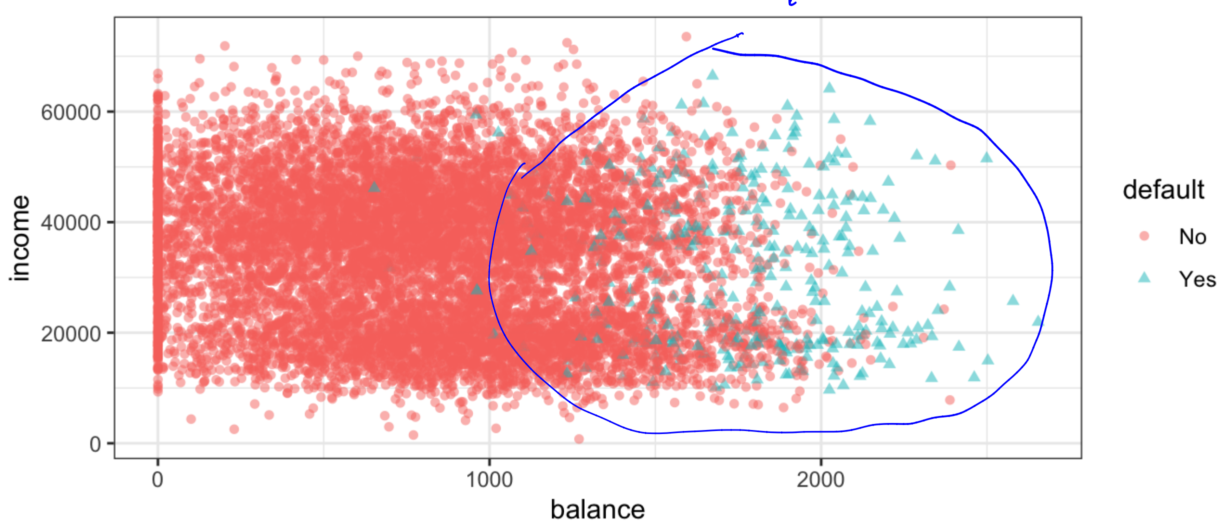
head(Default)

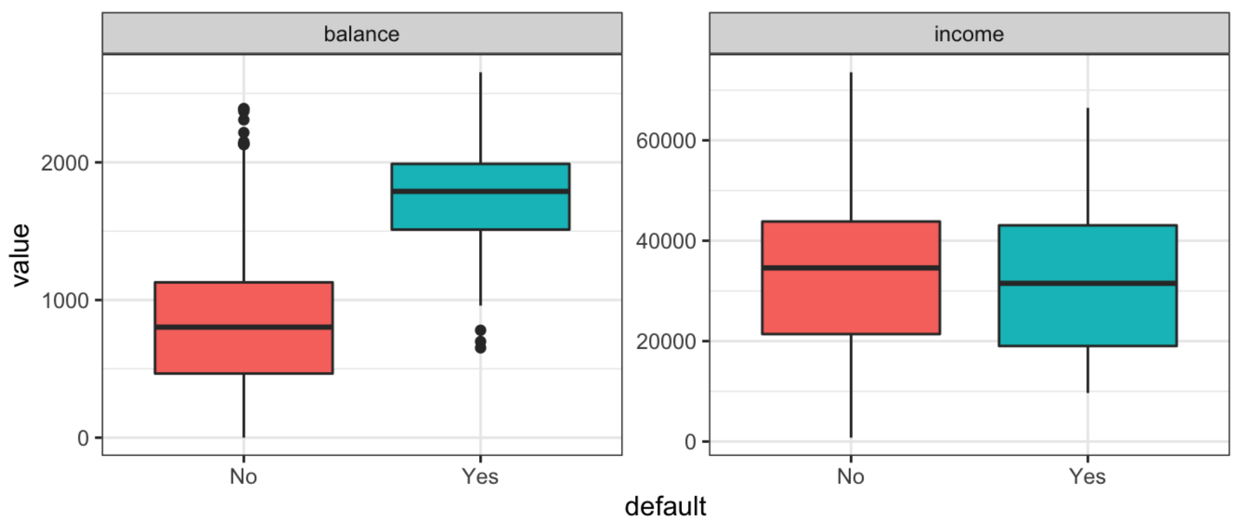
Y

features

```
##   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
```

pretty good separation





pronounced relationship between balance and default.

> in most real world problems the relationship between predictor and response is not so clear.

1 Why not Linear Regression?

I have said that linear regression is not appropriate in the case of a categorical response. Why not?

Let's try it anyways. We could consider encoding the values of `default` in a quantitative repsonse variable Y

$$Y = \begin{cases} 1 & \text{if } \texttt{default} = \text{Yes} \\ 0 & \text{otherwise} \end{cases}$$

Using this coding, we could then fit a linear regression model to predict Y on the basis of `income` and `balance`. This implies an ordering on the outcome, not defaulting comes first before defaulting and insists the difference between these two outcomes is 1 unit. In practice, there is no reason for this to be true.

we could let $Y = \begin{cases} 0 & \text{if default = Yes} \\ 1 & \text{otherwise} \end{cases}$ or $Y = \begin{cases} 1 & \text{default = Yes} \\ 10 & \text{otherwise} \end{cases}$

There is no natural reason w/ 0/1 encoding, but it does have 1 advantage.

Using the dummy encoding, we can get a rough estimate of $P(\texttt{default}|X)$, but it is not guaranteed to be scaled correctly.

doesn't have to be between 0 and 1, but it does provide an ordering.

Real problem: this cannot be easily extended to more than 2 classes.

We can instead use methods specifically formulated for categorical responses.

2 Logistic Regression

Let's consider again the `default` variable which takes values `Yes` or `No`. Rather than modeling the response directly, logistic regression models the *probability* that Y belongs to a particular category. given our feature values

e.g. $P(\underbrace{\text{default} = \text{Yes}}_{Y} \mid \underbrace{\text{balance}}_{X})$

we will abbreviate this as $p(\text{balance}) \in [0, 1]$.

For any given value of `balance`, a prediction can be made for `default`.

e.g. predict default = Yes if $p(\text{balance}) > \boxed{0.5}$

or the CC company could be more conservative and predict default = Yes if $p(\text{balance}) > \boxed{0.1}$

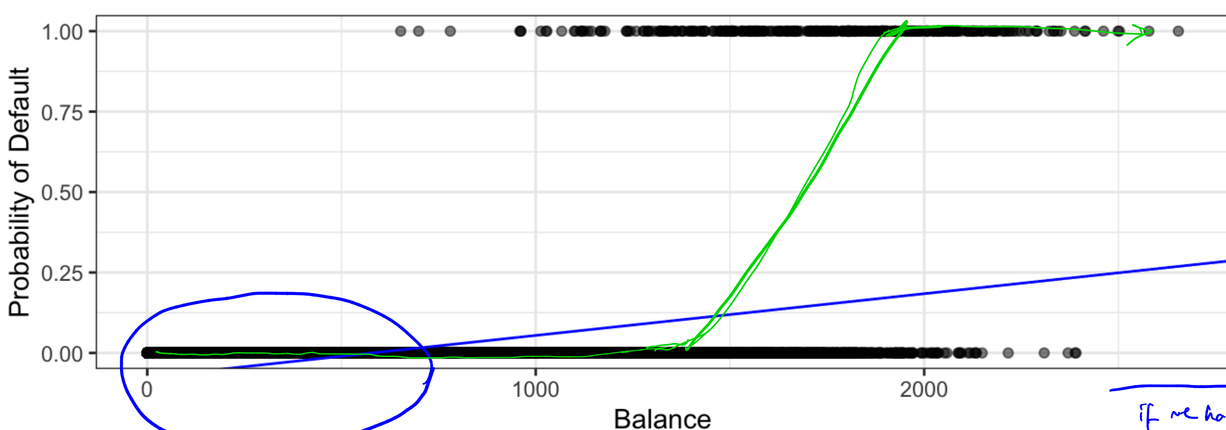
threshold

2.1 The Model

using 0/1 dummy encoded

How should we model the relationship between $p(X) = P(Y = 1|X)$ and X? We could use a linear regression model to represent those probabilities

$$p(X) = \beta_0 + \beta_1 X$$



$\hat{p}(X)$

problem:
for balances close to 0, we predict negative probability of default

neither makes any sense

if we had a very large balance, we would predict probability greater than 1 of defaulting

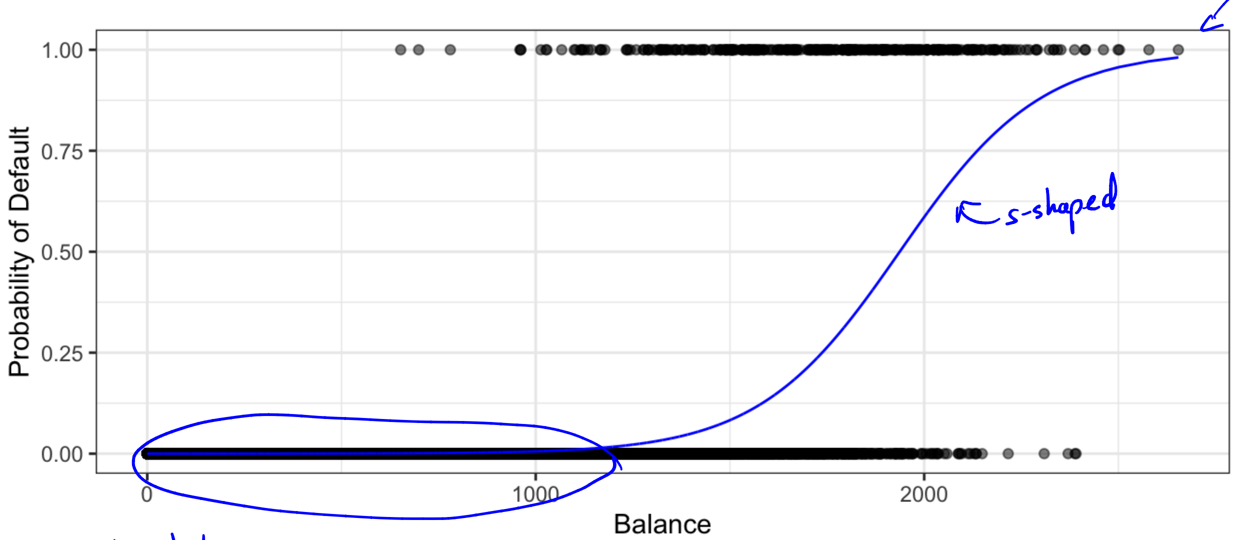
To avoid this, we must model $p(X)$ using a function that gives outputs between 0 and 1 for all values of X. Many functions meet this description, but in *logistic* regression, we use the *logistic* function,

probit regression

$$f(x) = \frac{e^x}{1+e^x}$$

$$p(X) = \frac{e^{\beta_0+\beta_1 X}}{1+e^{\beta_0+\beta_1 X}} \leftarrow \text{logistic function of "regression line" } \beta_0+\beta_1 X.$$

will never predicted probabilities above 1.



low balances
no predict prob. of default
close to zero, but never below!

will always get sensible prediction for $p(X)$.

After a bit of manipulation,

$$\underbrace{\frac{p(X)}{1-p(X)}}_{\text{"odds"}} = e^{\beta_0+\beta_1 X}$$

"odds" $\longrightarrow$ can take any value between 0 and ∞

low prob of default = Yes

high prob. of default = Yes

e.g. $p(X) = 0.2$ (1 in 5 people default) $\Rightarrow$ odds $= \frac{0.2}{1-0.2} = \frac{1}{4}$

By taking the logarithm of both sides we see,

$$\overset{\text{natural log}}{\log}\underbrace{\left(\frac{p(X)}{1-p(X)}\right)}_{\text{"log-odds" "logit"}} = \beta_0 + \beta_1 X \qquad \leftarrow \text{logit is linear in } X$$

Recall from Ch. 3 that β_1 gives the "average change in Y associated with a one unit increase in X." In contrast, in a logistic model,

increasing X by one unit change the log-odds by β_1

$\Longleftrightarrow$

increasing X by one unit multiplies the odds by e^{β_1}

However, because the relationship between $p(X)$ and X is not linear, β_1 does **not** correspond to the change in $p(X)$ associated with a one unit increase in X. The amount that $p(X)$ changes due to a 1 unit increase in X depends on the current value of X.

regardless of the value of X,

if β_1 is positive $\Rightarrow$ increasing X increases $p(X)$

if β_1 is negative $\Rightarrow$ increasing X decreases $p(X)$.

2.2 Estimating the Coefficients

The coefficients β_0 and β_1 are unknown and must be estimated based on the available training data. To find estimates, we will use the method of *maximum likelihood*.

The basic intuition is that we seek estimates for β_0 and β_1 such that the predicted probability $\hat{p}(x_i)$ of default for each individual corresponds as closely as possible to the individual's observed default status.

```
m1 <- glm(default ~ balance, family = "binomial", data = Default)

summary(m1)
```

```
##
## Call:
## glm(formula = default ~ balance, family = "binomial", data = Default)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.2697  -0.1465  -0.0589  -0.0221   3.7589
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.065e+01  3.612e-01  -29.49   <2e-16 ***
## balance      5.499e-03  2.204e-04   24.95   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2920.6  on 9999  degrees of freedom
## Residual deviance: 1596.5  on 9998  degrees of freedom
## AIC: 1600.5
##
## Number of Fisher Scoring iterations: 8
```

2.3 Predictions

Once the coefficients have been estimated, it is a simple matter to compute the probability of `default` for any given credit card balance. For example, we predict that the default probability for an individual with `balance` of $1,000 is

In contrast, the predicted probability of default for an individual with a balance of $2,000 is

2.4 Multiple Logistic Regression

We now consider the problem of predicting a binary response using multiple predictors. By analogy with the extension from simple to multiple linear regression,

Just as before, we can use maximum likelihood to estimate $\beta_0, \beta_1, \ldots, \beta_p$.

```r
m2 <- glm(default ~ ., family = "binomial", data = Default)
summary(m2)
```

```
##
## Call:
## glm(formula = default ~ ., family = "binomial", data = Default)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.4691  -0.1418  -0.0557  -0.0203   3.7383
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.087e+01  4.923e-01 -22.080  < 2e-16 ***
## studentYes  -6.468e-01  2.363e-01  -2.738  0.00619 **
## balance      5.737e-03  2.319e-04  24.738  < 2e-16 ***
## income       3.033e-06  8.203e-06   0.370  0.71152
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2920.6  on 9999  degrees of freedom
## Residual deviance: 1571.5  on 9996  degrees of freedom
## AIC: 1579.5
##
## Number of Fisher Scoring iterations: 8
```

By substituting estimates for the regression coefficients from the model summary, we can make predictions. For example, a student with a credit card balance of \$1,500 and an income of \$40,000 has an estimated probability of default of

A non-student with the same balance and income has an estimated probability of default of

2.5 Logistic Regression for $>$ 2 Classes

We sometimes which to classify a response variable that has more than two classes. There are multi-class extensions to logistic regression ("multinomial regression"), but there are far more popular methods of performing this.