Predictor variables must be categorical (factor) variables in the

standard naive Bayes algorithm. See “Numeric Predictor Variables”

on page 200 for two workarounds for using continuous variables. **B**

When a predictor category is absent in the training data, the algorithm

assigns *zero probability* to the outcome variable in new data,

rather than simply ignoring this variable and using the information

from other variables, as other methods might. Most implementations

of Naive Bayes use a smoothing parameter (Laplace Smoothing)

to prevent this. **O**

Linear discriminant analysis should not be confused with Latent

Dirichlet Allocation, also referred to as LDA. Latent Dirichlet Allocation

is used in text and natural language processing and is unrelated

to linear discriminant analysis. **O**

Recall that the standard deviation is used to normalize a variable to

a *z*-score; the covariance matrix is used in a multivariate extension

of this standardization process. This is known as Mahalanobis distance

(see “Other Distance Metrics” on page 242) and is related to

the LDA function. **B**

Using Discriminant Analysis for Feature Selection

If the predictor variables are normalized prior to running LDA, the

discriminator weights are measures of variable importance, thus

providing a computationally efficient method of feature selection. **B**

Extensions of Discriminant Analysis

More predictor variables: while the text and example in this section

used just two predictor variables, LDA works just as well with more

than two predictor variables. The only limiting factor is the number

of records (estimating the covariance matrix requires a sufficient

number of records per variable, which is typically not an issue

in data science applications).

There are other variants of discriminant analysis. The best known

is quadratic discriminant analysis (QDA). Despite its name, QDA

is still a linear discriminant function. The main difference is that in

LDA, the covariance matrix is assumed to be the same for the two

groups corresponding to *Y* = 0 and *Y* = 1. In QDA, the covariance

matrix is allowed to be different for the two groups. In practice, the

difference in most applications is not critical. **B**

Handling Factor Variables

In logistic regression, factor variables should be coded as in linear

regression; see “Factor Variables in Regression” on page 163. In *R*

and other software, this is normally handled automatically, and

generally reference encoding is used. All of the other classification

methods covered in this chapter typically use the one hot encoder

representation (see “One Hot Encoder” on page 242). In *Python*’s

scikit-learn, it is easiest to use one hot encoding, which means

that only *n – 1* of the resulting dummies can be used in the

regression. **O**

Some of the output from the summary function can effectively be

ignored. The dispersion parameter does not apply to logistic

regression and is there for other types of GLMs. The residual deviance

and the number of scoring iterations are related to the maximum

likelihood fitting method; see “Maximum Likelihood

Estimation” on page 215. **O**

False Positive Rate Confusion

False positive/negative rates are often confused or conflated with

specificity or sensitivity (even in publications and software!).

Sometimes the false positive rate is defined as the proportion of

true negatives that test positive. In many cases (such as network

intrusion detection), the term is used to refer to the proportion of

positive signals that are true negatives. **O**

Uplift

Sometimes the term *uplift* is used to mean the same thing as lift.

An alternate meaning is used in a more restrictive setting, when an

A/B test has been conducted and the treatment (A or B) is then

used as a predictor variable in a predictive model. The uplift is the

improvement in response predicted *for an individual case* with

treatment A versus treatment B. This is determined by scoring the

individual case first with the predictor set to A, and then again with

the predictor toggled to B. Marketers and political campaign consultants

use this method to determine which of two messaging

treatments should be used with which customers or voters.  **O**

Adapting the Loss Function

Many classification and regression algorithms optimize a certain

criteria or *loss function*. For example, logistic regression attempts to

minimize the deviance. In the literature, some propose to modify

the loss function in order to avoid the problems caused by a rare

class. In practice, this is hard to do: classification algorithms can be

complex and difficult to modify. Weighting is an easy way to

change the loss function, discounting errors for records with low

weights in favor of records with higher weights. **B**