Linear Discriminant Analysis

Introduction

Suppose we are given a learning set $\mathcal{L}$ of multivariate observations (i.e., input values in $\mathcal{R}^r$), and suppose each observation is known to have come from one of K predefined classes having similar characteristics.

These classes may be identified, for example, as species of plants, levels of credit worthiness of customers, presence or absence of a specific medical condition, different types of tumors, views on Internet censorship, or whether an e-mail message is spam or non-spam.

To distinguish the known classes from each other, we associate a unique class label (or output value) with each class; the observations are then described as labeled observations.

In each of these situations, there are two main goals:

- **Discrimination:** Use the information in a learning set of labeled observations to construct a classifier (or classification rule) that will separate the predefined classes as much as possible.

- **Classification:** Given a set of measurements on a new unlabeled observation, use the classifier to predict the class of that observation.

A classifier is a combination of the input variables.

In the machine learning literature, discrimination and classification are described as supervised learning techniques; together, they are also referred to as tasks of class prediction.

Whether these goals are at all achievable depends upon the information provided by the input variables.

When there are two classes (i.e., $K = 2$), we need only one classifier, and when there are more than two classes, we need at least two (and at most $K-1$) classifiers to differentiate between the classes and to predict the class of a future observation.

Consider the following medical diagnosis example.

If a patient enters the emergency room with severe stomach pains and symptoms consistent with both food poisoning and appendicitis, a decision has to be made as to which illness is more likely for that patient; only then can the patient be treated.

For this example, the problem is that the appropriate treatment for one cause of illness is the opposite treatment for the other: appendicitis requires surgery, whereas food poisoning does not, and an incorrect diagnosis could lead to a fatal result.

In light of the results from the clinical tests, the physician has to decide upon a course of treatment to maximize the likelihood of success.

If the combination of test results points in a particular direction, surgery is recommended; otherwise, the physician recommends a non-surgical treatment.

A classifier is constructed from past experience based upon the test results of previously treated patients (the learning set).

The more reliable the classifier, the greater the chance for a successful diagnostic outcome for a future patient.

Similarly, a credit card company or a bank uses loan histories of past customers to decide whether a new customer would be a good or bad credit risk.

A post office uses handwriting samples of a large number of individuals to design an automated method for distinguishing between different handwritten digits and letters.

Molecular biologists use gene expression data to distinguish between known classes of tumors.

Political scientists use frequencies of word usage to identify the authorship of different political tracts.

A person who uses e-mail would certainly like to have a filter that recognizes whether a message is spam or not.

In this chapter, we focus upon the most basic type of classifier: a linear combination of the input variables.

This problem has been of interest to statisticians since R. A. Fisher introduced the **linear discriminant function** (Fisher, 1936).

Example: Wisconsin Diagnostic Breast Cancer Data

See the textbook.

Classes and Features

We assume that the population $\mathcal{P}$ is partitioned into K unordered classes, groups, or subpopulations, which we denote by $\Pi_1, \Pi_2, \ldots, \Pi_K$.

Each item in $\mathcal{P}$ is classified into one (and only one) of those classes.

Measurements on a sample of items are to be used to help assign future unclassified items to one of the designated classes.

The random r-vector $\mathbf{X}$, given by

$$\mathbf{X} = (X_1, \ldots, X_r)^T,$$

represents the r measurements on an item (i.e., $\mathbf{X} \in \mathcal{R}^r$).

The variables $X_1, \ldots, X_r$ are likely to be chosen because of their suspected ability to distinguish between the K classes.

The variables are called **discriminating** or **feature variables**, and the vector $\mathbf{X}$ is the **feature vector**.

It may sometimes be appropriate to include in an analysis the additional classes of Π_D and Π_O to signify that decisions could not be made due to either an element of doubt in the assignment or indications that certain items constitute outliers and could not possibly belong to any of the designated classes.

Why Not Linear Regression?

Suppose that we are trying to predict the medical condition of a patient in the emergency room on the basis of her symptoms.

In this simplified example, there are three possible diagnoses: stroke, drug overdose, and epileptic seizure.

We could consider encoding these values as a quantitative response variable, Y, as follows:

$$Y = \begin{cases} 1, & \text{if stroke}, \\ 2, & \text{if drug overdose}, \\ 3, & \text{if epileptic seizure}. \end{cases}$$

Using this coding, least squares could be used to fit a linear regression model to predict Y on the basis of a set of predictors $X_1, \ldots, X_p$.

Unfortunately, this coding implies an ordering on the outcomes, putting drug overdose in between stroke and epileptic seizure, and insisting that the difference between stroke and drug overdose is the same as the difference between drug overdose and epileptic seizure.

In practice there is no particular reason that this needs to be the case.

For instance, one could choose an equally reasonable coding,

$$Y = \begin{cases} 1, & \text{if epileptic seizure,} \\ 2, & \text{if stroke,} \\ 3, & \text{if drug overdose,} \end{cases}$$

which would imply a totally different relationship among the three conditions.

Each of these codings would produce fundamentally different linear models that would ultimately lead to different sets of predictions on test observations.

If the response variable's values did take on a natural ordering, such as mild, moderate, and severe, and we felt the gap between mild and moderate was similar to the gap between moderate and severe, then a 1, 2, 3 coding would be reasonable.

Unfortunately, in general there is no natural way to convert a qualitative response variable with more than two levels into a quantitative response that is ready for linear regression.

For a binary (two level) qualitative response, the situation is better.

For instance, perhaps there are only two possibilities for the patient's medical condition: stroke and drug overdose.

We could then potentially use the dummy variable approach to code the response as follows:

$$Y = \begin{cases} 0, & \text{if stroke,} \\ 1, & \text{if drug overdose.} \end{cases}$$

We could then fit a linear regression to this binary response, and predict drug overdose if $\hat{Y} > 0.5$ and stroke otherwise.

In the binary case it is not hard to show that even if we flip the above coding, linear regression will produce the same final predictions.

For a binary response with a 0/1 coding as above, regression by least squares does make sense.

It can be shown that the $\mathbf{X}\hat{\beta}$ obtained using linear regression is in fact an estimate of Pr(drug overdose|$\mathbf{X}$) in this special case.

However, if we use linear regression, some of our estimates might be outside the $[0, 1]$ interval, making them hard to interpret as probabilities.

Nevertheless, the predictions provide an ordering and can be interpreted as crude probability estimates.

Curiously, it turns out that the classifications that we get if we use linear regression to predict a binary response will be the same as for the linear discriminant analysis (LDA) procedure.

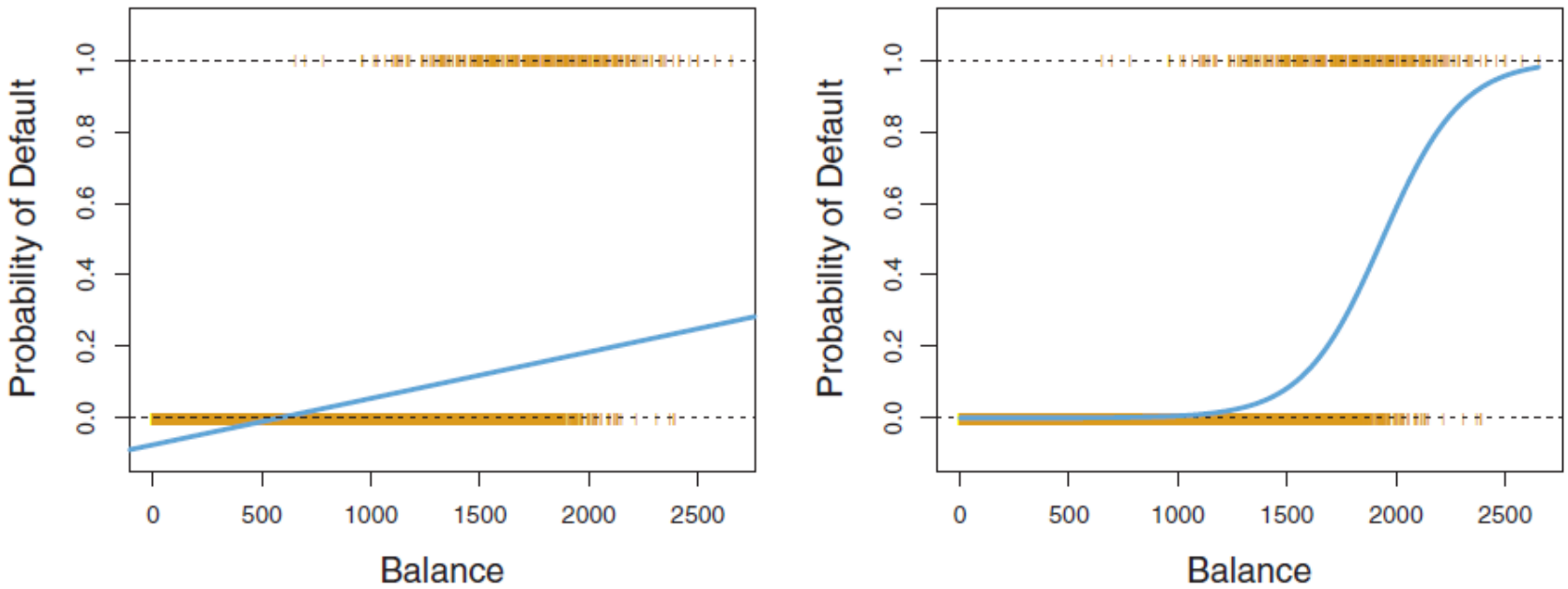

FIGURE: Left: Estimated probability using linear regression. Some estimated probabilities are negative! The orange ticks indicate the 0/1 values coded for Y. Right: Predicted probabilities of default using logistic regression. All probabilities lie between 0 and 1.

However, the dummy variable approach cannot be easily extended to accommodate qualitative responses with more than two levels.

For these reasons, it is preferable to use a classification method that is truly suited for qualitative response values.

Binary Classification

Consider, first, the **binary** classification problem ($K = 2$) where we wish to discriminate between two classes Π_1 and Π_2, such as the malignant and benign tumors in the breast cancer example.

Bayes's Rule Classifier

Let

$$P(\mathbf{X} \in \Pi_i) = \pi_i, \quad i = 1, 2,$$

be the **prior probabilities** that a randomly selected observation $\mathbf{X} = \mathbf{x}$ belongs to either Π_1 or Π_2.

Suppose also that the conditional multivariate probability density of $\mathbf{X}$ for the ith class is

$$P(\mathbf{X} = \mathbf{x}|\mathbf{X} \in \Pi_i) = f_i(\mathbf{x}), \quad i = 1, 2.$$

We note that there is no requirement that the $\{f_i(\cdot)\}$ be continuous; they could be discrete or be finite mixture distributions or even have singular covariance matrices.

From Bayes's theorem yields the **posterior probability**,

$$P(\Pi_i|\mathbf{x}) = P(\mathbf{X} \in \Pi_i|\mathbf{X} = \mathbf{x}) = \frac{\pi_i f_i(\mathbf{x})}{\pi_1 f_1(\mathbf{x}) + \pi_2 f_2(\mathbf{x})},$$

that the observed $\mathbf{x}$ belongs to Π_i, $i = 1, 2$.

For a given $\mathbf{x}$, a reasonable classification strategy is to assign $\mathbf{x}$ to that class with the higher posterior probability.

This strategy is called the **Bayes's rule classifier**.

In other words, we assign $\mathbf{x}$ to Π_1 if

$$\frac{P(\Pi_1|\mathbf{x})}{P(\Pi_2|\mathbf{x})} > 1,$$

and we assign $\mathbf{x}$ to Π_2 otherwise.

The ratio $P(\Pi_1|\mathbf{x})/P(\Pi_2|\mathbf{x})$ is referred to as the **odds-ratio** that Π_1 rather than Π_2 is the correct class given the information in $\mathbf{x}$.

The Bayes's rule classifier is equivalent to that assign $\mathbf{x}$ to Π_1 if

$$\frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} > \frac{\pi_2}{\pi_1},$$

and to Π_2 otherwise.

On the boundary $\{\mathbf{x} \in \mathcal{R}^r | f_1(\mathbf{x})/f_2(\mathbf{x}) = \pi_2/\pi_1\}$, we randomize (e.g., by tossing a fair coin) between assigning $\mathbf{x}$ to either Π_1 or Π_2.

Gaussian Linear Discriminant Analysis

We now make the Bayes's rule classifier more specific by following Fisher's assumption that both multivariate probability densities are multivariate Gaussian having arbitrary mean vectors and a common covariance matrix.

That is, we take $f_1(\cdot)$ to be a $N_r[\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1]$ density and $f_2(\cdot)$ be a $N_r[\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2]$ density, and we make the homogeneity assumption that $\boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_2 = \boldsymbol{\Sigma}_{XX}$.

The ratio of the two densities is given by

$$\frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} = \frac{\exp\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_1)^T\boldsymbol{\Sigma}_{XX}^{-1}(\mathbf{x}-\boldsymbol{\mu}_1)\}}{\exp\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_2)^T\boldsymbol{\Sigma}_{XX}^{-1}(\mathbf{x}-\boldsymbol{\mu}_2)\}}.$$

Taking logarithms (a monotonically increasing function), we have

$$\begin{aligned}\ln\frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} &= (\boldsymbol{\mu}_1-\boldsymbol{\mu}_2)^T\boldsymbol{\Sigma}_{XX}^{-1}\mathbf{x} - \frac{1}{2}(\boldsymbol{\mu}_1-\boldsymbol{\mu}_2)^T\boldsymbol{\Sigma}_{XX}^{-1}(\boldsymbol{\mu}_1+\boldsymbol{\mu}_2)\\ &= (\boldsymbol{\mu}_1-\boldsymbol{\mu}_2)^T\boldsymbol{\Sigma}_{XX}^{-1}(\mathbf{x}-\bar{\boldsymbol{\mu}}),\end{aligned}$$

- $\bar{\boldsymbol{\mu}} = (\boldsymbol{\mu}_1+\boldsymbol{\mu}_2)/2.$

Since

$$(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T \boldsymbol{\Sigma}_{XX}^{-1} (\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2) = \boldsymbol{\mu}_1^T \boldsymbol{\Sigma}_{XX}^{-1} \boldsymbol{\mu}_1 - \boldsymbol{\mu}_2^T \boldsymbol{\Sigma}_{XX}^{-1} \boldsymbol{\mu}_2,$$

we have

$$L(\mathbf{x}) = \ln \left\{ \frac{f_1(\mathbf{x})\pi_1}{f_2(\mathbf{x})\pi_2} \right\} = b_0 + \mathbf{b}^T \mathbf{x},$$

$$b_0 = -\frac{1}{2} \left(\boldsymbol{\mu}_1^T \boldsymbol{\Sigma}_{XX}^{-1} \boldsymbol{\mu}_1 - \boldsymbol{\mu}_2^T \boldsymbol{\Sigma}_{XX}^{-1} \boldsymbol{\mu}_2 \right) + \ln(\pi_2/\pi_1),$$

$$\mathbf{b} = \boldsymbol{\Sigma}_{XX}^{-1} (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2).$$

It can be seen that $L(\mathbf{x})$ is a linear function of $\mathbf{x}$.

Thus, we assign $\mathbf{x}$ to Π_1 if the logarithm of the ratio of the two posterior probabilities is greater than zero; that is, $L(\mathbf{x}) > 0$.

Otherwise, we assign $\mathbf{x}$ to Π_2.

Note that on the boundary $\{\mathbf{x} \in \mathcal{R}^r | L(\mathbf{x}) = 0\}$, the resulting equation is linear in $\mathbf{x}$ and, therefore, defines a hyperplane that divides the two classes.

The rule is generally referred to as **Gaussian linear discriminant analysis (LDA)**.

The part of the function $L(\mathbf{x})$ that depends upon $\mathbf{x}$,

$$U = \mathbf{b}^T\mathbf{x} = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T\boldsymbol{\Sigma}_{XX}^{-1}\mathbf{x},$$

is known as **Fisher's linear discriminant function (LDF)**.

Fisher actually derived the LDF using a nonparametric argument that involved no distributional assumptions.

He looked for that linear combination, $\mathbf{a}^T\mathbf{X}$, of the feature vector $\mathbf{X}$ that separated the two classes as much as possible.

In particular, he showed that $\mathbf{a} \propto \boldsymbol{\Sigma}_{XX}^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)$ maximized the squared difference of the two class means of $\mathbf{a}^T\mathbf{X}$ relative to the within-class variation of that difference.

Total Misclassification Probability

The LDF partitions the feature space $\mathcal{R}^r$ into disjoint classification regions R_1 and R_2.

If $\mathbf{x}$ falls into region R_1, it is classified as belonging to Π_1, whereas if $\mathbf{x}$ falls into region R_2, it is classified into Π_2.

We now calculate the probability of misclassifying $\mathbf{x}$.

Misclassification occurs either if $\mathbf{x}$ is assigned to Π_2, but actually belongs to Π_1, or vice versa.

Define

$$\Delta^2 = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T \boldsymbol{\Sigma}_{XX}^{-1} (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)$$

to be the **squared Mahalanobis distance** between Π_1 and Π_2.

Then, for $i = 1, 2$,

$$\mathrm{E}(U|\mathbf{X} \in \Pi_i) = \mathbf{b}^T \boldsymbol{\mu}_i = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T \boldsymbol{\Sigma}_{XX}^{-1} \boldsymbol{\mu}_i,$$

$$\mathrm{Var}(U|\mathbf{X} \in \Pi_i) = \mathbf{b}^T \boldsymbol{\Sigma}_{XX} \mathbf{b} = \Delta^2.$$

The total misclassification probability is, therefore,

$$P(\Delta) = P(\mathbf{X} \in R_2 | \mathbf{X} \in \Pi_1)\pi_1 + P(\mathbf{X} \in R_1 | \mathbf{X} \in \Pi_2)\pi_2,$$

$$P(\mathbf{X} \in R_2 | \mathbf{X} \in \Pi_1) = P(L(\mathbf{X}) < 0 | \mathbf{X} \in \Pi_1) = P\left(Z < -\frac{\Delta}{2} - \frac{1}{\Delta} \ln \frac{\pi_2}{\pi_1}\right)$$

$$= \Phi\left(-\frac{\Delta}{2} - \frac{1}{\Delta} \ln \frac{\pi_2}{\pi_1}\right),$$

$$P(\mathbf{X} \in R_1 | \mathbf{X} \in \Pi_2) = \Phi\left(-\frac{\Delta}{2} + \frac{1}{\Delta} \ln \frac{\pi_2}{\pi_1}\right).$$

If $\pi_1 = \pi_2 = 1/2$, then

$$P(\mathbf{X} \in R_2 | \mathbf{X} \in \Pi_1) = P(\mathbf{X} \in R_1 | \mathbf{X} \in \Pi_2) = \Phi\left(-\frac{\Delta}{2}\right),$$

and, hence,

$$P(\Delta) = \Phi\left(-\frac{\Delta}{2}\right).$$

A graph of $P(\Delta)$ against Δ shows a downward-sloping curve, as one would expect; it has the value 0.5 when $\Delta = 0$ (i.e., the two populations are identical) and tends to zero as Δ increases.

In other words, the greater the distance between the two population means, the less likely one is to misclassify $\mathbf{x}$.

Sampling Scenarios

Usually, the $2r + r(r+1)/2$ distinct parameters in $\boldsymbol{\mu}_1$, $\boldsymbol{\mu}_2$, and $\boldsymbol{\Sigma}_{XX}$ will be unknown, but can be estimated from learning data on $\mathbf{X}$.

Assume, then, that we have available independent learning samples from the two classes Π_1 and Π_2.

Let $\{\mathbf{X}_{1j}\}$ be a learning sample of size n_1 taken from Π_1 and let $\{\mathbf{X}_{2j}\}$ be a learning sample of size n_2 taken from Π_2.

The following different scenarios are possible when sampling from population $\mathcal{P}$:

1. Conditional sampling, where a sample of fixed size $n = n_1 + n_2$ is randomly selected from $\mathcal{P}$, and at a fixed $\mathbf{x}$ there are $n_i(\mathbf{x})$ observations from Π_i, $i = 1, 2$. This sampling scenario often appears in bioassays.

2. Mixture sampling, where a sample of fixed size $n = n_1 + n_2$ is randomly selected from $\mathcal{P}$ so that n_1 and n_2 are randomly selected. This is quite common in discrimination studies.

3. Separate sampling, where a sample of fixed size n_i is randomly selected from Π_i, $i = 1, 2$, and $n = n_1 + n_2$. Overall, this is the most popular scenario.

Sample Estimates

The ML estimates of $\boldsymbol{\mu}_i$, $i = 1, 2$, and $\boldsymbol{\Sigma}_{XX}$ are given by

$$\widehat{\boldsymbol{\mu}}_i = \bar{\mathbf{x}}_i = \frac{1}{n_i} \sum_{j=1}^{n_i} \mathbf{x}_{ij}, \quad i = 1, 2,$$

$$\widehat{\boldsymbol{\Sigma}}_{XX} = \frac{1}{n} \mathbf{S}_{XX}, \quad \mathbf{S}_{XX} = \mathbf{S}_{XX}^{(1)} + \mathbf{S}_{XX}^{(2)},$$

$$\mathbf{S}_{XX}^{(i)} = \sum_{j=1}^{n_i} (\mathbf{x}_{ij} - \bar{\mathbf{x}}_i)(\mathbf{x}_{ij} - \bar{\mathbf{x}}_i)^T, \quad i = 1, 2,$$

$$n = n_1 + n_2.$$

Note

If we wish to compute an unbiased estimator of $\mathbf{\Sigma}_{XX}$, we can divide $\mathbf{S}_{XX}$ its degrees of freedom $n - 2 = n_1 + n_2 - 2$ (rather than by n) to make $\widehat{\mathbf{\Sigma}}_{XX}$.

The prior probabilities, π_1 and π_2, may be known or can be closely approximated in certain situations from past experience.

If π_1 and π_2 are unknown, they can be estimated by

$$\hat{\pi}_i = \frac{n_i}{n}, \quad i = 1, 2.$$

Substituting these estimates into $L(\mathbf{x})$ in yields the ML estimates of b_0 and $\mathbf{b}_1$:

$$\widehat{L}(\mathbf{x}) = \hat{b}_0 + \widehat{\mathbf{b}}_1 \mathbf{x},$$

$$\hat{b}_0 = \widehat{\Sigma}_{XX}^{-1}(\bar{\mathbf{X}}_1 - \bar{\mathbf{X}}_2),$$

$$\widehat{\mathbf{b}}_1 = \frac{1}{2}(\bar{\mathbf{X}}_1^T \widehat{\Sigma}_{XX}^{-1} \bar{\mathbf{X}} - \bar{\mathbf{X}}_2^T \widehat{\Sigma}_{XX}^{-1} \bar{\mathbf{X}}_2) + \ln\left(\frac{n_1}{n_2}\right) - \ln\left(\frac{n_2}{n}\right).$$

The classification rule assigns $\mathbf{x}$ to

Π_1: if $L(\mathbf{x}) > 0$,

Π_2: otherwise.

The second term of $\widehat{L}(\mathbf{x})$,

$$\widehat{\mathbf{b}}^T\mathbf{x} = (\bar{\mathbf{X}}_1 - \bar{\mathbf{X}}_2)^T\widehat{\mathbf{\Sigma}}_{XX}^{-1}\mathbf{x},$$

estimates Fisher's LDF.

For large samples ($n_i \to \infty,\ i = 1, 2$), the distribution of $\widehat{\mathbf{b}}$ in is Gaussian.

This result allows us to study the separation of two given training samples, as well as the assumptions of normality and covariance matrix homogeneity, by drawing a histogram or normal probability plot of the LDF evaluated for every observation in the training samples.

Nonparametric density estimates of the LDF scores for each class are especially useful in this regard.

Example: Wisconsin Breast Cancer Data

See the textbook.

LDA via Multiple Regression

See the textbook.

Variable Selection

High-dimensional data often contain pairs of highly correlated variables, which introduce collinearity into discrimination and classification problems. So, variable selection becomes a priority.

The connection between Fisher's LDF and multiple regression provides us with a vehicle for selecting important discriminating variables. Thus, the variable selection techniques of FS and BE stepwise procedures, C_p, LARS, and Lasso can all be used in the discrimination context as well as in regression.

Logistic Discrimination

See the textbook.

Gaussian LDA or Logistic Discrimination?

Theoretical and empirical comparisons have been carried out between Gaussian LDA and logistic discriminant analysis. Some of the differences are the following:

1. The conditional log-likelihood (8.54) is valid under general exponential family assumptions on $f(\cdot)$ (which includes the multivariate Gaussian model with common covariance matrix). This suggests that logistic discrimination is more robust to nonnormality than Gaussian LDA.

2. Simulation studies have shown that when the Gaussian distributional assumptions or the common covariance matrix assumption are not satisfied, logistic discrimination performs much better.

3. Sensitivity to gross outliers can be a problem for Gaussian LDA, whereas outliers are reduced in importance in logistic discrimination, which essentially fits a sigmoidal function (rather than a linear function).

4. Logistic discriminant analysis is asymptotically less efficient than is Gaussian LDA because the latter is based upon full ML rather than conditional ML.

5. At the point when we would expect good discrimination to take place, logistic discrimination requires a much larger sample size than does Gaussian LDA to attain the same (asymptotic) error rate distribution (Efron, 1975), and this result extends to LDA using an exponential family with plug-in estimates.

Quadratic Discriminant Analysis

How is the classification rule in LDA affected if the covariance matrices of the two Gaussian populations are not equal to each other?

That is, if $\boldsymbol{\Sigma}_1 \neq \boldsymbol{\Sigma}_2$.

In this case,

$$\ln \frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} = c_0 - \frac{1}{2}\left\{(\mathbf{x}-\boldsymbol{\mu}_1)^T\boldsymbol{\Sigma}_1^{-1}(\mathbf{x}-\boldsymbol{\mu}_1) - (\mathbf{x}-\boldsymbol{\mu}_2)^T\boldsymbol{\Sigma}_2^{-1}(\mathbf{x}-\boldsymbol{\mu}_2)\right\}$$

$$= c_1 - \frac{1}{2}\mathbf{x}^T(\boldsymbol{\Sigma}_1^{-1} - \boldsymbol{\Sigma}_2^{-1})\mathbf{x} + (\boldsymbol{\mu}_1^T\boldsymbol{\Sigma}_1^{-1} - \boldsymbol{\mu}_2\boldsymbol{\Sigma}_2^{-1})\mathbf{x}.$$

The log-likelihood ratio has the form of a quadratic function of $\mathbf{x}$. In this case, set

$$Q(\mathbf{x}) = \beta_0 + \boldsymbol{\beta}^T\mathbf{x} + \mathbf{x}^T\boldsymbol{\Omega}\mathbf{x},$$

$$\boldsymbol{\Omega} = -\frac{1}{2}(\boldsymbol{\Sigma}_1^{-1} - \boldsymbol{\Sigma}_2^{-1}),$$

$$\boldsymbol{\beta} = \boldsymbol{\Sigma}_1^{-1}\boldsymbol{\mu}_1 - \boldsymbol{\Sigma}_2^{-1}\boldsymbol{\mu}_2,$$

$$\beta_0 = -\frac{1}{2}\left\{\ln\frac{|\boldsymbol{\Sigma}_1|}{|\boldsymbol{\Sigma}_2|} + \boldsymbol{\mu}_1^T\boldsymbol{\Sigma}_1^{-1}\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2^T\boldsymbol{\Sigma}_2^{-1}\boldsymbol{\mu}_2\right\} - \ln(\pi_2/\pi_1).$$

The classification rule is that we assign $\mathbf{x}$ to Π_1 if $Q(\mathbf{x}) > 0$ and to Π_2 otherwise.

The function $Q(\mathbf{x})$ of $\mathbf{x}$ is called a **quadratic discriminant function (QDF)** and the classification rule is referred to as **quadratic discriminant analysis (QDA)**.

The boundary $\{\mathbf{x} \in \mathcal{R}^r | Q(\mathbf{x}) = 0\}$ that divides the two classes is a quadratic function of $\mathbf{x}$.

An approximation to the boundaries obtained by QDA can be obtained using an LDA approach that enlists the aid of the linear terms, squared terms, and all pairwise products of the feature variables.

For example, if we have two feature variables X_1 and X_2, then quadratic LDA would use X_1, X_2, X_1^2, X_2^2, and $X_1 X_2$ in the linear discriminant function with $r = 5$.

Types of Errors

In practice, a binary classifier such as this one can make two types of errors:

- it can incorrectly assign an individual who defaults to the no default category, or
- it can incorrectly assign an individual who does not default to the default category.

It is often of interest to determine which of these two types of errors are being made.

Confusion Matrix

Table: A confusion matrix compares the LDA predictions to the true default statuses for the 10, 000 training observations in the Default data set.

		True default status		
		No	Yes	Total
Predicted default status	No	9,644	252	9,896
	Yes	23	81	104
	Total	9,667	333	10,000

A **confusion matrix**, shown for the Default data in the table, is a convenient way to display this information.

The table reveals that LDA predicted that a total of 104 people would default.

Of these people, 81 actually defaulted and 23 did not.

Hence only 23 out of 9,667 of the individuals who did not default were incorrectly labeled.

This looks like a pretty low error rate!

However, of the 333 individuals who defaulted, 252 (or 75.7%) were missed by LDA.

So while the overall error rate is low, the error rate among individuals who defaulted is very high.

From the perspective of a credit card company that is trying to identify high-risk individuals, an error rate of $252/333 = 75.7\%$ among individuals who default may well be unacceptable.

Sensitivity and Specificity

Sensitivity and specificity are statistical measures of the performance of a binary classification test, also known in statistics as classification function.

Sensitivity (also called the **true positive rate**, the **recall**, or **probability of detection** in some fields) measures the proportion of positives that are correctly identified as such (i.e. the percentage of sick people who are correctly identified as having the condition).

Specificity (also called the **true negative rate**) measures the proportion of negatives that are correctly identified as such (i.e., the percentage of healthy people who are correctly identified as not having the condition).

		Predicted		
		Positive	Negative	Total
True	Positive	True Positive (TP)	False Negative (FN)	P
	Negative	False Positive (FP)	True Negative (TN)	N
Class	Total	N^*	P^*	

- Sensitivity (TPR): TP/P.
- Specificity (TNR): TN/N.
- False positive rate (FPR): FP/N.

In the above example, the sensitivity is the percentage of true defaulters that are identified, a low 24.3% in this case.

The specificity is the percentage of non-defaulters that are correctly identified, here $(123/9,667) \times 100\% = 99.8\%$.

Why does LDA do such a poor job of classifying the customers who default?

In other words, why does it have such a low sensitivity?

As we have seen, LDA is trying to approximate the Bayes classifier, which has the lowest total error rate out of all classifiers (if the Gaussian model is correct).

That is, the Bayes classifier will yield the smallest possible total number of misclassified observations, irrespective of which class the errors come from.

That is, some misclassifications will result from incorrectly assigning a customer who does not default to the default class, and others will result from incorrectly assigning a customer who defaults to the non-default class.

In contrast, a credit card company might particularly wish to avoid incorrectly classifying an individual who will default, whereas incorrectly classifying an individual who will not default, though still to be avoided, is less problematic.

We will now see that it is possible to modify LDA in order to develop a classifier that better meets the credit card company's needs.

Threshold

The Bayes classifier works by assigning an observation to the class for which the posterior probability $P(\Pi_k|\mathbf{x})$ is greatest.

In the two-class case, this amounts to assigning an observation to the default class if

$$\Pr(\text{default=Yes}|\mathbf{x}) > 0.5.$$

Thus, the Bayes classifier, and by extension LDA, uses a threshold of 50% for the posterior probability of default in order to assign an observation to the default class.

However, if we are concerned about incorrectly predicting the default status for individuals who default, then we can consider lowering this threshold.

For instance, we might label any customer with a posterior probability of default above 20% to the default class.

In other words, instead of assigning an observation to the default class if

$$\Pr(\text{default=Yes}|\mathbf{x}) > 0.5$$

holds, we could instead assign an observation to this class if

$$\Pr(\text{default=Yes}|\mathbf{x}) > 0.2.$$

TABLE: A confusion matrix compares the LDA predictions to the true default statuses for the 10,000 training observations in the Default data set, using a modified threshold value that predicts default for any individuals whose posterior default probability exceeds 20%.

		True default status		
		No	Yes	Total
Predicted default status	No	9,432	138	9,570
	Yes	235	195	430
	Total	9,667	333	10,000

Now LDA predicts that 430 individuals will default. Of the 333 individuals who default, LDA correctly predicts all but 138, or 41.4%.

This is a vast improvement over the error rate of 75.7% that resulted from using the threshold of 50%.

However, this improvement comes at a cost: now 235 individuals who do not default are incorrectly classified.

As a result, the overall error rate has increased slightly to 3.73%.

But a credit card company may consider this slight increase in the total error rate to be a small price to pay for more accurate identification of individuals who do indeed default.

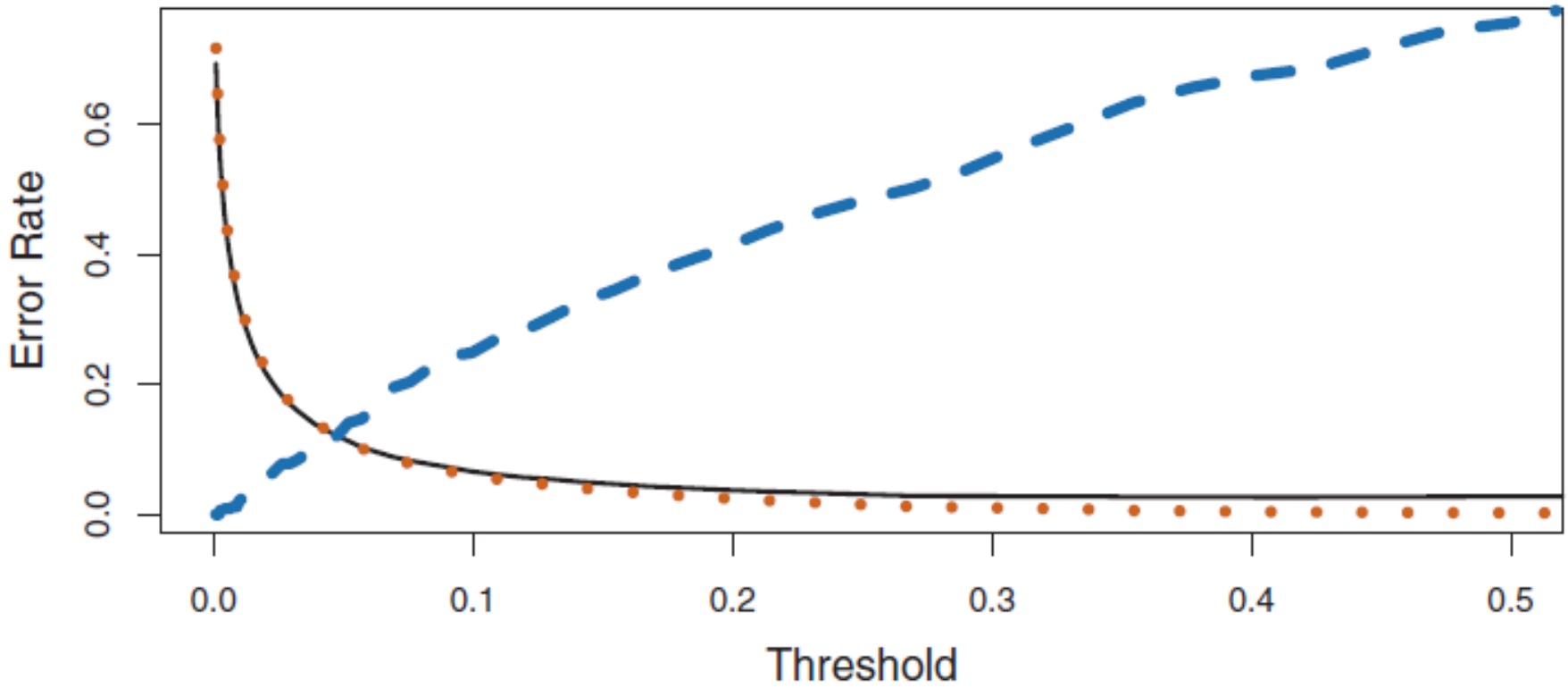

FIGURE: For the Default data set, error rates are shown as a function of the threshold value for the posterior probability that is used to perform the assignment. The black solid line displays the overall error rate. The blue dashed line represents the fraction of defaulting customers that are incorrectly classified, and the orange dotted line indicates the fraction of errors among the non-defaulting customers.

Various error rates are shown as a function of the threshold value.

Using a threshold of 0.5 minimizes the overall error rate, shown as a black solid line.

This is to be expected, since the Bayes classifier uses a threshold of 0.5 and is known to have the lowest overall error rate.

But when a threshold of 0.5 is used, the error rate among the individuals who default is quite high (blue dashed line).

As the threshold is reduced, the error rate among individuals who default decreases steadily, but the error rate among the individuals who do not default increases.

How can we decide which threshold value is best?

Such a decision must be based on domain knowledge, such as detailed information about the costs associated with default.

ROC Curve

In statistics, a **receiver operating characteristic curve**, or **ROC curve**, is a graphical plot that illustrates the performance of a binary classifier system as its discrimination threshold is varied.

The curve is created by plotting the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings.

ROC analysis provides tools to select possibly optimal models and to discard suboptimal ones independently from (and prior to specifying) the cost context or the class distribution.

ROC analysis is related in a direct and natural way to cost/benefit analysis of diagnostic decision making.

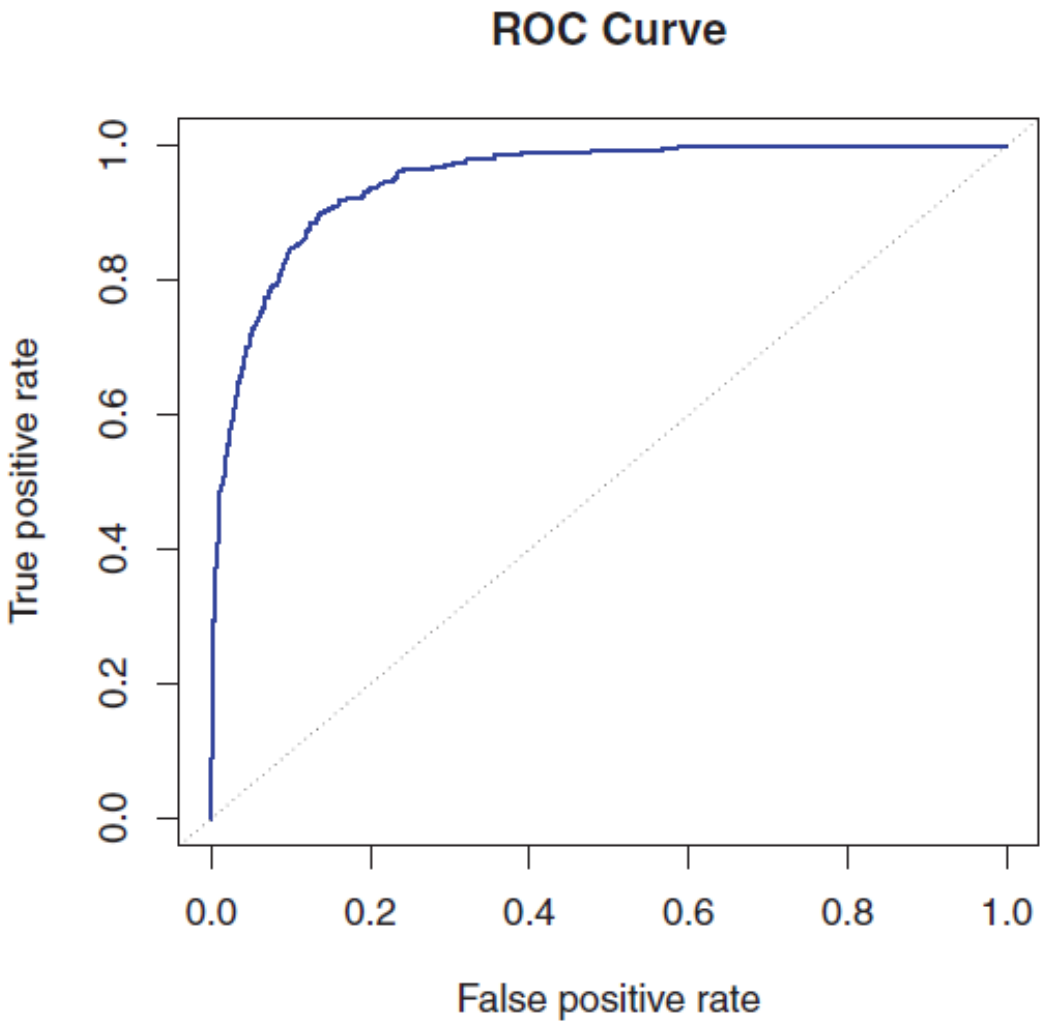

FIGURE: A ROC curve for the LDA classifier on the Default data. It traces out two types of error as we vary the threshold value for the posterior probability of default. The actual thresholds are not shown.

The overall performance of a classifier, summarized over all possible thresholds, is given by the **area under the (ROC) curve (AUC)**.

The best possible prediction method would yield a point in the upper left corner or coordinate $(0, 1)$ of the ROC space, representing 100% sensitivity (no false negatives) and 100% specificity (no false positives).

The $(0, 1)$ point is also called a **perfect classification**.

For the Default data the AUC is 0.95, which is close to the maximum of one so would be considered very good.

A random guess would give a point along a diagonal line (the so-called **line of no-discrimination**) from the left bottom to the top right corners (regardless of the positive and negative base rates).

An intuitive example of random guessing is a decision by flipping coins.

As the size of the sample increases, a random classifier's ROC point migrates towards the diagonal line.

In the case of a balanced coin, it will migrate to the point $(0.5, 0.5)$.

We expect a classifier that performs no better than chance to have an AUC of 0.5 (when evaluated on an independent test set not used in model training).

ROC curves are useful for comparing different classifiers, since they take into account all possible thresholds.

Multiclass LDA

Assume now that the population of interest is divided into $K > 2$ nonoverlapping (disjoint) classes.

For example, in a database made publicly available by the U.S. Postal Service, each item is a (16×16) pixel image of a digit extracted from a real-life zip code that is handwritten onto an envelope.

The database consists of thousands of these handwritten digits, each of which is viewed as a point in an input space of 256 dimensions.

The classification problem is to assign each digit to one of the 10 classes $0, 1, 2, \ldots, 9$.

We could carry out $\binom{K}{2}$ different two-class linear discriminant analyses, where we set up a sequence of one class versus the rest classification scenarios.

Such a solution does not work because it would produce regions that do not belong to any of the K classes considered.

Instead, the two-class methodology carries over in a straightforward way to the multiclass situation.

Specifically, we wish to partition the sample space into K non-overlapping regions $R_1, R_2, \ldots, R_K$, such that an observation $\mathbf{x}$ is assigned to class Π_i if $\mathbf{x} \in R_i$.

The partition is to be determined so that the total misclassification rate is a minimum.

Bayes's Rule Classifier

Let $P(\mathbf{X} \in \Pi_i) = \pi_i$, $i = 1, 2, \ldots, K$, be the prior probabilities of a randomly selected observation $\mathbf{X}$ belonging to each of the different classes in the population, and let

$$P(\mathbf{X} = \mathbf{x} | \mathbf{X} \in \Pi_i) = f_i(\mathbf{x}), \qquad i = 1, 2, \ldots, K,$$

be the multivariate probability density for each class.

The resulting posterior probability that an observed $\mathbf{x}$ belongs to the ith class is given by

$$P(\Pi_i|\mathbf{x}) = P(\mathbf{X} \in \Pi|\mathbf{X} = \mathbf{x}) = \frac{\pi_i f_i(\mathbf{x})}{\sum_{k=1}^{K} \pi_k f_k(\mathbf{x})}, \quad i = 1, 2, \ldots, K.$$

The Bayes's rule classifier for K classes assigns $\mathbf{x}$ to that class with the highest posterior probability.

We assign $\mathbf{x}$ to Π_i if

$$\pi_i f_i(\mathbf{x}) = \max_{1 \leq j \leq K} \pi_j f_j(\mathbf{x}).$$

If the maximum does not uniquely define a class assignment for a given $\mathbf{x}$, then use a random assignment to break the tie between the appropriate classes.

Thus, $\mathbf{x}$ gets assigned to Π_i if $\pi_i f_i(\mathbf{x}) > \pi_j f_j(\mathbf{x})$ for all $j \neq i$, or equivalently, if $\ln\left(\pi_i f_i(\mathbf{x})\right) > \ln\left(\pi_j f_j(\mathbf{x})\right)$ for all $j \neq i$.

The Bayes's rule classifier can be defined in an equivalent form by pairwise comparisons of posterior probabilities.

We define the log-odds that $\mathbf{x}$ is assigned to Π_i rather than to Π_j as follows:

$$L_{ij}(\mathbf{x}) = \ln\left\{\frac{p(\Pi_i|\mathbf{x})}{p(\Pi_j|\mathbf{x})}\right\} = \ln\left\{\frac{\pi_i f_i(\mathbf{x})}{\pi_j f_j(\mathbf{x})}\right\}.$$

Then, we assign $\mathbf{x}$ to Π_i if $L_{ij}(\mathbf{x}) > 0$ for all $j \neq i$.

We define classification regions, $R_1, R_2, \ldots, R_K$, as those areas of $\mathcal{R}^r$ such that

$$R_i = \{\mathbf{x} \in \mathcal{R}^r | L_{ij}(\mathbf{x}) > 0,\ j = 1, 2, \ldots, K,\ j \neq i\}, \quad i = 1, 2, \ldots, K.$$

This argument can be made more specific by assuming for the ith class Π_i that $f_i(\cdot)$ is the $N_r[\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i]$ density.

We further assume that the covariance matrices for the K classes are identical, $\boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_2 = \cdots = \boldsymbol{\Sigma}_K = \boldsymbol{\Sigma}_{XX}$.

Under these multivariate Gaussian assumptions, the log-odds of assigning $\mathbf{x}$ to Π_i (as opposed to Π_j) is a linear function of $\mathbf{x}$,

$$L_{ij}(\mathbf{x}) = b_{0ij} + \mathbf{b}_{ij}^T \mathbf{x},$$

$$\mathbf{b}_{ij} = (\mu_i - \mu_j)^T \mathbf{\Sigma}_{XX}^{-1},$$

$$b_{0ij} = -\frac{1}{2}\left\{ \mu_i^T \mathbf{\Sigma}_{XX}^{-1} \boldsymbol{\mu}_i - \boldsymbol{\mu}_j^T \mathbf{\Sigma}_{XX} \boldsymbol{\mu}_j \right\} + \ln(\pi_i/\pi_j).$$

Because $L_{ij}(\mathbf{x})$ is linear in $\mathbf{x}$, the regions $\{R_i\}$ partition r-dimensional space by means of hyperplanes.

Maximum-Likelihood Estimates

See the textbook.

Multiclass Logistic Discrimination

See the textbook.

LDA via Reduced-Rank Regression

See the textbook.

Example: Gilgaied Soil

See the textbook.

Separating Hyperplanes

We have seen that linear discriminant analysis and logistic regression both estimate linear decision boundaries in similar but slightly different ways.

Next we describe separating hyperplane classifiers.

These procedures construct linear decision boundaries that explicitly try to separate the data into different classes as well as possible.

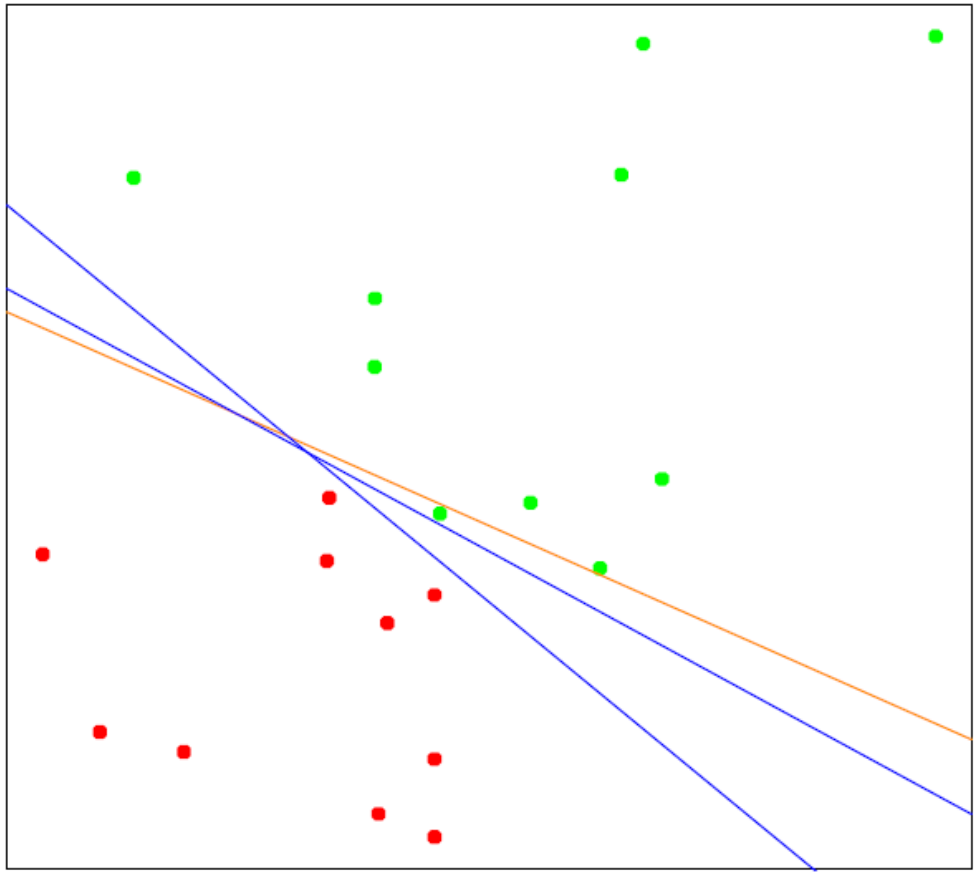

FIGURE: A toy example with two classes separable by a hyperplane. The orange line is the least squares solution, which misclassifies one of the training points. Also shown are two blue separating hyperplanes found by the perceptron learning algorithm with different random starts.

The above figure shows 20 data points in two classes in $\mathcal{R}^2$.

These data can be separated by a linear boundary.

Included in the figure (blue lines) are two of the infinitely many possible **separating hyperplanes**.

Optimal Separating Hyperplanes

The **optimal separating hyperplane separates** the two classes and maximizes the distance to the closest point from either class.

Not only does this provide a unique solution to the separating hyperplane problem, but by maximizing the margin between the two classes on the training data, this leads to better classification performance on test data.

This idea leads to the method called **support vector machine**.