Classification (Chapter 4)

Classification: Prediction for a qualitative Y based on p predictors $X_1, \ldots, X_p$ denoted by X

Values of Y: Class labels $k = 1, \ldots, K$ — **unordered**

Training data: (Y_i, X_i), $i = 1, \ldots, n$

Predicted value for a given x: $\hat{Y}$ (a class label)

Error: Zero-one error, i.e.,

$$I(\hat{Y} \neq Y) = \begin{cases} 1, & \text{if } \hat{Y} \neq Y \\ 0, & \text{if } \hat{Y} = Y \end{cases}$$

Expected error rate: $E\{I(\hat{Y} \neq Y)\} = P(\hat{Y} \neq Y)$ — **prob of misclassification**

Bayes classifier: Predicts the most likely class, i.e., the class k for which $p_k(x) = P(Y = k|x)$ is maximum — **optimal in that it minimizes the expected error rate**

Bayes error rate: $1 - E\{\max_k p_k(X)\}$ — **expected error rate for the Bayes classifier** — provides a **lower bound**

Issue: $p_k(x)$ **is unknown** — need to estimate it from training data so that $\hat{p}_k(x)$ can be used for classification.

Posterior probability: $p_k(x) = P(Y = k|X = x)$ — prob of falling in class k **conditional** on predictor x; $\sum_k p_k(x) = 1$

Prior probability: $\pi_k = P(Y = k)$ — **marginal** probability of falling in class k (aka **prevalence**); $\sum_k \pi_k = 1$

Class-conditional distribution: $f_k(x)$ — joint pdf/pmf of $X|Y = k$, i.e., distribution of X for an observation that comes from class k. In other words, if X is discrete, $f_k(x)$ represents $P(X = x|Y = k)$, whereas if X is continuous, $f_k(x)$ represents the pdf of X conditional on $Y = k$.

Q: Why do the class-conditional distributions matter?
A: If they are well-separated, the classification is easier.

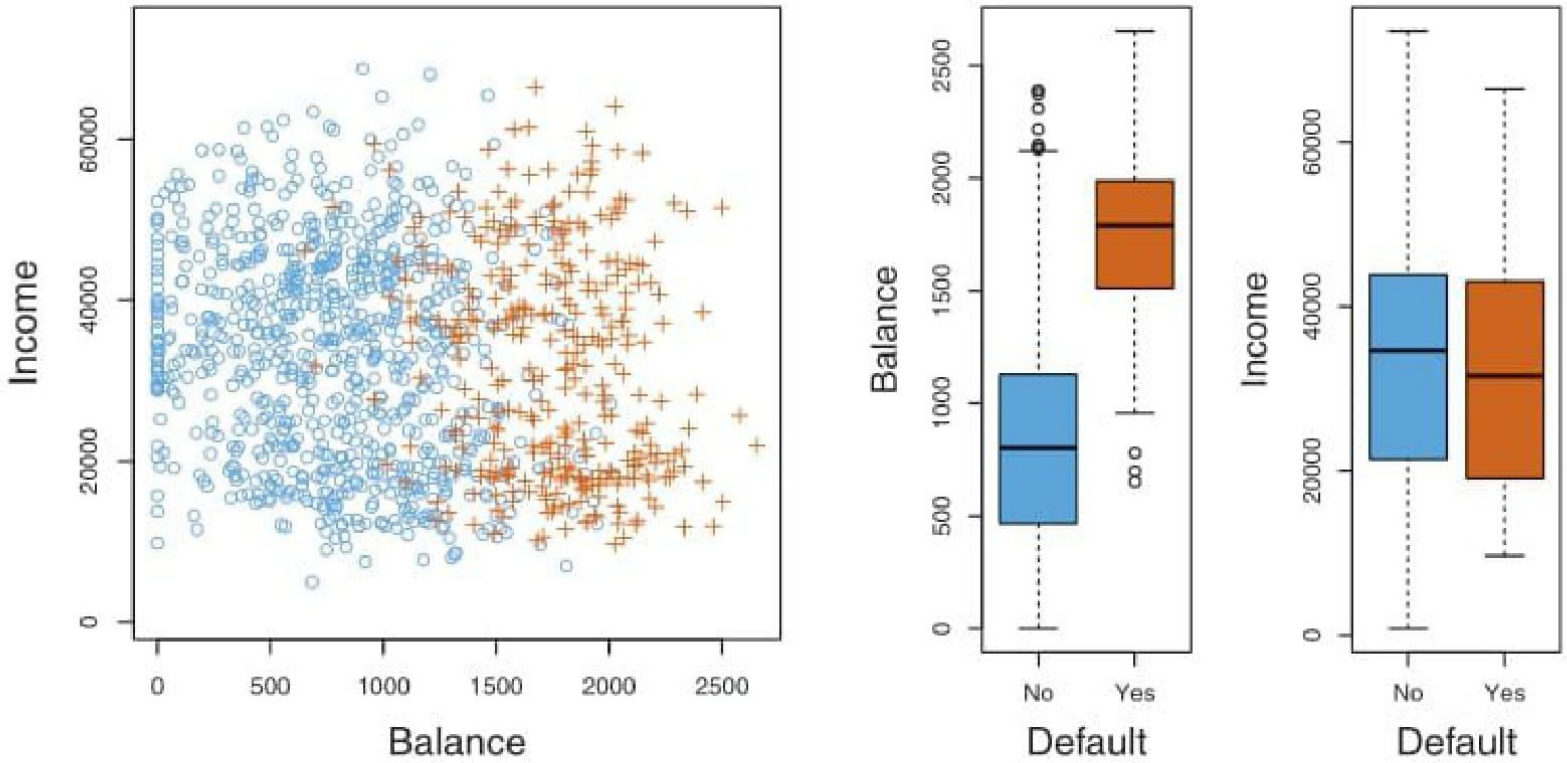

FIGURE 4.1. *The* `Default` *data set.* Left: *The annual incomes and monthly credit card balances of a number of individuals. The individuals who defaulted on their credit card payments are shown in orange, and those who did not are shown in blue.* Center: *Boxplots of* `balance` *as a function of* `default` *status.* Right: *Boxplots of* `income` *as a function of* `default` *status.*

Source: ISL

Discriminant Analysis for $K = 2$ Classes

Bayes classifier: Assign x to class 1 if

$$p_1(x) > p_2(x) \equiv \pi_1 f_1(x) > \pi_2 f_2(x) \equiv \delta_1(x) > \delta_2(x),$$

where $\delta_k(x)$ is called the *discriminant function*. It is obtained by starting with $\log(\pi_k f_k(x))$ and dropping terms that are common to both class-conditionals because they cancel out upon differencing.

- Both $p_k(x)$ and $\delta_k(x)$ induce the same ordering of classes, implying that we can focus on the latter
- *Bayes decision boundary*: $\{x : \delta_1(x) = \delta_2(x)\}$
- Assign x to class 2 if $\delta_1(x) < \delta_2(x)$
- If x is such that $\delta_1(x) = \delta_2(x)$, break tie randomly

Assumption: $X|Y = k \sim N(\mu_k, \mathbf{\Sigma}_k)$, i.e.,

$$f_k(x) = \frac{1}{(2\pi)^{p/2}|\mathbf{\Sigma}_k|^{1/2}} \exp\left\{-\frac{1}{2}(x-\mu_k)^T\mathbf{\Sigma}_k^{-1}(x-\mu_k)\right\}$$

- Class-conditionals are p-dimensional normal
- $E(X|Y = k) = \mu_k$, $\text{var}(X|Y = k) = \mathbf{\Sigma}_k$
- $|\mathbf{\Sigma}_k|$ = determinant of $\mathbf{\Sigma}_k$

Verify:

$$\delta_k(x) = -\frac{1}{2}\log(|\mathbf{\Sigma}_k|) - \frac{1}{2}x^T\mathbf{\Sigma}_k^{-1}x + x^T\mathbf{\Sigma}_k^{-1}\mu_k - \frac{1}{2}\mu_k^T\mathbf{\Sigma}_k^{-1}\mu_k + \log(\pi_k)$$

In addition, if $\mathbf{\Sigma}_1 = \mathbf{\Sigma}_2 = \mathbf{\Sigma}$ is assumed,

$$\delta_k(x) = x^T\mathbf{\Sigma}^{-1}\mu_k - \frac{1}{2}\mu_k^T\mathbf{\Sigma}^{-1}\mu_k + \log(\pi_k)$$

Note:

- **Class-specific cov matrix**: $\delta_k(x)$ is *quadratic* in x
- **Common cov matrix for all classes**: $\delta_k(x)$ is *linear* in x

Linear Discriminant Analysis (LDA), $X|Y = k \sim N(\mu_k, \Sigma)$

To use $\delta_k(x)$ in practice, we need to estimate μ_k and $\mathbf{\Sigma}$ (and also π_k if unknown) from the training data. For class k, let

- n_k = total number of observations
- $\overline{x}_k$ and $\mathbf{S}_k$ = sample mean and covariance matrix of predictor values

Estimates:

$$\hat{\pi}_k = \frac{n_k}{n}, \quad \hat{\mu}_k = \overline{x}_k, \quad \hat{\mathbf{\Sigma}} = \frac{(n_1 - 1)\mathbf{S}_1 + (n_2 - 1)\mathbf{S}_2}{n - 2}$$

- "Natural" estimators
- $\hat{\mathbf{\Sigma}}$ is the **pooled sample covariance matrix**
- Plug in to get:

$$\hat{\delta}_k(x) = \hat{\mu}_k^T \hat{\mathbf{\Sigma}}^{-1} x - \frac{1}{2} \hat{\mu}_k^T \hat{\mathbf{\Sigma}}^{-1} \hat{\mu}_k + \log(\hat{\pi}_k)$$

LDA Decision Rule

Assign x to class 1 if $\hat{\delta}_1(x) - \hat{\delta}_2(x) > 0$ or equivalently (verify)

$$(\hat{\mu}_1 - \hat{\mu}_2)^T \hat{\mathbf{\Sigma}}^{-1} x > \frac{1}{2}(\hat{\mu}_1 - \hat{\mu}_2)^T \hat{\mathbf{\Sigma}}^{-1}(\hat{\mu}_1 + \hat{\mu}_2) + \log\left(\frac{\hat{\pi}_2}{\hat{\pi}_1}\right)$$

Let $\hat{a}^T = (\hat{\mu}_1 - \hat{\mu}_2)^T \hat{\mathbf{\Sigma}}^{-1}$ and c be the cutoff on the RHS. Thus, the LDA decision rule assigns x to class 1 if $\hat{a}^T x > \hat{c}$.

LDA decision boundary: $\{x : \hat{a}^T x = \hat{c}\}$ — **linear**

- **Estimated Bayes classifier** under the assumption of normality with equal variance matrix
- **Does not** share the optimality property of the Bayes classifier because the unknowns therein are replaced with their estimates from the training data. However, when the assumptions hold, it often tends to approximate the Bayes classifier quite well
- If $\hat{\pi}_1 = \hat{\pi}_2$, the $\log(\hat{\pi}_k)$ term drops out from $\hat{\delta}_k(x)$.

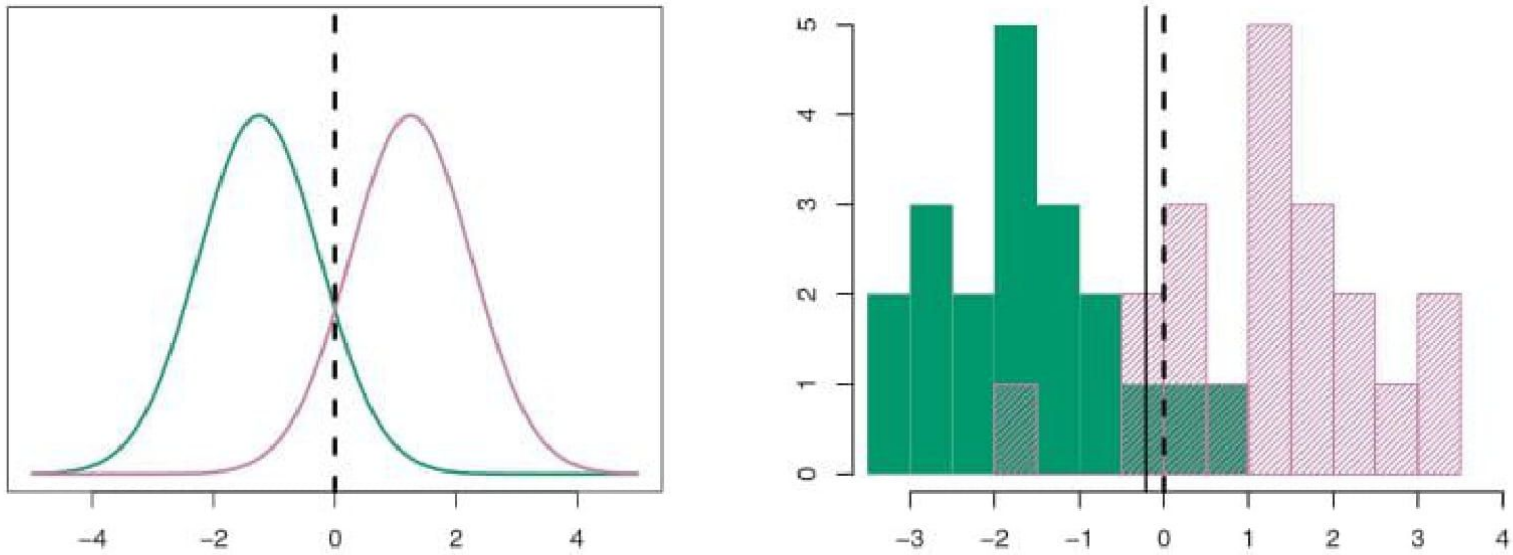

FIGURE 4.4. Left: *Two one-dimensional normal density functions are shown. The dashed vertical line represents the Bayes decision boundary.* Right: *20 observations were drawn from each of the two classes, and are shown as histograms. The Bayes decision boundary is again shown as a dashed vertical line. The solid vertical line represents the LDA decision boundary estimated from the training data.*

Source: ISL

- **Decision boundary**: (**quadratic**)

$$\left\{x : -\frac{1}{2}x^T\left(\hat{\mathbf{\Sigma}}_1^{-1} - \hat{\mathbf{\Sigma}}_2^{-1}\right)x + \left(\hat{\mu}_1^T\hat{\mathbf{\Sigma}}_1^{-1} - \hat{\mu}_2^T\hat{\mathbf{\Sigma}}_2^{-1}\right)x = \hat{c}\right\}$$

- **Estimated Bayes classifier** under the assumption of normality with unequal variance matrix
- Not optimal as the unknowns in the Bayes classifier are replaced by estimates. However, if the assumptions hold, it approximates the Bayes classifier quite well.
- If $\hat{\pi}_1 = \hat{\pi}_2$, the $\log(\hat{\pi}_k)$ term drops out from $\hat{\delta}_k(x)$
- Reduces to LDA when $\hat{\mathbf{\Sigma}}_k$ is replaced by a common $\hat{\mathbf{\Sigma}}$

Discriminant Analysis for $K > 2$ Classes

The extension to $K > 2$ case is straightforward keeping in mind that the Bayes classifier assigns x to the class for which $p_k(x)$ and hence $\delta_k(x)$ is largest. Just let the index k run from 1 to K, and replace the comparison of $\delta_1(x)$ and $\delta_2(x)$ with that of $\delta_k(x)$ and $\delta_l(x)$ for $k \neq l$. To get the decision boundaries, we need to divide the predictor space into K regions $R_1, \ldots, R_K$ such that R_k is the region for which the predicted class is k, i.e.,

$$R_k = \{x : \delta_k(x) > \delta_l(x) \text{ for all } l \neq k\}$$

In case of LDA with $p = 2$, the decision boundaries will be formed by intersecting lines. Of course, in application, we will use the estimated $\delta_k(x)$.

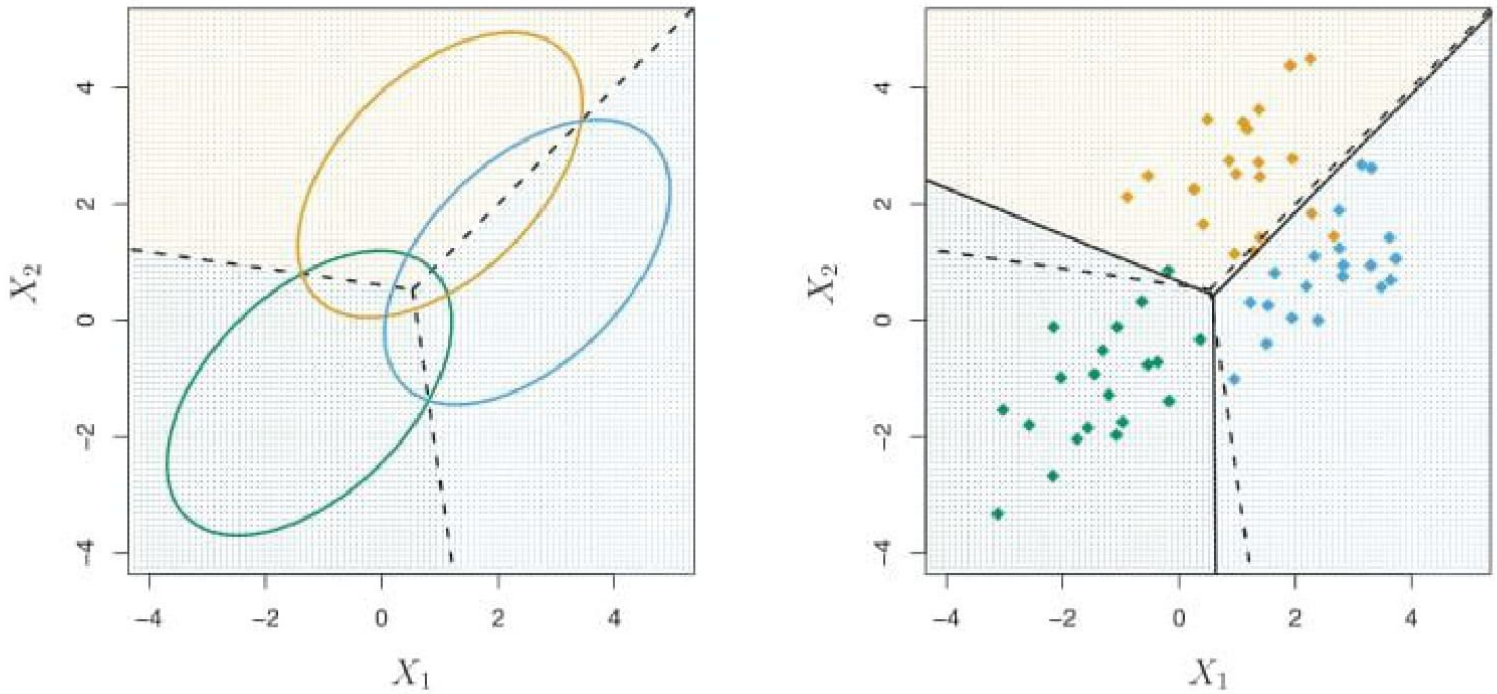

FIGURE 4.6. *An example with three classes. The observations from each class are drawn from a multivariate Gaussian distribution with $p = 2$, with a class-specific mean vector and a common covariance matrix.* Left: *Ellipses that contain 95 % of the probability for each of the three classes are shown. The dashed lines are the Bayes decision boundaries.* Right: 20 *observations were generated from each class, and the corresponding LDA decision boundaries are indicated using solid black lines. The Bayes decision boundaries are once again shown as dashed lines.*

Source: ISL

LDA vs QDA

- **Just one difference**: common covariance matrix or class-specific covariance matrices
- **Bias-variance tradeoff** — $Kp(p+1)/2$ cov parameters for QDA whereas only $p(p+1)/2$ for LDA — both have Kp mean parameters
- LDA is simple and is less flexible (and tends to have higher bias but lower variance) than QDA
- In practice, QDA generally works better than LDA when the training set is large (so that the variance of classifier the not a major concern) or if the equal variance assumption is clearly wrong
- Sometimes QDA leads to rather odd decision boundaries (e.g., disjointed intervals in case of $p = 1$)
- QDA is more sensitive than LDA to normality assumption
- **Proof is in the pudding** — try both and compare!

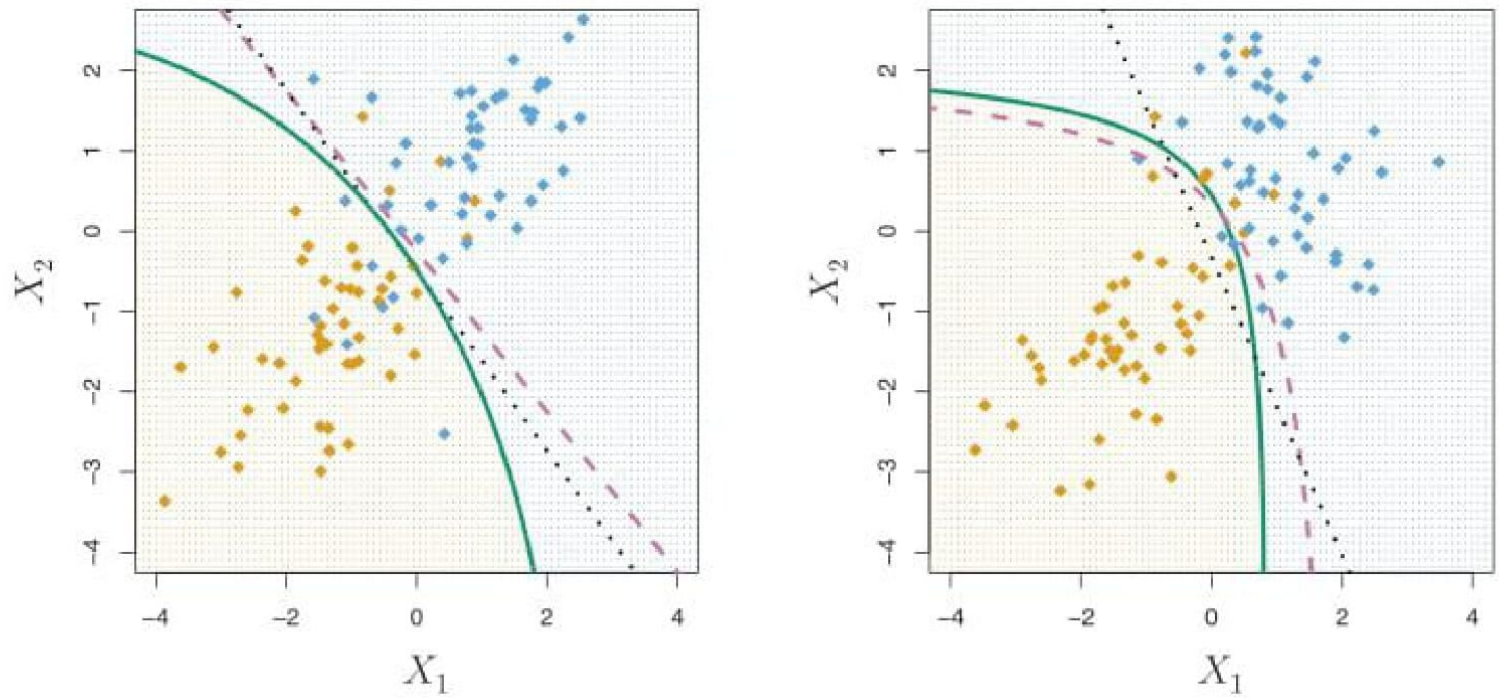

FIGURE 4.9. Left: *The Bayes (purple dashed), LDA (black dotted), and QDA (green solid) decision boundaries for a two-class problem with* $\boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_2$. *The shading indicates the QDA decision rule. Since the Bayes decision boundary is linear, it is more accurately approximated by LDA than by QDA.* Right: *Details are as given in the left-hand panel, except that* $\boldsymbol{\Sigma}_1 \neq \boldsymbol{\Sigma}_2$. *Since the Bayes decision boundary is non-linear, it is more accurately approximated by QDA than by LDA.*

Source: ISL

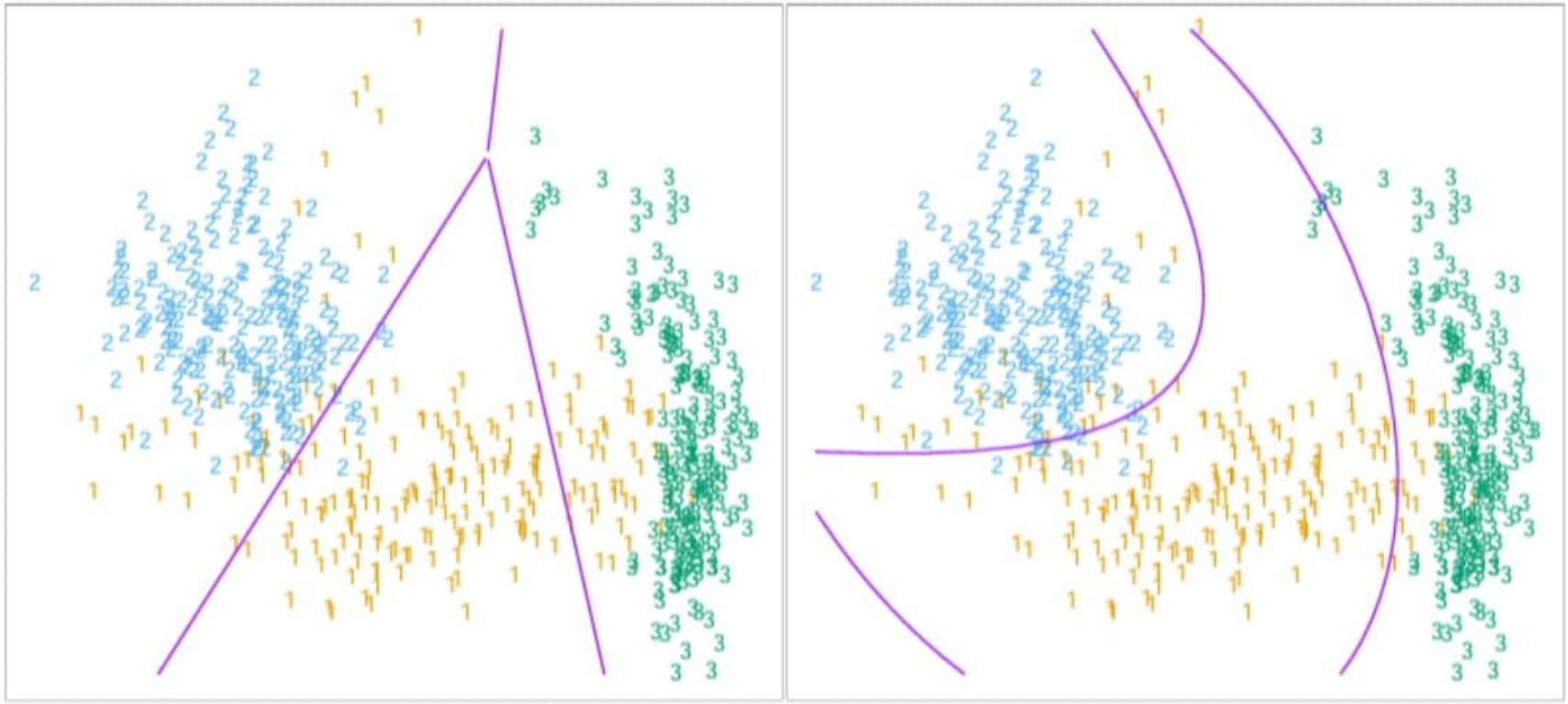

FIGURE 4.1. *The left plot shows some data from three classes, with linear decision boundaries found by linear discriminant analysis. The right plot shows quadratic decision boundaries. These were obtained by finding linear boundaries in the five-dimensional space* $X_1, X_2, X_1X_2, X_1^2, X_2^2$. *Linear inequalities in this space are quadratic inequalities in the original space.*

Regularized Discriminant Analysis (RDA)

- A compromise between LDA and QDA.
- Shrinks the separate covariance matrices of QDA ($\hat{\mathbf{\Sigma}}_\mathbf{k}$) toward a common covariance matrix as in LDA — *similar to ridge regression*.
- Regularized covariance matrices have the form:

$$\hat{\mathbf{\Sigma}}_\mathbf{k}(\alpha) = \alpha\hat{\mathbf{\Sigma}}_\mathbf{k} + (1-\alpha)\hat{\mathbf{\Sigma}}$$

  where $\hat{\mathbf{\Sigma}}$ is the pooled covariance used in LDA and $\alpha \in [0, 1]$.

- In practice, α can be chosen based on model performance assessment on validation data, or by cross validation.
- RDA-Applications in Python

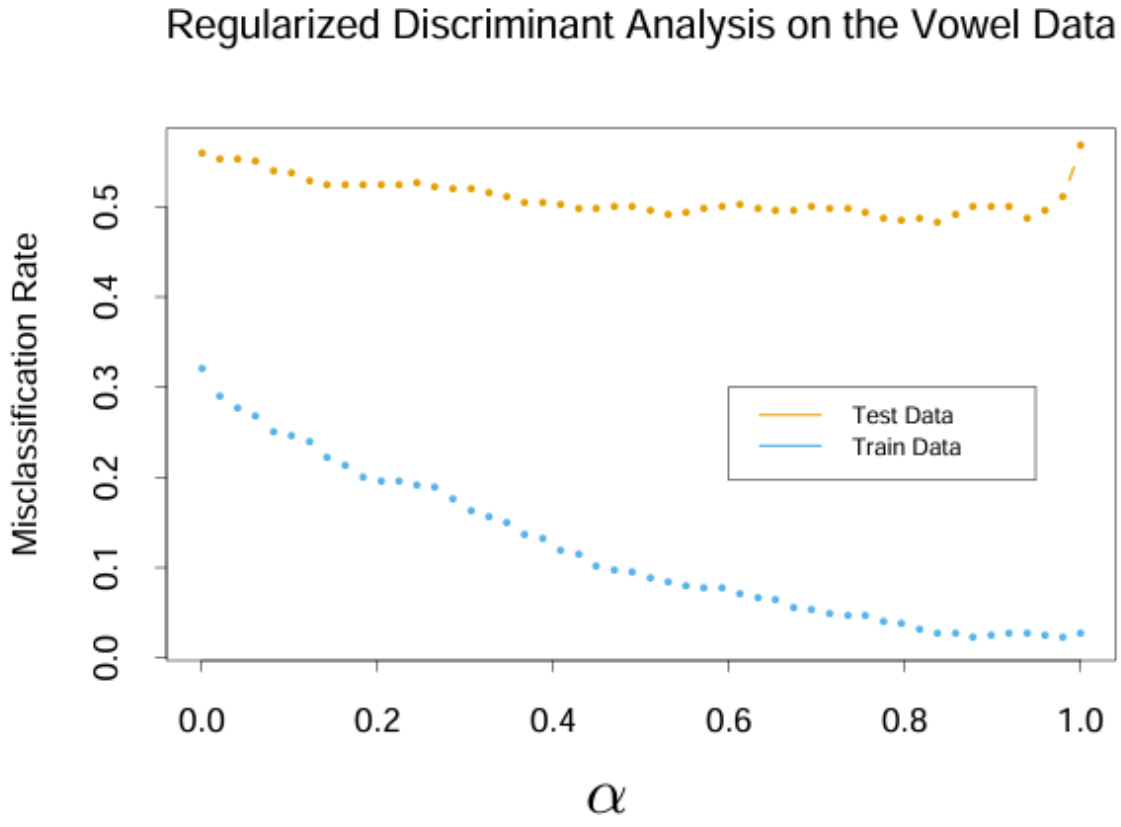

FIGURE 4.7. *Test and training errors for the vowel data, using regularized discriminant analysis with a series of values of* $\alpha \in [0, 1]$. *The optimum for the test data occurs around* $\alpha = 0.9$, *close to quadratic discriminant analysis.*

Performance Measures for a Binary Classifier (i.e., $K = 2$)

Bayes classifier: Minimizes **expected error rate**, or equivalently, $P(\hat{Y} \neq Y)$ — **probability of misclassification.** We can write this probability as

$$
\begin{aligned}
& P(\hat{Y} \neq Y, Y = 1) + P(\hat{Y} \neq Y, Y = 2) \\
&= P(\hat{Y} \neq Y | Y = 1) P(Y = 1) + P(\hat{Y} \neq Y | Y = 2) P(Y = 2) \\
&= P(\hat{Y} = 2 | Y = 1) P(Y = 1) + P(\hat{Y} = 1 | Y = 2) P(Y = 2) \\
&= P(\hat{Y} = 2 | Y = 1) \pi_1 + P(\hat{Y} = 1 | Y = 2) \pi_2
\end{aligned}
$$

- **Two class-specific errors**: Predict 1 as 2 and 2 as 1
- $P(\hat{Y} \neq Y)$ combines probabilities of the class-specific errors and the marginal proportions into one overall measure — can call it **total probability of misclassification**
- Overall error may be low but the class-specific errors may be high

Measures of Class-Specific Performance

Oftentimes, we can think of a classifier as a diagnostic test with

- Class 1: $+$ — aka **non-null** class (indicates a change from the normal state, e.g., a disease)
- Class 2: $-$ — aka **null** class (indicates no change from the normal state, e.g., no disease)

		Predicted class		
		$-$ or Null	$+$ or Non-null	Total
True class	$-$ or Null	True Neg. (TN)	False Pos. (FP)	N
	$+$ or Non-null	False Neg. (FN)	True Pos. (TP)	P
	Total	N*	P*	

TABLE 4.6. *Possible results when applying a classifier or diagnostic test to a population.*

- aka **confusion matrix** — correct predictions on the diagonal; misclassifications on the off-diagonal

Source: ISL

Name	Definition	Synonyms
False Pos. rate	FP/N	Type I error, 1−Specificity
True Pos. rate	TP/P	1−Type II error, power, sensitivity, recall
Pos. Pred. value	TP/P*	Precision, 1−false discovery proportion
Neg. Pred. value	TN/N*	

TABLE 4.7. *Important measures for classification and diagnostic testing, derived from quantities in Table 4.6.*

- **True positive rate:** $P(\hat{Y} = +|Y = +)$ — **sensitivity**
- **True negative rate:** $P(\hat{Y} = -|Y = -)$ — **specificity**
- **False positive rate:** $P(\hat{Y} = +|Y = -)$
- **False positive rate + true negative rate = 1**, i.e., **false positive rate = 1− specificity**
- **False positive rate + false negative rate ≠ 1**
- **Class-specific error rates:** **1** − sensitivity and **1** − specificity
- Want both sensitivity and specificity to be high, or equivalently, the two error rates to be small.

Source: ISL

Returning to **probability of misclassification**, we see that:

$$P(\hat{Y} \neq Y) = (1 - \text{sensitivity})\pi_+ + (1 - \text{specificity})\pi_-$$
$$= \frac{\text{FN}}{\text{P}} \frac{\text{P}}{\text{N}+\text{P}} + \frac{\text{FP}}{\text{N}} \frac{\text{N}}{\text{N}+\text{P}} = \frac{\text{FN}+\text{FP}}{\text{N}+\text{P}},$$

which is as expected. Next, consider a general classifier that predicts class '+' if $p_+(x) \geq p$ and class '−' otherwise, where p is a cutoff for the posterior probability. Of course, the Bayes classifier uses $p = 0.5$.

Q: What would be the sensitivity and specificity if $p = 0$?
A: $p = 0 \implies$ everybody is classified as '+'
$\implies$ sensitivity $= \text{TP}/\text{P} = 1$ and specificity $= \text{TN}/\text{N} = 0$

Q: What would be the sensitivity and specificity if $p = 1$?
A: $p = 1 \implies$ everybody is classified as '−'
$\implies$ sensitivity $= \text{TP}/\text{P} = 0$ and specificity $= \text{TN}/\text{N} = 1$

Tradeoff Between Sensitivity & Specificity

Fact: In general, as the cutoff $p \uparrow 1$, the sensitivity $\downarrow 0$, and the specificity $\uparrow 1$.

- **tradeoff between sensitivity and specificity** — if one increases, the other decreases
- Effect of p on probability of misclassification is not as clear cut as it would depend on the relative rates of change and also on the marginal proportions. However, for a Bayes classifier, this probability will be minimum when $p = 0.5$.
- Can plot both sensitivity and specificity (or their one-minus versions) against the cutoff p to see the class-specific performance of a classifier

Default data

- class '+': `default = yes`, class '−': `default = no`
- Predictors: `balance` and `student` status — note: using LDA with a qualitative predictor (a common practice)
- sensitivity $= P(+|+) = P$(correctly predict a defaulter)
- specificity $= P(-|-) = P$(correctly predict a non-defaulter)

		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

TABLE 4.4. *A confusion matrix compares the LDA predictions to the true default statuses for the* 10,000 *training observations in the* `Default` *data set. Elements on the diagonal of the matrix represent individuals whose default statuses were correctly predicted, while off-diagonal elements represent individuals that were misclassified. LDA made incorrect predictions for* 23 *individuals who did not default and for* 252 *individuals who did default.*

- With $p = 0.5$: overall error rate $= (23 + 252)/10000 =$ 2.8%, sensitivity $= 81/333 = 24.3\%$, specificity $=$ $9644/9667 = 99.8\%$ — **sensitivity too low**

		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

TABLE 4.5. *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 %.

- With $p = 0.2$: overall error rate = (235+138)/10000 = 3.7%, sensitivity = 195/333 = 58.6%, specificity = 9432/9667 = 97.6% — **may be more acceptable**

Source: ISL

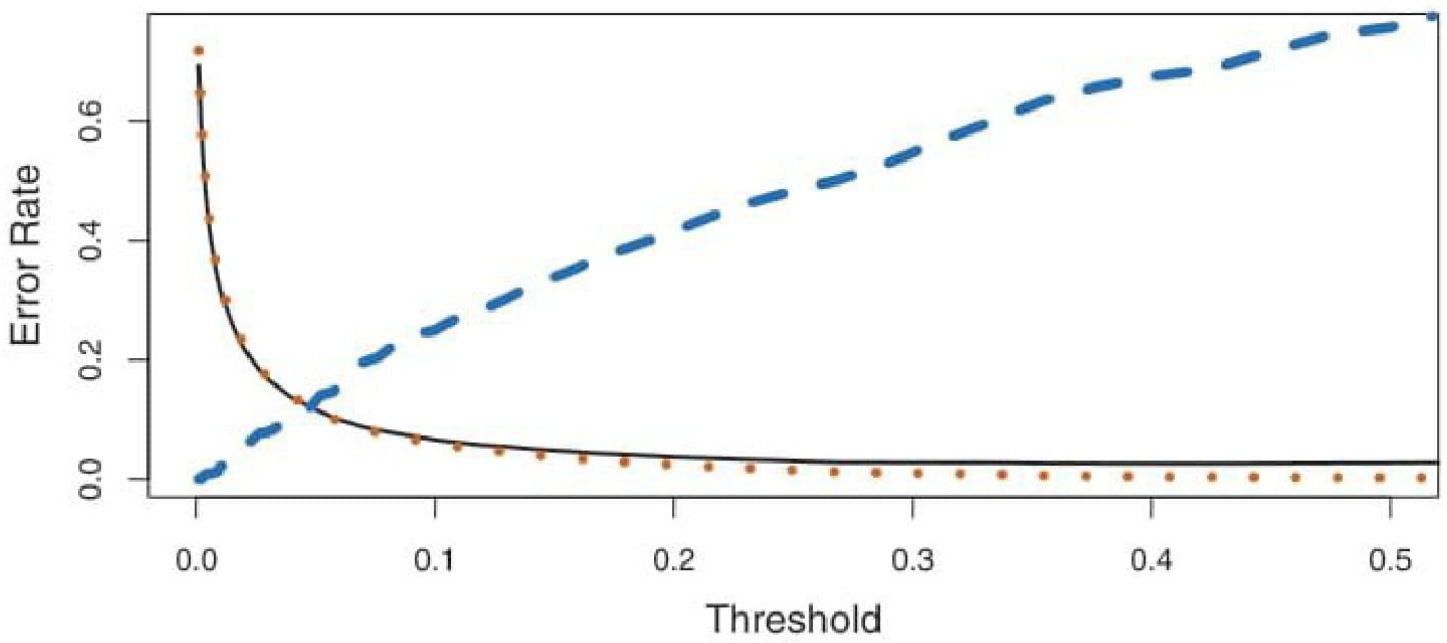

FIGURE 4.7. *For the* `Default` *data set, error rates are shown as a function of the threshold value for the posterior probability that is used to perform the assignment. The black solid line displays the overall error rate. The blue dashed line represents the fraction of defaulting customers that are incorrectly classified, and the orange dotted line indicates the fraction of errors among the non-defaulting customers.*

Source: ISL

ROC Curve

Receiver Operating Characteristic (ROC) Curve: Plot of sensitivity (i.e., true positive rate) against $1 -$ specificity (i.e., false positive rate)

- Set up a fine grid of cutoffs $p \in (0, 1)$. For each cutoff p on this grid, compute the corresponding sensitivity and $1 -$ specificity, and plot the former against the latter.
- **Perfect classifier**: Inverted L-shaped ROC curve — both error rates are zero
- **Random guess classifier**: It randomly decides the class by tossing a coin with $P(H) = p$ — predicts '+' in case of heads and '$-$' in case of tails — **no use of data at all**

Q: What is the ROC curve for the random guess classifier?
A: sensitivity $= P(+|+) = P(+) = p$, and specificity $= P(-|-) = P(-) = 1 - p$, implying that the ROC curve is a 45-degree line

- A 45-degree line is superimposed on the curve for reference
- **Area under the ROC curve (AUC)** — an overall (but scalar) measure of performance. It equals 1 for **perfect classifier** and 0.5 for **random guess classifier**
- **Another interpretation**: Randomly select two subjects from the population — one '+' and one '−.' AUC represents the probability that the '+' subject is more likely to be classified as '+' than the '−' subject.
- ROC curve and AUC are used to compare classifiers. For example, if the ROC curve of one classifier lies entirely above that of the other, the former is clearly superior than the latter. However, the ROC curves often cross.
- In practice, look at the ROC curve and use **domain knowledge** to come up with acceptable values for sensitivity and specificity. Then, use the corresponding posterior probability cutoff for classification.
- Other measures available for comparison of classifiers (e.g., compare observed vs expected counts, etc.)

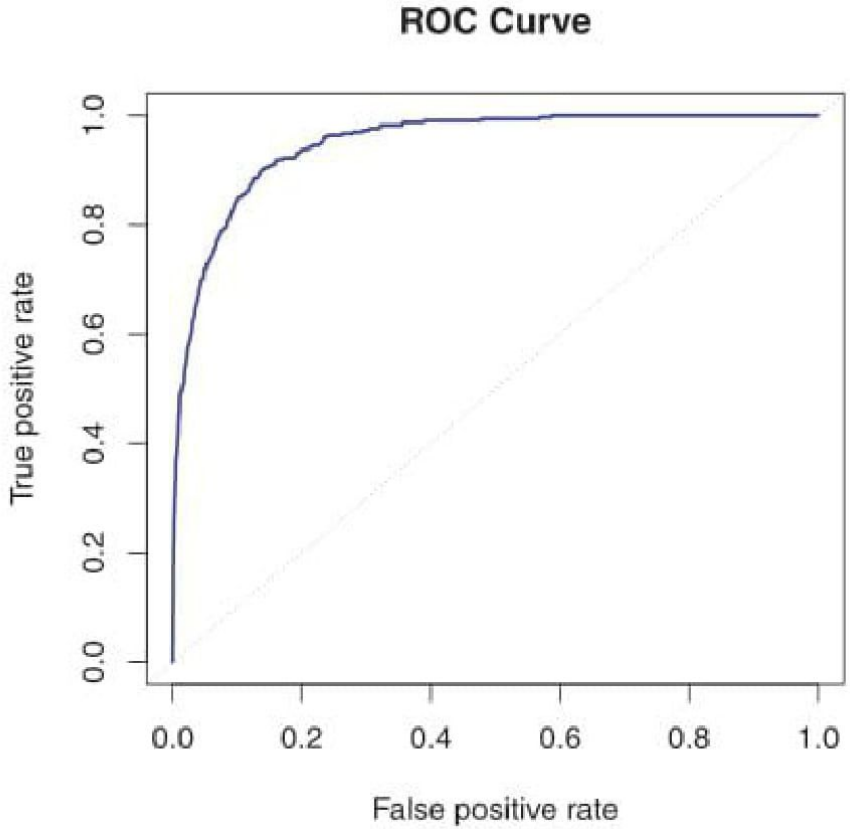

FIGURE 4.8. *A ROC curve for the LDA classifier on the* `Default` *data. It traces out two types of error as we vary the threshold value for the posterior probability of default. The actual thresholds are not shown. The true positive rate is the sensitivity: the fraction of defaulters that are correctly identified, using a given threshold value. The false positive rate is 1-specificity: the fraction of non-defaulters that we classify incorrectly as defaulters, using that same threshold value. The ideal ROC curve hugs the top left corner, indicating a high true positive rate and a low false positive rate. The dotted line represents the "no information" classifier; this is what we would expect if student status and credit card balance are not associated with probability of default.*

Source: ISL

Logistic Regression for Binary Data

Discriminant analysis: Model $f_k(x)$ and use Bayes theorem to get $p_k(x)$

Logistic regression: Model $p_k(x)$ directly — enough to focus on $p_1(x)$ as $p_2(x) = 1 - p_1(x)$.

Assume: The two classes are coded as 0/1 — 1 for 'success', 0 for 'failure.' Thus, the response $Y \sim$ Bernoulli (p), where $p = P(Y = 1) = E(Y)$. Letting p depend on the covariate vector X, we get $p(x) = P(Y = 1|X = x) = E(Y|X = x)$.

A common statistical modeling principle: Find an $\uparrow$ function g of $E(Y|X)$ whose value can be any real number & use

$$g\{E(Y|X = x)\} = x^T\beta.$$

- RHS is linear in β — **linear model structure**
- **Linear model**: g is identity, i.e., $g\{E(Y|X)\} = E(Y|X)$
- What is the need for g, aka **link function**?

How to model $p(x)$?

Linear regression: $p(x) = x^T\beta$ — **identity link**

- No guarantee that $p(x) \in (0, 1)$ for all x

Probit regression: $\Phi^{-1}\{p(x)\} = x^T\beta$, where Φ is the CDF of a $N(0, 1)$ distribution — **probit link**, i.e., $g = \Phi^{-1}$

- $p(x) = \Phi(x^T\beta)$ — always in $(0, 1)$

Logistic regression (our focus): $\text{logit}\{p(x)\} = x^T\beta$, where

$$\text{logit}\{p(x)\} = \log\left\{\frac{p(x)}{1-p(x)}\right\}$$

is the **logit** or **log-odds** function — **logit link**, i.e., $g = \text{logit}$

- $p(x) = \frac{\exp(x^T\beta)}{1+\exp(x^T\beta)}$ — **logistic function** — always in $(0, 1)$
- As $p \uparrow$ in $(0, 1)$, odds $\uparrow$ in $(0, \infty)$. Thus, odds close to zero or ∞ indicate very small or very large probabilities.

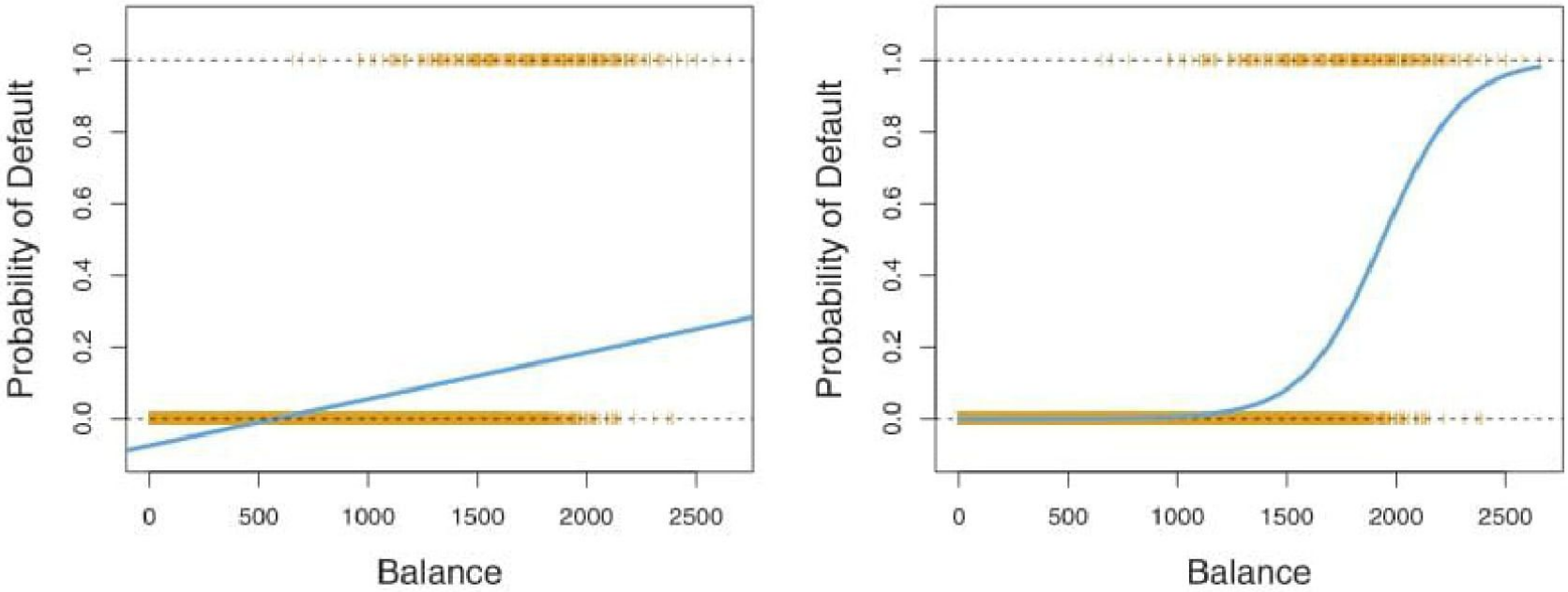

FIGURE 4.2. *Classification using the* `Default` *data.* Left: *Estimated probability of* `default` *using linear regression. Some estimated probabilities are negative! The orange ticks indicate the 0/1 values coded for* `default` *(*`No` *or* `Yes`*).* Right: *Predicted probabilities of* `default` *using logistic regression. All probabilities lie between* 0 *and* 1.

Source: ISL

Interpreting Logistic Regression Coeffs

Case 1: One continuous predictor X. The model is

$$\log\left\{\frac{p(x)}{1-p(x)}\right\} = \beta_0 + \beta_1 x \text{ — } \textbf{logit is linear in } x$$

Here β_0 and β_1 are intercept and slope of the logit function. Thus, β_1 is the change in logit when $x \uparrow$ by one unit. Since

$$\begin{aligned}\beta_1 &= \beta_0 + \beta_1(x+1) - \beta_0 - \beta_1 x \\ &= \log\left\{\frac{p(x+1)}{1-p(x+1)}\right\} - \log\left\{\frac{p(x)}{1-p(x)}\right\} \\ &= \log(\text{odds ratio}), \text{ where odds ratio} = \frac{\text{new odds}}{\text{old odds}},\end{aligned}$$

it follows that **new odds** $= \exp(\beta_1) \times$ **old odds**.

- $\beta_1 > 0$: odds and hence $p(x)$ increase with x
- $\beta_1 < 0$: odds and hence $p(x)$ decrease with x
- $\beta_1 = 0$: $p(x)$ is free of x — X is not useful for predicting Y
- If $p(x)$ is small, odds ratio $\approx p(x+1)/p(x)$ — **relative risk**

Case 2: One categorical predictor X with C levels, coded using $C-1$ **indicator variables**, $Z_1, \ldots, Z_{C-1}$. The model is

$$\log\left\{\frac{p(z)}{1-p(z)}\right\} = \beta_0 + \beta_1 z_1 + \ldots + \beta_{C-1} z_{C-1}.$$

As before, **logit for base level** $= \beta_0$ and **logit for level** $j = \beta_0 + \beta_j$. Therefore,

$$\begin{aligned}\beta_j &= \text{logit for level } j - \text{ logit for base level} \\ &= \log(\text{odds ratio}), \text{ where odds ratio} = \frac{\text{odds for level } j}{\text{odds for base level}},\end{aligned}$$

meaning **odds for level** $j = \exp(\beta_j) \times$ **odds for base level**.

- Odds and hence $p(z)$ are larger than those for base if $\beta_j > 0$ and they are smaller than those for base if $\beta_j < 0$
- Odds ratio of level j vs $k = \exp(\beta_j - \beta_k)$
- No effect of X: $\beta_1 = \ldots = \beta_{C-1} = 0$ (simultaneously)
- **If $p(z)$ is small, odds ratio $\approx$ relative risk**

Ex: Suppose Y = **indicator of lung cancer** (1 = yes, 0 = no) and X = **smoking history** with three levels — never-smoker (base), occasional smoker (z_1), and serious smoker (z_2). We also know that proportion p of lung cancer patients in the general population is small. For example, in 2014, there were an estimated 527,228 people living with lung cancer in the US. Therefore, we can interpret an odds ratio as a relative risk. Now, **assume** that $(\beta_1, \beta_2) = (1.1, 3.0)$. Thus:

- *odds ratio for occasional vs never smokers* = $\exp(1.1) = 3 \implies$ an occasional smoker is three times more likely than a never-smoker to get lung cancer.
- *odds ratio for serious vs never smokers* = $\exp(3.0) = 20 \implies$ a serious smoker is 20 times more likely than a never-smoker to get lung cancer.
- *odds ratio for serious vs occasional smokers* = $\exp(3.0 - 1.1) = 6.7 \implies$ a serious smoker is about 7 times more likely than an occasional smoker to get lung cancer.

Case 3: Multiple predictors $X_1 \ldots, X_p$ — continuous or categorical (represented using indicators) and their interactions. In this case, we proceed in the same way as in the case of **linear model** except that

$$\text{logit}\{p(x)\} = x^T\beta,$$

rather than $E(Y|x) = x^T\beta$ is the model equation, and

$$p(x) = \frac{\exp(x^T\beta)}{1 + \exp(x^T\beta)}$$

Decision Rule Based on Logistic Regression

- Estimate β from training data and plug in to get $\text{logit}\{\hat{p}(x)\} = x^T\hat{\beta}$ and $\hat{p}(x)$.
- Estimated Bayes classifier predicts class 1 if $\hat{p}(x) \geq 0.5$, or equivalently, $\text{logit}\{\hat{p}(x)\} \geq 0$ (because $\text{logit}(0.5) = 0$)
- **Decision boundary**:

$$\{x : \hat{p}(x) = 0.5\} \equiv \{x : \text{logit}\{\hat{p}(x)\} = x^T\hat{\beta} = 0\}$$

- The decision boundary is linear — **just like LDA**
- Differs with LDA only in fitting — maximum likelihood for logistic regression and method of moments assuming normality for LDA
- With quantitative predictors, logistic regression and LDA tend to give similar classification performance
- May use cutoffs other than 0.5 for $p(x)$ to get specified sensitivity and specificity performance

Pros and Cons of Logistic Regression

Pros:

- Can be used for both inference (e.g., to select useful predictors) and prediction (whereas LDA and QDA are designed only for prediction)
- Works with both quantitative and qualitative predictors (although LDA and QDA are also often used with qualitative predictors)
- Does not have any distributional assumptions for predictors (whereas LDA and QDA work under normality assumption, which makes sense only for quantitative predictors)

Cons:

- Unstable if classes are well-separated (in fact, it will fail in case of perfect separation) or if n is small (whereas LDA and QDA do not have this issue)
- Can be generalized to $K > 2$ but LDA and QDA are more common in this case

Comparison of Various Classifiers

Classifiers: KNN, LDA, QDA, and logistic regression

- KNN is nonparametric (the others have assumptions)
- Logistic regression can be used for inference
- Excerpt from Section 4.5:

These six examples illustrate that no one method will dominate the others in every situation. When the true decision boundaries are linear, then the LDA and logistic regression approaches will tend to perform well. When the boundaries are moderately non-linear, QDA may give better results. Finally, for much more complicated decision boundaries, a non-parametric approach such as KNN can be superior. But the level of smoothness for a non-parametric approach must be chosen carefully. In the next chapter we examine a number of approaches for choosing the correct level of smoothness and, in general, for selecting the best overall method.

Source: ISL

Inference for Logistic Regression

Issue: How to estimate β, perform tests, and construct confidence intervals? The procedure is similar to that of linear models but with some changes. Let's discuss this procedure in the context of **generalized linear models** of which both linear regression and logistic regression models are special cases.

Generalized Linear Model (GLM)

LM: $Y \sim N(\mu, \sigma^2)$, $\mu = E(Y), \sigma^2 = \text{var}(Y)$. There are three components of this model:

1. **Random component**: Y follows a **normal** distribution
2. **Systematic component**: $\eta = x^T\beta$ — **linear predictor**
3. **Link function** that relates linear predictor with mean: $\eta = g(\mu) = \mu$ — **identity link**, i.e., $g(\mu) = \mu$

GLM: Generalizes LM in two ways:

1. **Random component**: Distribution of Y may be from an **exponential family** of distributions (well-known members: normal, binomial, Poisson, gamma, etc.)
2. **Link function**: g may be an increasing, differentiable function

Exponential Family of Distributions

A random variable Y has a distribution in the **exponential family** if its pdf/pmf can be written as

$$f_Y(y;\theta,\phi) = \exp\left\{\frac{y\theta - b(\theta)}{a(\phi)} + c(y,\phi)\right\}$$

for some specific functions $a(\cdot)$, $b(\cdot)$ and $c(\cdot)$.

- **Canonical form** with θ as **canonical parameter**
- ϕ: **dispersion parameter** — known or unknown
- $E(Y) = b'(\theta)$, $\text{var}(Y) = b''(\theta)a(\phi)$ (verify)

For example, verify that if $Y \sim N(\mu, \sigma^2)$, we have $\theta = \mu, \phi = \sigma^2$,

$$b(\theta) = \frac{\theta^2}{2}, a(\phi) = \phi, c(y, \phi) = -\frac{1}{2}\left\{\frac{y^2}{\sigma^2} + \log(2\pi - \sigma^2)\right\}.$$

Likewise, if $Y \sim \textbf{Bernoulli}(p)$, we have $\theta = \text{logit}(p), \phi = 1$,

$$b(\theta) = -\log\{1 + \exp(\theta)\}, a(\phi) = 1, c(y, \phi) = 0.$$

Moreover, if $Y \sim \textbf{Poisson}(\mu)$, we have $\theta = \log(\mu), \phi = 1$,

$$b(\theta) = \exp(\theta), a(\phi) = \phi, c(y, \phi) = -\log(y!).$$

Common examples of link function $g(\mu)$:

- $\log\{\mu/(1-\mu)\}$ — **logit link** with $\mu = p$
- $\Phi^{-1}(\mu)$ — **probit link** with $\mu = p$ and Φ as $N(0,1)$ cdf
- $\log(\mu)$ — **log link**
- $\log\{-\log(1-\mu)\}$ — **complementary log-log link**

Data: $(Y_i, X_i = (X_{i1}, X_{i2}, \ldots, X_{ip}))$, $i = 1, \ldots, n$ from n independent subjects and p predictors

Model: $Y_i \sim$ indep $f_{Y_i}(y_i; \theta_i, \phi)$ with $\eta_i = g(\mu_i) = x_i^T \beta$.

- μ_i involves θ_i
- If canonical link is chosen, $\eta_i = \theta_i = g(\mu_i)$
- **Parameters**: β and ϕ (if unknown)
- Reduces to LM for normal distribution with identity link
- **Log-likelihood function**: $l(\beta, \phi)$
- For a fixed ϕ, maximize $l(\beta, \phi)$ wrt β using **Fisher's scoring** method — a variant of Newton-Raphson. It's an iterative method that fits a suitably defined LM with weighted least squares in each iteration — **IRWLS** method
- **Estimation of** β: ML estimator $\hat{\beta}$
- **Estimation of** ϕ: Maximize $l(\hat{\beta}, \phi)$ wrt ϕ or use a method of moment type estimator to get $\hat{\phi}$

Inference in a GLM

Fitted values: $\hat{\mu}_i$, $i = 1, \ldots, n$ — estimates of the means μ_i

Distribution of $\hat{\beta}$: When n is large, $\hat{\beta} \approx N(\beta, \hat{\mathbf{J}}^{-1})$, where

$$\hat{\mathbf{J}} = -\frac{\partial^2 l(\beta, \phi)}{\partial \beta \, \partial \beta^T}$$

is the $p \times p$ **Hessian matrix** — aka **information matrix** — evaluated at $(\beta, \phi) = (\hat{\beta}, \hat{\phi})$, and has a closed-form expression.

Testing significance of jth coefficient: (**Wald test**)

$$H_0 : \beta_j = 0 \quad \text{vs.} \quad H_1 : \beta_j \neq 0$$

- **Test statistic:** $(\hat{\beta}_j - 0)/\text{SE}(\hat{\beta}_j) \approx N(0, 1)$ when H_0 is true and n is large
- **Rejection region** or p**-value** or **CI:** as before — use $N(0, 1)$ instead of t distribution, i.e., z-test instead of t-test

Comparing Two Nested Models

First, consider two models on extremes of the spectrum:

Null model: Only has a common intercept, i.e., a common mean (no predictors) — as simple as possible

- Regression does not explain any variation — all variation is consigned to the random component. It has $n - 1$ df.

Saturated model: Has one mean parameter per *observation* so that $\hat{\mu}_i = y_i$ — as complex as possible

- Regression explains all the variation — all variation is consigned to the systematic component, leaving none for the random component. It has 0 df.
- Does not summarize data, it merely reproduces them
- Serves as a baseline for comparing two models
- Definition of saturated model depends on how *observation* is defined, e.g., Bernoulli vs binomial

Analysis of Deviance

Let l_{model} denote the maximum log-likelihood for a model with p predictors (i.e., $p+1$ regression coefficients and $n-p-1$ df)

Deviance: $D_{\text{model}} = 2(l_{\text{saturated}} - l_{\text{model}})$

- ≥ 0
- Provides a measure of discrepancy of fit of the model. Use it to compare two models, rather than evaluating goodness of fit of a model because the definition of saturated model depends on how the data are organized
- **Linear model**: $D_{\text{model}} = SS_{\text{ERR}}$ — error or residual SS
- $D_{\text{model}} = \sum_{i=1}^{n} d_i^2$, where $r_i = \text{sign}(y_i - \mu_i)\sqrt{d_i}$ is called **deviance residual**. Other definitions of residuals exist.

	# reg coeff	(resid) df	(resid) deviance
Saturated	n	0	0
Null	1	$n-1$	D_{null}
Model	$p+1$	$n-p-1$	D_{model}

To compare two nested models — **full** and **reduced** with p_{full} and p_{reduced} regression coefficients, respectively, we compute the **change in deviance** statistic:

$$\begin{aligned}\Delta D &= D_{\text{reduced}} - D_{\text{full}} \\ &= 2(l_{\text{saturated}} - l_{\text{reduced}}) - 2(l_{\text{saturated}} - l_{\text{full}}) \\ &= 2(l_{\text{full}} - l_{\text{reduced}})\end{aligned}$$

When H_0 : **full model = reduced model** is true and n is large, ΔD approximately follows a χ^2 distribution with df equal to $p_{\text{full}} - p_{\text{reduced}}$, which also represents the difference in residual df of the two models.

- χ^2 **test** or **likelihood ratio test**
- analog of **partial F-test** from LM
- For one slope, reduces to z-test
- Taking null model as the reduced model, gives the **test of model significance**

Other Model Comparison Criteria

For models fit with ML method, other model comparison criteria exist, e.g., **Akaike Information Criterion** (AIC) and **Bayesian Information Criterion** (BIC):

$$\text{AIC} = -2l_{\text{model}} + 2(\#\text{ parameters})$$
$$\text{BIC} = -2l_{\text{model}} + \log(n)(\#\text{ parameters})$$

- Like deviance $D = 2(l_{\text{saturated}} - l_{\text{model}})$ but use a constant $\times$ # parameters instead of $l_{\text{saturated}}$
- Offer a compromise between the goodness-of-fit of a model (1st term) and the complexity of the model (2nd term).
- **Penalized measures** with 2nd term as a penalty for model complexity: more parameters $\Longrightarrow$ higher penalty
- Differ only in the penalty term
- **Smaller is better**
- Especially useful for comparing non-nested models