9. MÁQUINAS DE VETORES DE SUPORTE

Neste capítulo, discutimos a *máquina de vetor de suporte* (SVM), uma abordagem para classificação que foi desenvolvida na comunidade da ciência da computação nos anos 90 e que teve sua popularidade aumentada desde então. SVMs mostraram ter boa performance em uma variedade de contextos, e são frequentemente consideradas uns dos melhores classificadores não-convencionais.

A máquina de vetor de suporte é uma generalização de um classificador simples e intuitivo chamado o *classificador de margem máxima*, o qual introduzimos na seção 9.1. Apesar de ser elegante e simples, vemos que este classificador infelizmente não pode ser aplicado à maioria dos conjuntos de dados, já que este requer que as classes sejam separadas por uma fronteira linear. Na seção 9.2, introduzimos o *classificador de vetor de suporte*, uma extensão do classificador de margem máxima que pode ser aplicado a um espectro mais amplo de casos. A seção 9.3 introduz a *máquina de vetor de suporte*, que é uma extensão maior do classificador de vetor de suporte que acomoda fronteiras de classes não-lineares. Máquinas de vetores de suporte são projetadas para o contexto de classificação binária no qual há duas classes; na seção 9.4 discutimos extensões das máquinas de vetores de suporte para o caso em que há mais de duas classes. Na seção 9.5 discutimos as conexões próximas entre máquinas de vetores de suporte e outros métodos estatísticos como regressão logística.

Geralmente, as pessoas se referem ao classificador de margem máxima, o classificador de vetor de suporte e a máquina de vetor de suporte como "máquina de vetores de suporte". Para evitar confusão, distinguiremos cuidadosamente estas três noções neste capítulo.

9.1 Classificador de margem máxima

Nesta seção, definimos um hiperplano e introduzimos o conceito de um hiperplano de separação ótima.

9.1.1 O que é um hiperplano?

Em um espaco p-dimensional, um *hiperplano* é um subespaço plano afim de dimensão $p - 1$. Por exemplo, em duas dimensões, um hiperplano é um subespaço sem curvas unidimensional - em outras palavras, uma linha. Em três dimensões, um hiperplano é um subespaço sem curvas bidimensional - ou seja, um plano. Em dimensões $p > 3$, pode ser difícil de visualizar um hipeplano, mas a noção de um subespaço sem curvas $(p - 1)$-dimensional ainda se aplica.

A definição matemática de um hiperplano é simples. Em duas dimensões, um hiperplano é definido pela equação

$$\beta_0 + \beta_1 X_1 + \beta_2 X_2 = 0$$

(9.1)

para parâmetros β_0, β_1 e β_2. Quando dizemos que (9.1) "define" o hiperplano, queremos fizer que qualquer $X = (X_1, X_2)^T$ que (9.1) considera é um ponto no hiperplano. Note que (9.1) é simplesmente a equação de uma linha, já que, de fato, em duas dimensões, um hiperplano é uma linha.

A equação 9.1 pode ser facilmente extendida para um contexto $p-$dimensional:

$$\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_p X_p = 0$$

(9.2)

definindo um hiperplano $p-$dimensional, novamente no sentido de que se um ponto $X = (X_1, X_2, \ldots, X_p)^T$ em um espaço $p-$dimensional (por exemplo, um vetor de comprimento p) satisfaz (9.2), então X posiciona-se no hiperplano.

Agora, suponha que X não satisfaz (9.2):

$$\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_p X_p > 0$$

(9.3)

Então, isto nos diz que X posiciona-se em um lado do hiperplano. Por outro lado, se

$$\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_p X_p < 0 \tag{9.4}$$

Então X se encontra no outro lado do hiperplano.

Daí, podemos pensar no hiperplano como um divisor de um espaço $p-$dimensional em duas metades. Pode-se facilmente determinar de que lado do hiperplano um ponto se encontra ao calcular o sinal do lado esquerdo de (9.2). Um hiperplano em um espaço bidimensional é mostrado na figura 9.1.

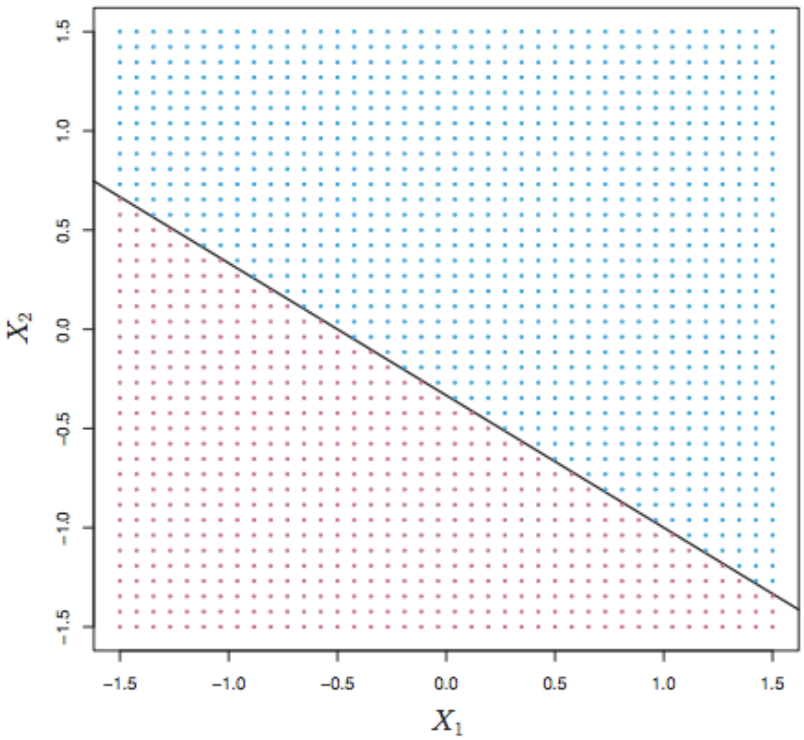

FIGURA 9.1.
O hiperplano $1 + 2X_1 + 3X_2$ *é mostrado. A região azul é o conjunto de pontos para qual* $1 + 2X_1 + 3X_2 > 0$*, e a região roxa é o conjunto de pontos para qual* $2X_1 + 3X_2 < 0$.

9.1.2. Classificação utilizando um hiperplano separador

Agora suponha que temos uma matriz $n \times p$ de dados X que consiste de n observações de treino em um espaço $p-$dimensional,

$$x_1 = \begin{pmatrix} x_{11} \\ \vdots \\ x_{1p} \end{pmatrix}, \ldots, x_n = \begin{pmatrix} x_{n1} \\ \vdots \\ x_{np} \end{pmatrix}, \qquad (9.5)$$

e que essas observações caem em duas classes - isto é, $y_1, \ldots, y_n \in$ {-1,1} onde -1 representa uma classe e 1 a outra classe. Também temos uma observação de teste, um $p-$vetor de características observadas $x^* = (x_1^* \ldots x_p^*)^T$. Nosso objetivo é desenvolver um classificador baseado nos dados de treino que irão classificar corretamente a observação de teste utilizando suas medidas características. Vimos um número de abordagens para esta tarefa, tal como análise discriminante linear e regressão logística no Capítulo 4, e árvores de classificação, bagging e boosting no Capítulo 8. Agora, veremos uma nova abordagem que é baseada no conceito de um $hiperplano separador$.

Suponha que é possível construir um hiperplano que separa as observações de treino perfeitamente de acordo com as suas legendas de classe. Exemplos de tais $hiperplanos separadores$ são mostrados no painel à esquerda da Figura 9.2. Podemos rotular as observações da classe azul como $y_i = 1$ e as da classe roxa como $y_i = -1$. Então, um hiperplano separador possui a propriedade que

$$\beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_p X_{ip} > 0 \; se \; y_i = 1 \qquad (9.6)$$

e

$$\beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_p X_{ip} < 0 \; se \; y_i = -1 \qquad (9.7)$$

Equivalentemente, um hiperplano separador possui a propriedade de que

$$y_i(\beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_p X_{ip}) > 0 \; se \; y_i = 1 \tag{9.8}$$

para todo $i = 1, \ldots, n$.

Se um hiperplano separador existe, podemos utilizá-lo para construir um classificador muito natural: uma observação de teste é atribuída a uma classe dependendo de qual lado do hiperplano está localizado. O painel direito da figura 9.2 mostra um exemplo de tal classificador. Isto é, classificamos a observação de teste x^* beaseado no sinal de $f(x^*) = \beta_0 + \beta_1 x_1^* + \beta_2 x_2^* + ... + \beta_p x_p^*$. Se $f(x^*)$ for positiva, então atribuímos a observação de teste para a classe 1, e se $f(x^*)$ for negativa, então a atribuímos para a classe -1. Podemos também fazer uso da $magnitude$ de $f(x^*)$. Se $f(x^*)$ estiver longe de zero, isto significa que x^* encontra-se longe do hiperplano, então podemos ter confiança sobre nossa atribuição de classe para x^*. Por outro lado, se $f(x^*)$ estiver mais próximo de zero, então x^* é localizado próximo ao hiperplano, e então ficamos menos certos sobre a atribuição de classe para x^*. Como vimos na Figura 9.2, um classificador que é baseado em um hiperplano separador leva a uma fronteira de decisão linear.

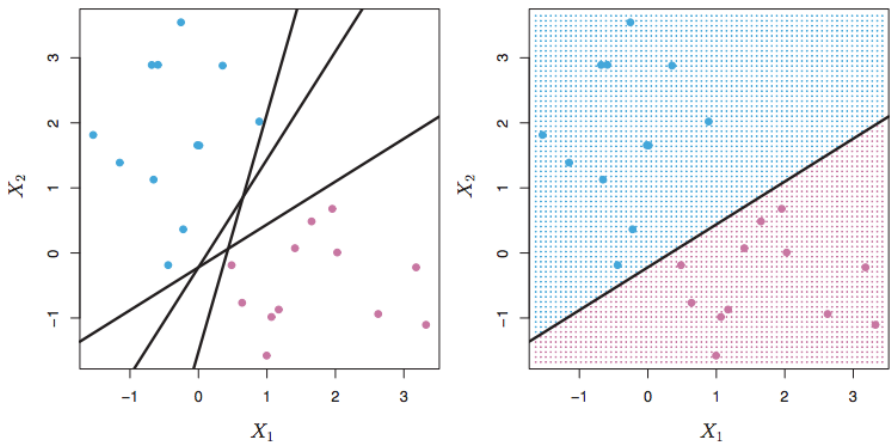

FIGURA 9.2.
Esquerda: *Há duas classes de observações, mostradas em azul e em roxo, cada qual possuindo medidas em duas variáveis. Três hiperplanos separadores (uns dos muitos possíveis), são mostrados em preto.* Direita: *Um hiperplano separador é mostrado em preto. A grade azul e roxa indica a regra de decisão feita por um classificador baseado neste hiperplano separador: uma observação de teste que cai na porção azul da grade será atribuída à classe azul, e uma observação de teste que cai na porção roxa da grade será atribuída à classe roxa.*

9.1.3. O classificador de margem máxima

Em geral, se nossos dados podem ser perfeitamente separados utilizando um hiperplano, então haverá de fato um número infinito de tais hiperplanos. Isto ocorre porque, dado um hiperplano separador, este pode ser geralmente deslocado um pouco para cima ou para baixo ou rotacionado, sem entrar em contato com nenhuma das observações. Três hiperplanos separadores possíveis estão mostrados no painel à esquerda da Figura 9.2. Para construir um classificador baseado em um hiperplano separador, devemos ter uma forma razoável para decidir quais dos infinitos hiperplanos separadores possíveis usaremos.

Uma escolha natural é o hiperplano de margem máxima (também conhecido como o hiperplano de separação ótima), o qual é o hiperplano separador mais distante das observações de treino. Isto é, podemos computar a distância (perpendicular) de cada observação de treino para um dado hiperplano separador; o menor cuja distância é a mínima das observações para o hiperplano, e é chamada de *margem*. O hiperplano de margem máxima é o hiperplano separador para o qual a margem é a maior - isto é, é o hiperplano que possui uma distância mínima mais distante das observações de treino. Podemos então classificar uma observação de teste baseada em qual lado do hiperplano de margem máxima se encontra. Isto é conhecido como *classificador de margem máxima*. Esperamos que um classificador que tenha margem grande nos dados de treino terá também uma margem grande nos dados de teste, e, portanto, classificará as observações de teste corretamente. Apesar do classificador de margem máxima ser frequentemente bem-sucedido, pode também levar a um sobreajuste quando *p* é grande.

Se $\beta_0, \beta_1,...,\beta_p$ são os coeficientes do hiperplano de margem máxima, então o classificador de margem máxima classifica a observação de teste x^* baseado no sinal de $f(x^*) = \beta_0 + \beta_1 x_1^* + \beta_2 x_2^* + ... + \beta_p x_p^*$.

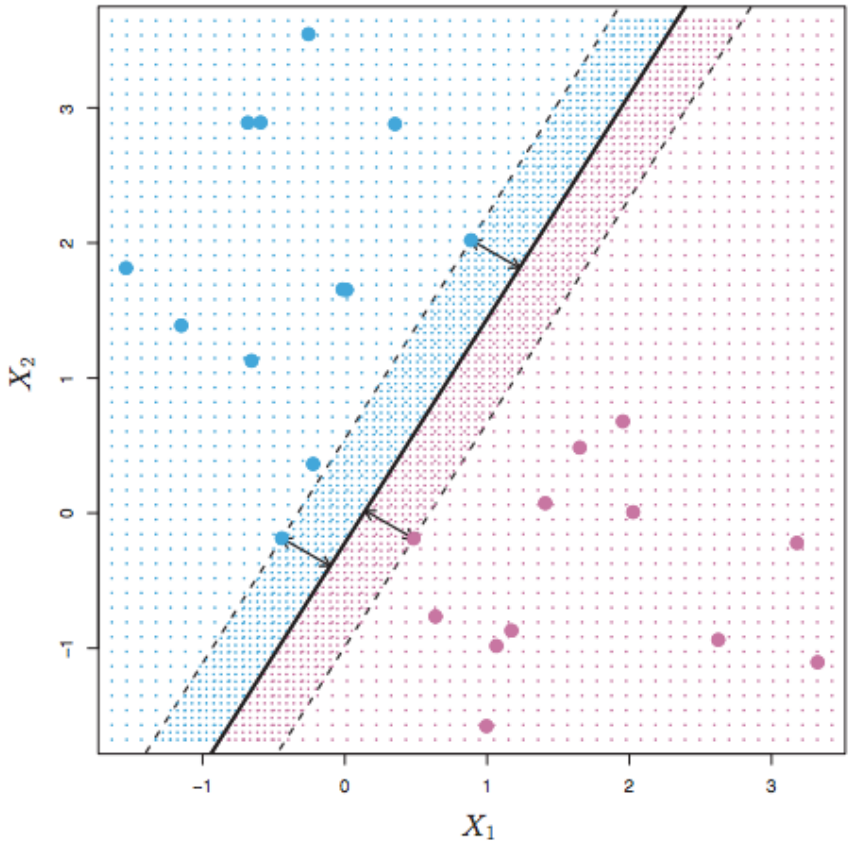

FIGURA 9.3.
Há duas classes de observações, mostradas em azul e em roxo. O hiperplano de margem máxima é mostrado como uma linha sólida. A margem é a distância da linha sólida para uma das linhas tracejadas. Os dois pontos azuis e o ponto roxo que localizam-se nas linhas tracejadas são os vetores de suporte, e a distância destes pontos ao hiperplano está indicada por setas. A grade roxa e azul indicam a regra de decisão feita por um classificador baseado neste hiperplano separador.

A Figura 9.3 mostra o hiperplano de margem máxima no conjunto de dados da Figura 9.2. Comparando o painel á direita da Figura 9.2 à figura 9.3, vemos que o hiperplano de margem máxima mostrado em (9.3) de fato resulta em uma distância mínima maior entre as observações e o hiperplano separador - isto é, uma margem maior. De certa forma, o hiperplano de margem máxima representa a linha do meio da maior "placa" que podemos inserir entre as duas classes. Examinando a Figura 9.3, vemos que três observações de teste estão equidistantes do hiperplano de margem máxima e se encontram ao longo das linhas tracejadas, indicando a largura da margem. Estas três observações são conhecidas como *vetores de suporte*, já que são vetores em um espaço $p-$dimensional (na Figura 9.3, p = 2), e eles "suportam" o hiperplano de margem máxima no sentido de que, se estes pontos fossem um pouco movidos, o hiperplano de margem máxima também se moveriam. O interessante é que o hiperplano de margem máxima dependem diretamente dos vetores de suporte, mas não das outras observações: um movimento para qualquer uma das observações não afetaria o hiperplano separador, desde que o movimento da observação não cruze a fronteira colocada pela margem. O fato de que o hiperplano de margem máxima depende diretamente de apenas um pequeno subconjunto das observações é uma propriedade importante quando discutirmos o classificador de suporte e as máquinas de vetores de suporte.

9.1.4 Construção do Classificador de Margem Máxima

Agora consideramos a tarefa de construir o hiperplano de margem máxima baseado em um conjunto de n observações de treino $x_1, \ldots, x_n \in \mathbb{R}^p$ e legendas de classe associadas $y_1, \ldots, y_n \in$ {-1,1}. Resumidamente, o hiperplano de margem máxima é a solução para o problema de otimização

$$\underset{\beta_0,\beta_1,\ldots,\beta_p,M}{\text{maximize}} \; M \tag{9.9}$$

$$\text{subject to } \sum_{j=1}^{p} \beta_j^2 = 1, \tag{9.10}$$

$$y_i(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \ldots + \beta_p x_{ip}) \geq M \;\; \forall\, i = 1, \ldots, n. \tag{9.11}$$

Este problema de otimização (9.9) - (9.11) é mais simples do que parece. Primeiramente, a restrição em (9.11) garante que cada observação estará no lado correto do hiperplano, dado que M é positivo. (Na verdade, para que cada observação esteja no lado correto do hiperplano, precisaríamos apenas que $y_i(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \ldots + \beta_p x_{ip}) > 0$. Logo, a restrição em (9.11) de fato erquer que cada observação esteja no lado correto do hiperplano, com algum amortecedor, visto que M é positivo.)

Em seguida, note que (9.10) não é realmente uma restrição no hiperplano, visto que se $\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \ldots + \beta_p x_{ip} = 0$ define um hiperplano, então $k(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \ldots + \beta_p x_{ip}) = 0$ para qualquer $k \neq 0$. No entanto, (9.10) adiciona significado à (9.11); pode-se mostrar que, com essa restrição, a distância perpendicular da i-ésima observação ao hiperplano é dada por

$$y_i(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \ldots + \beta_p x_{ip})$$

Portanto, as restrições (9.10) e (9.11) asseguram que cada observação está no lado correto do hiperplano e a pelo menos uma distância M do hiperplano. Então, M representa a margem do nosso hiperplano, e o problema de otimização escolhe $\beta_0, \beta_1, ..., \beta_p$ para maximizar M. Isto é exatamente a definição de hiperplano de margem máxima!

9.1.5. O caso não-separável

O classificador de margem máxima é uma forma bastante natural de realizar classificação, **se um hiperplano separador existir**. No entanto, como poderíamos imaginar, em muitos casos não existe hiperplano separador, e então não há classificador de margem máxima. Neste caso, o problema de otimização (9.9)-(9.11) não possui solução com $M > 0$. Um exemplo é mostrado na Figura 9.4. Neste caso, não podemos separar *exatamente* as duas classes. No entanto, como veremos na próxima seção, podemos extender o conceito de um hiperplano separador com o objetivo de desenvolver um hiperplano que *quase* separa as classes, utilizando uma chamada *margem suave*. A generalização do classificador de margem máxima ao caso não-separável é conhecida como *classificador de vetor de suporte*.

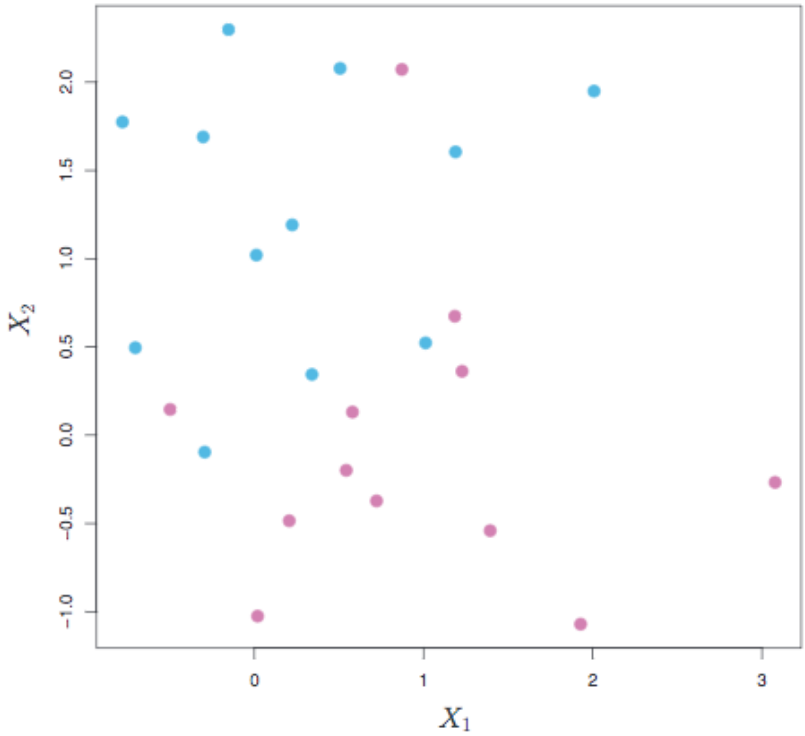

FIGURA 9.4.
Há duas classes de observações, mostradas em azul e roxo. Neste caso, as duas classes não são separáveis por um hiperplano, e então o classificador de margem máxima não pode ser utilizado.

9.2 Classificadores de vetores de suporte

9.2.1 Visão geral do classificador de vetor de suporte

Na Figura 9.4, vemos que observações que pertencem a duas classes não são necessariamente separáveis por um hiperplano. De fato, mesmo se um hiperplano separador existir, haverá instâncias em que um classificador baseado em um hiperplano separador talvez possa não ser desejável. Um classificador baseado em um hiperplano separador necessariamente classificará todas as observações de treino; isto pode resultar em sensibilidade à observações individuais. Um exemplo é mostrado na Figura 9.5. A adição de uma única observação no painel à direita da Figura 9.5 leva a uma mudança dramática no hiperplano de margem máxima. O hiperplano de margem máxima não é satisfatório - por exemplo, possui uma margem muito pequena. Isto é problemático, pois como discutido é anteriormente, a distância de uma observação do hiperplano pode ser vista como uma medida da nossa confiança de que a observação foi corretamente classificada. Além disso, o fato de que o hiperplano de margem máxima é extremamente sensível a uma mudança em uma única observação sugere que ele pode sobreajustar os dados de treino.

Neste caso, podemos desejar considerar um classificador baseado em um hiperplano que não separa perfeitamente as duas classes, podendo estar interessados em:

• Observações individuais mais robustas

• Melhor classificação da *maioria* das observações de treino.

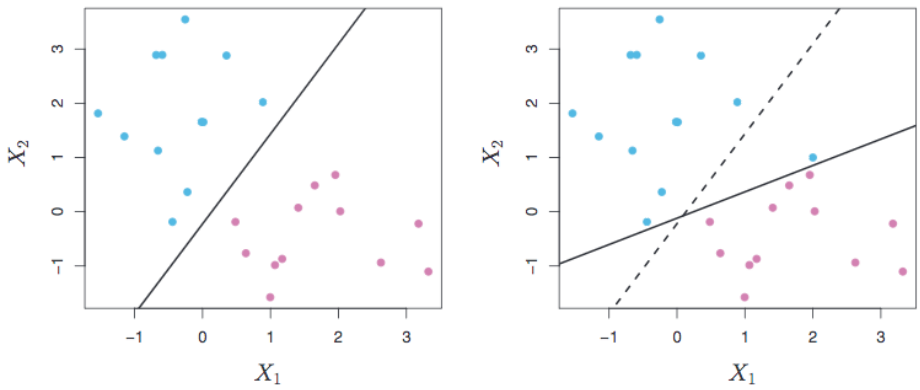

FIGURA 9.5.
À esquerda: *Duas classes de observações são mostradas em azul e em roxo, juntamente com o hiperplano de margem máxima.* À direita: *Uma observação azul foi adicionada, levando a um deslocamento dramático no hiperplano de margem máxima, mostrado como uma linha sólida. A linha tracejada indica o hiperplano de margem máxima que foi obtido na ausência deste ponto adicional.*

Isto é, pode valer a pena classificar erroneamente algumas observações de treino para que seja feito um melhor trabalho na classificação das observações restantes.

O *classificador de vetor de suporte*, às vezes chamado, de *classificador de margem suave*, faz exatamente isso. Em vez de procurar a maior margem possível tal que toda observação não está apenas no lado correto do hiperplano, mas também está no lado correto da margem, podemos permitir que algumas observações estejam no lado incorreto da margem, ou até mesmo no lado incorreto do plano. (A margem é *suave*, pois pode ser violada por algumas das observações de treino). Um exemplo é mostrado no painel à esquerda da Figura 9.6. A maioria das observações estão no lado correto da margem. No entanto, um pequeno subconjunto das observações estão no lado errado da margem.

Uma observação pode estar ao mesmo tempo no lado errado da margem e no lado errado do hiperplano. De fato, quando não há hiperplano separador, tal situação é inevitável. Observações no lado errado do hiperplano correspondem às observações de treino que são classificadas erroneamente pelo classificador de vetor de suporte. O painel à direita da Figura 9.6 ilustra tal cenário.

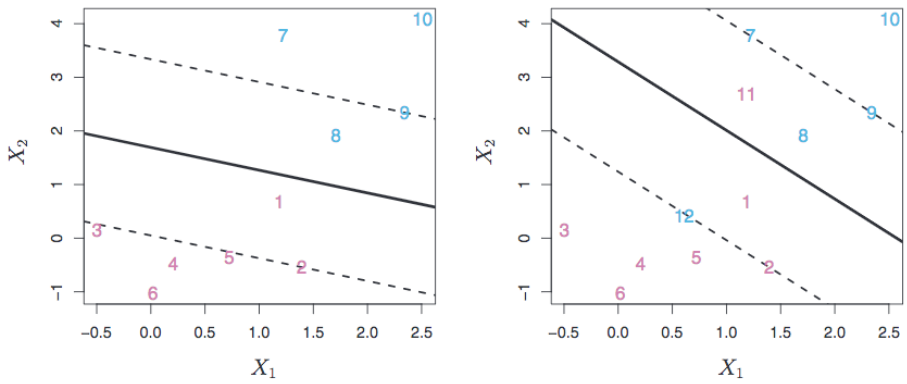

FIGURA 9.6.
À esquerda: *Um classificador de vetor de suporte foi ajustado a um pequeno conjunto de dados. O hiperplano é mostrado como uma linha sólida e as margens são mostradas como linhas tracejadas.* Observações roxas: *As observações 3,4,5 e 6 estão no lado correto da margem, a observação 2 está em cima da margem, e a observação 1 está no lado errado da margem.* Observações azuis: *As observações 7 e 10 estão no lado correto da margem, a observação 9 está em cima da margem, e a observação 8 está no lado errado da margem. Não há observações no lado errado do hiperplano.* À direita: *É o mesmo painel à esquerda com dois pontos adicionais, 11 e 12. Estas duas observações estão no lado errado do hiperplano e no lado errado da margem.*

9.2.2 Detalhes do classificador de vetor de suporte

O classificador de vetor de suporte classifica uma observação de teste dependendo de qual lado de um hiperplano ela se encontra. O hiperplano é escolhido para separar corretamente a maioria das observações de treino entre as duas classes, mas pode classificar erroneamente algumas observações. É a solução para o problema de otimização

$$\underset{\beta_0,\beta_1,\ldots,\beta_p,\epsilon_1,\ldots,\epsilon_n,M}{\text{maximize}} \; M \tag{9.12}$$

$$\text{subject to } \sum_{j=1}^{p} \beta_j^2 = 1, \tag{9.13}$$

$$y_i(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \ldots + \beta_p x_{ip}) \geq M(1 - \epsilon_i), \tag{9.14}$$

$$\epsilon_i \geq 0, \; \sum_{i=1}^{n} \epsilon_i \leq C, \tag{9.15}$$

onde C é um parâmetro de afinação não-negativo. Como em (9.11), M é a largura da margem; procuramos fazer que este valor seja o maior possível. Em (9.14), ε_1,...,ε_n são *variáveis de folga* que permitem que observações individuais estejam no lado errado da margem ou do hiperplano. Uma vez que resolvemos (9.12)-(9.15), classificamos uma observação de teste x^* como antes, simplesmente determinando em qual lado do hiperplano ela se encontra. Isto é, classificamos a observação de teste baseada no sinal de $f(x^*) = \beta_0 + \beta_1 x_1^* + \beta_2 x_2^* + ... + \beta_p x_p^*$.

O problema (9.12)-(9.15) parece complexo, mas insights em relação ao seu comportamento podem ser feitos por meio de uma séries de simples observações apresentadas abaixo. Primeiramente, a variável de folga ε_i nos diz onde a i-ésima observação é localizada, relativo ao hiperplano e relativo à margem. Se $\varepsilon_i = 0$ então a i-ésima observação está no lado correto da margem. Se $\varepsilon_i > 0$ então a i-ésima observação está no lado errado da margem, e dizemos que a i-ésima observação violou a margem. Se $\varepsilon_i > 1$ então ela está no lado errado do hiperplano.

Agora, consideramos o trabalho do parâmetro de afinação C. Em (9.15), C é a soma dos ε_i's, determinando o número e a gravidade das violações à margem (e ao hiperplano) que nós toleraremos. Podemos pensar em C como um orçamento para o quanto a margem pode ser violada pelas n observações. Se C = 0 então não há orçamento para violações à margem, e deve ser o caso em que ε_1 = ... = ε_n = 0. Neste caso, (9.12)-(9.15) simplesmente se torna o problema de otimização do hiperplano de margem máxima (9.9)-(9.11). (Obviamente, um hiperplano de margem maximal existe apenas se as duas classes são separáveis.) Para C > 0 não mais que C observações podem estar no lado errado do hiperplano, pois senão $\Sigma\varepsilon_i \leq C$. À medida que o orçamento C aumenta, nos tornamos mais tolerantes às violações na margem, e com isso a margem se alargará. Analogamente, à medida que C diminui, nos tornamos menos tolerantes às violações à margem e então a margem se estreita. Um exemplo é mostrado na Figura 9.7.

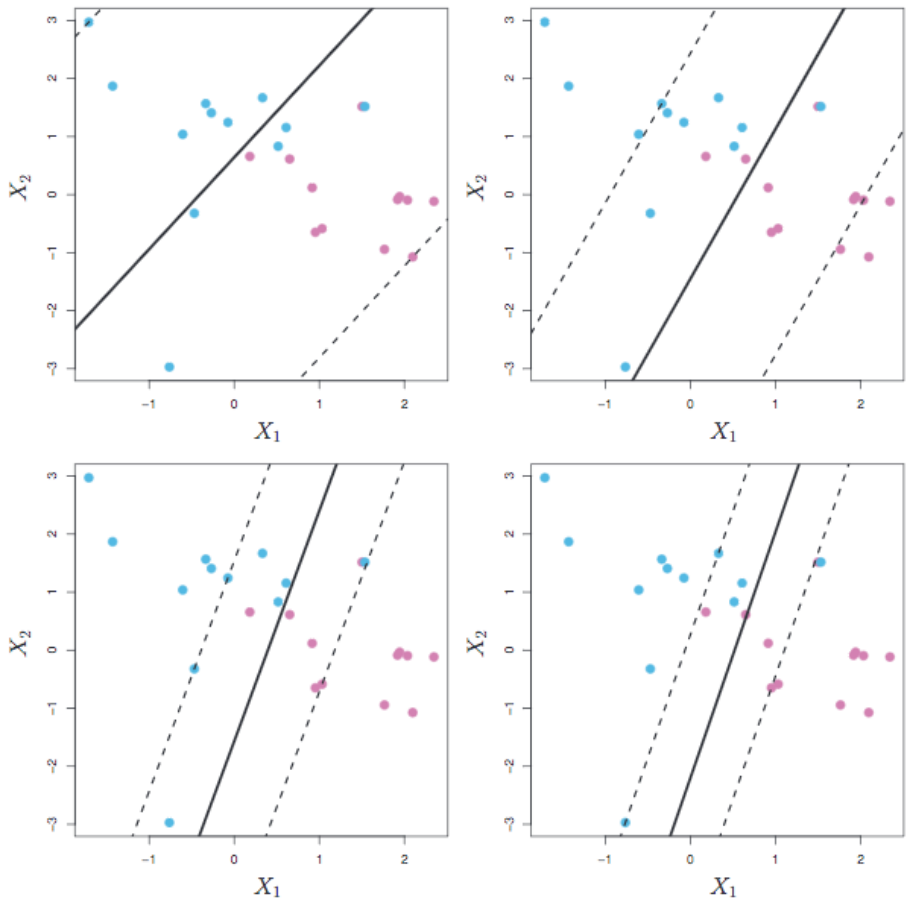

FIGURA 9.7.
Um classificador de vetor de suporte foi ajustado utilizando quatro valores diferentes do valor de afinação C em (9.12-9.15). O maior valor de C foi utilizado no painel de cima da esquerda, e valores menores foram utilizados nos outros painéis. Quando C é grande, há uma grande tolerância para observações que estão do lado errado da margem, e então a margem será grande. À medida que C diminui, a tolerância para observações que estão do lado errado da imagem diminui, e então a margem se estreita.

Na prática, C é tratado como um parâmetro de afinação que é geralmente escolhido por meio de validação cruzada. C controla o trade-off entre variância e polarização da técnica de aprendizado estatístico. Quando C é pequeno, procuramos margens estreiras que são raramente violadas; isto leva a um classificador que é altamente ajustado aos dados, o qual pode haver baixa polarização, mas alta variância. Por outro lado, quando C é maior, a margem é mais larga e permite maiores violações; isto leva à um ajuste menor e a obtenção de um classificador que possui potencialmente maior polarização e menor variância.

O problema de otimização (9.12)-(9.15) possui uma propriedade muito interessante: apenas observações que caem em cima da margem ou a violam afetarão o hiperplano, e consequentemente o classificador obtido. Em outras palavras, uma observação que se encontra estritamente no lado correto da margem não afeta o classificador de vetor de suporte. Mudar a posição desta observação não mudaria o classificador, visto que sua posição continuaria no lado correto da margem. Observações que caem diretamente em cima da margem, ou no lado errado da margem para sua classe, são conhecidos como *vetores de suporte*. Estas observações afetam de fato o classificador de vetor de suporte.

O fato de que apenas vetores de suporte afetam o classificador está em concordância com a afirmativa anterior de que C controla o trade-off entre variância e polarização do classificador de vetor de suporte. Quando o parâmetro de afinação C é grande, então a margem é larga, muitas observações violam a margem, e então há muitos vetores de suporte. Neste caso, muitas observações estão envolvidas em determinar o hiperplano. O painel de cima à esquerda possui baixa variância (já que muitas observações são vetores de suporte) mas potencialmente alta polarização. Em contraste, se C for pequeno, então haverá menos vetores de suporte e então o classificador resultante terá baixa polarização mas alta variância. O painel de cima à esquerda ilustra esta situação, com apenas oito vetores de suporte.

O fato da regra de decisão do classificador de vetor de suporte ser baseada apenas em potencialmente um pequeno conjunto das observações de treino (os vetores de suporte) significa que isto é bem robusto para o comportamento de observações que estão longe do hiperplano. Esta propriedade é distinta de algum dos outros métodos de classificação que vimos anteriormente, tal como análise discriminante linear.

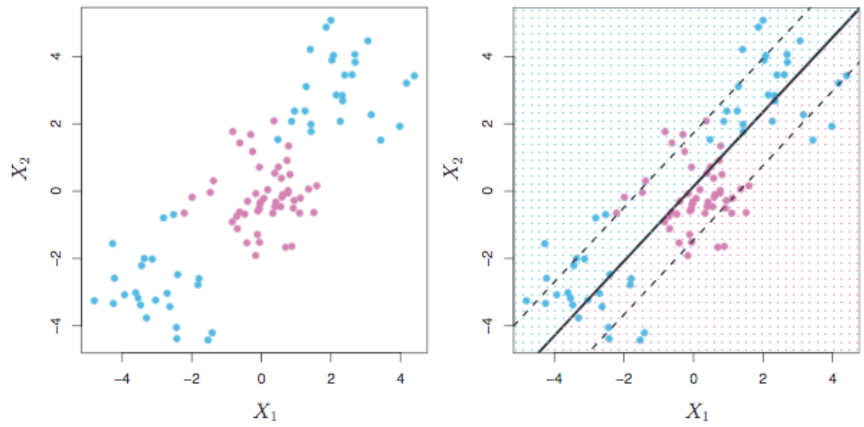

FIGURA 9.8.
À esquerda: *As observações caem em duas classes, com uma fronteira não-linear entre elas.* À direita: *O classificador de vetor de suporte procura uma fronteira linear, e consequentemente tem uma performance ruim.*

9.3 Máquinas de vetores de suporte

Vamos primeiramente classificar um mecanismo geral para converter um classificador linear em um que produz fronteiras de decisão não-lineares. Então, introduziremos a máquina de vetor de suporte, que faz isto de forma automática.

9.3.1 Classificação com fronteiras de decisão não-lineares

O classificador de vetor de suporte é uma abordagem natural para classificação no contexto de duas classes, se a fronteira entre as duas classes for linear. No entanto, na prática podemos nos deparar com fronteiras de classe não-lineares. Poor exemplo, considere os dados no painel à esquerda da Figura 9.8. É claro que um classificador de vetor de suporte ou qualquer outro classificador linear terá um desempenho ruim. De fato, o classificador de vetor de suporte mostrado no painel à direita na Figura 9.8 é inútil neste caso.

No Capítulo 7, nos deparamos com uma situação análoga. Vimos que a performance da regressão linear pode ser afetada quando há uma relação não-linear entre os preditores e a saída. Neste caso, consideramos aumentar o espaço de recurso utilizando funções dos preditores, como termos quadráticos e cúbicos, de modo a incluir esta não-linearidade.

No caso do classificador de vetor de suporte, podemos resolver o problema de possíveis fronteiras não-lineares entre classes de uma forma similar, ao aumentar o espaço de recurso utilizando funções polinomiais dos preditores. Por exemplo, em vez de ajustar um classificador de vetor de suporte utilizando p recursos

$$X_1, X_2, \ldots, X_p,$$

Então (9.12)-(9.15) se tornaria

$$\underset{\beta_0,\beta_{11},\beta_{12}\ldots,\beta_{p1},\beta_{p2},\epsilon_1,\ldots,\epsilon_n,M}{\text{maximize}} \; M \tag{9.16}$$

$$\text{subject to } y_i \left(\beta_0 + \sum_{j=1}^{p} \beta_{j1} x_{ij} + \sum_{j=1}^{p} \beta_{j2} x_{ij}^2 \right) \geq M(1-\epsilon_i),$$

$$\sum_{i=1}^{n} \epsilon_i \leq C, \;\; \epsilon_i \geq 0, \;\; \sum_{j=1}^{p} \sum_{k=1}^{2} \beta_{jk}^2 = 1.$$

Por que isto nos leva a uma fronteira de decisão não-linear? No espaço de recurso mais largo, a fronteira de decisão que resulta de (9.16) é de fato linear. Mas, no espaço de recurso original, a fronteira de decisão é da forma $q(x) = 0$, onde q é um polinômio quadrático, e suas soluções são geralmente não-lineares. Pode-se, além disso, deixar o espaço de recurso mais largo com termos polinomiais de ordens mais altas, ou termos de interação da forma $X_j X_{j'}$, para $j \neq j'$.

Além disso, outras funções dos preditores além das polinomiais podem ser consideradas. É facil ver que há muitas formas possíveis de tornar o espaço de recurso mais largo, e ao menos que sejamos cuidadosos, podemos nos deparar com um número enorme de recursos. Daí, as computações se tornariam impossíveis de ser manejadas. A máquina de vetor de suporte, que apresentaremos em seguida, nos permite aumentar o espaço de recurso utilizado pelo classificador de vetor de suporte de forma que obtemos computações eficientes.

9.3.2 A máquina de vetor de suporte

A *máquina de vetor de suporte* (SVM) é uma extensão do classificador de vetor de suporte que resulta do "enlarguecimento" do espaço de recurso de uma forma específica, utilizando $núcleos$. A ideia principal é: queremos aumentar nosso espaço de recurso de forma a acomodar uma fronteira não-linear entre as classes. A abordagem do núcleo que descrevemos aqui é simplesmente uma abordagem computacional eficiente para realizar esta ideia.

Não foi discutido exatamente como o classificador de vetor de suporte é computado, pois os detalhes se tornam técnicos, de certa forma. No entanto, a solução para o problema do classificador de vetor de suporte (9.12)-(9.15) envolve apenas os *produtos internos* das observações. O produto interno dos dois r-vetores a e b é definido como $< a, b >$ = $\Sigma a_i b_i$. Daí, o produto interno das duas observações x_i,$x_{i'}$ é dado por

$$\langle x_i, x_{i'} \rangle = \sum_{j=1}^{p} x_{ij} x_{i'j}. \qquad (9.17)$$

Pode ser mostrado que

> O classificador de vetor de suporte linear pode ser representado como:

$$f(x) = \beta_0 + \sum_{i=1}^{n} \alpha_i \langle x, x_i \rangle, \qquad (9.18)$$

onde há n parâmetros α_i, $i = 1, \ldots, n$, um por observação de treino.

> Para estimar os parâmetros α_1,...,α_n e β_0, precisamos dos $\binom{n}{2}$ produtos internos $< x_i, x_{i'} >$ entre todos os pares de observações de treino.

Note que em (9.18), para que seja avaliada a função $f(x)$, precisamos computar o produto interno entre o novo ponto x e cada um dos pontos de treino x_i. No entanto, α_1 é não-nulo apenas para os vetores de suporte na solução - isto é, se uma observação de treino não for um vetor de suporte, então seu α_i é igual a zero. Então, se S é a coleção de índices destes pontos de suporte, podemos reescrever qualquer função solução da forma (9.18) como

$$f(x) = \beta_0 + \sum_{i \in S} \alpha_i \langle x, x_i \rangle, \qquad (9.19)$$

que tipicamente envolve menos termos que em (9.18).

Resumindo, ao representar o classificador linear $f(x)$, e ao computar seus coeficientes, só precisamos dos produtos internos.

Agora suponha que toda vez que o produto interno (9.17) apareça na representação (9.18), ou em um cálculo da solução do classificador de vetor de suporte, o substituímos com uma *generalização* do produto interno da forma

$$K(x_i, x_{i'}), \qquad (9.20)$$

onde K é alguma função que nos referiremos a ela como um *núcleo*. Um núcleo é uma função que quantifica a similaridade de duas observações. Por exemplo, podemos simplesmente utilizar

$$K(x_i, x_{i'}) = \sum_{j=1}^{p} x_{ij} x_{i'j}, \qquad (9.21)$$

que nos devolveria o classificador de vetor de suporte. A equação 9.21 é conhecida como um núcleo *linear*, pois o classificador de vetor de suporte é linear nos recursos; o núcleo linear essencialmente quantifica a similaridade de um par de observações utilizando correlação (padrão) de Pearson. Mas poderíamos escolher, em vez disso, outra forma para (9.20). Por exemplo, poderia-se substituir toda instância de $\Sigma x_{ij} x_{i'j}$ pela quantidade

$$K(x_i, x_{i'}) = (1 + \sum_{j=1}^{p} x_{ij} x_{i'j})^d. \qquad (9.22)$$

Isto é conhecido como um *núcleo polinomial* de grau d, onde d é um inteiro positivo. Utilizar tal núcleo com $d > 1$, em vez do núcleo linear padrão (9.21), algoritmo de classificador de vetor de suporte leva a uma fronteira de decisão bem mais flexível. Essencialmente, ele leva ao ajuste de um classificador de vetor de suporte em um espaço com maior dimensão, envolvendo polinômios de grau d, em vez do espaço de recurso original. Quando o classificador de vetor de suporte é combinado com um núcleo não-linear como (9.22), o classificador resultante é conhecido como máquina de vetor de suporte. Note que neste caso a função (não-linear) possui a forma

$$f(x) = \beta_0 + \sum_{i \in \mathcal{S}} \alpha_i K(x, x_i). \qquad (9.23)$$

O painel à esquerda da figura 9.9 mostra um exemplo de uma máquina de vetor de suporte com um núcleo polinomial aplicado aos dados não-lineares da figura 9.8. O ajuste melhora substancialmente, se comparado ao classificador de vetor de suporte linear. Quando $d = 1$, então o SVM (máquina de vetor de suporte) é reduizado ao classificador de vetor de suporte visto no começo do capítulo.

O kernel polinomial mostrado em (9.22) é um exemplo de um possível núcleo não-linear. Outra escolha popular é o *núcleo radial*, que toma a forma

$$K(x_i, x_{i'}) = \exp(-\gamma \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2). \qquad (9.24)$$

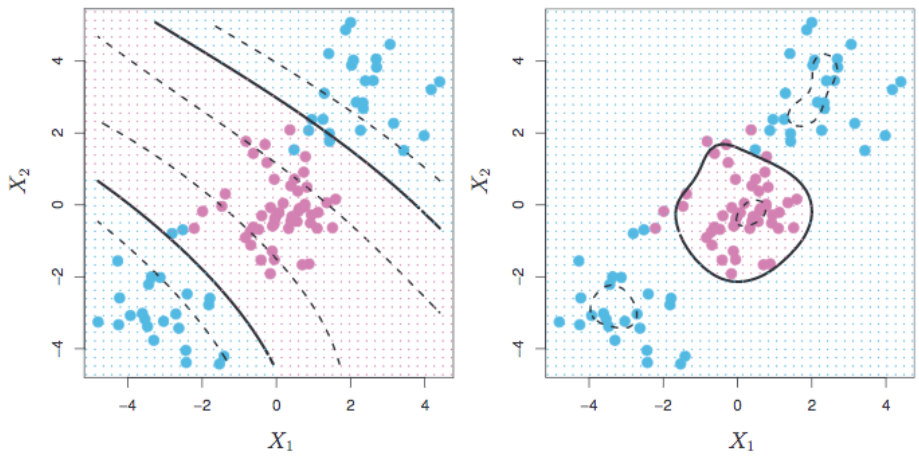

FIGURA 9.9.
À esquerda: *Um SVM com um núcleo polinomial de grau 3 é aplicado aos dados não-lineares da Figura 9.8, resultando em uma regra de decisão mais apropriada.* À direita: *Um SVM com um núcleo radial é aplicado. Neste exemplo, ambos os núcleos são capazes de captar a fronteira de decisão.*

Em (9.24), γ é uma constante positiva. O painel à direita da Figura 9.9 mostra um exemplo de um SVM com um núcleo radial nestes dados não-lineares.

Como o núcleo radial (9.24) realmente funciona? Se uma dada observação de teste $x^* = (x_1^* \ldots x_p^*)^T$ está longe da observação de treino x_i em ternos do espaço Euclidiano, então $\Sigma(x_j^* - x_{ij}^*)^2$ terá um valor alto, e então $K(x_i, x_{i'}) = exp(-\gamma\Sigma(x_j^* - x_{ij}^*)^2)$ terá um valor muito baixo. Isto significa que em (9.23) x_i virtualmente não terá importância em f^*. Relembre que a classificação da classe que foi prevista para a observação de teste x^* é baseada no sinal de $f(x^*)$. Em outras palarvas, observações de treino que estão longe de x^* não terão papel importante na classificação de classe prevista para x^*. Isto significa que o núcleo radial possui um comportamento bastante $radial$, no sentido de que apenas observações de treino próximas possuem um efeito na classificação de classe de uma observação de teste.

Qual a vantagem de utilizar um núcleo em vez de apenas aumentar o espaço de recurso utilizando funções de recursos originais, como em (9.16)? Uma vantagem é a computacional, e nos leva ao fato de que utilizando núcleos, precisamoso computar apenas $K(x_i, x_{i'})$ para todos os $\binom{n}{2}$ pares distintos i, i'. Isto é importante, pois em muitas aplicações de SVMs, o espaço de recurso aumentado é tão grande que as computações não são tratáveis. Para alguns núcleos, como o núcleo radial (9.24), o espaço de recurso é $implícito$ e dimensionalmente infinito, então não poderíamos computá-lo de qualquer forma.

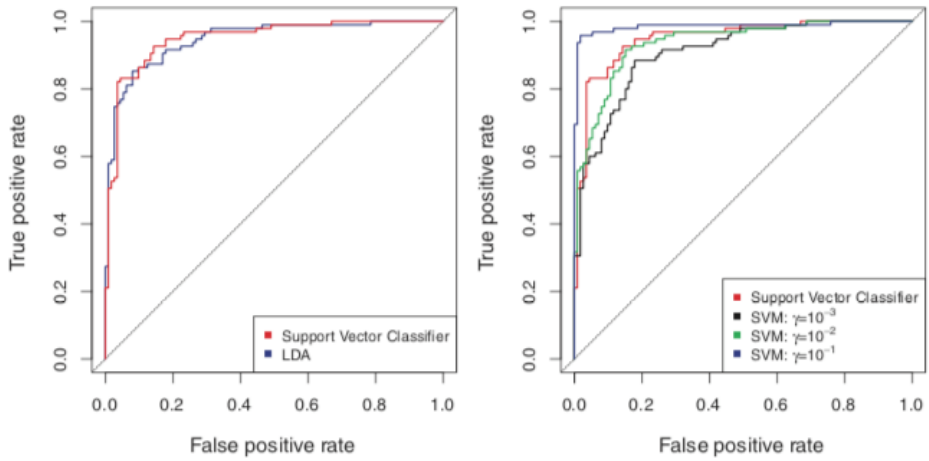

FIGURA 9.10.
Curvas ROC para o conjunto de dados de treino $Heart$. À esquerda: *O classificador de vetor de suporte e LDA são comparados.* À direita: *O classificador de vetor de suporte é comparado a um SVM utilizando um núcleo de base radial com* $\gamma = 10^{-3}, 10^{-2}, e\ 10^{-1}$.

9.3.3 Uma aplicação aos dados de Doenças Cardíacas

No Capítulo 8 aplicamos árvores de decisão e métodos relacionados aos dados $Heart$. O objetivo é utilizar 13 preditores tais como $Idade$, $Sexo$ e $Chol$ para prever se um indivíduo possui doenças cadíacas. Agora investigamos como um SVM se compara com LDA (análise linear discriminante). Após removermos 6 observações faltantes, os dados consistiam de 297 indivíduos, que dividimos aleatoriamente entre 207 de treino e 90 de teste.

Primeiramente, ajustamos LDA e o classificador de vetor de suporte aos dados de treino. Note que o classificador de vetor de suporte é equivalente a um SVM se utilizarmos um núcleo polinomial de grau d = 1. O painel à esquerda da Figura 9.10 mostra curvas ROC (descritas na Seção 4.4.3) para as predições dos dados de treino para ambos LDA e classificador de vetor de suporte. Ambos os classificadores computam escores da forma $\hat{f}(X) = \hat{\beta}_0 + \hat{\beta}_1 X_1 + \hat{\beta}_2 X_2 + \ldots + \hat{\beta}_p X_p$ para cada observação. Para qualquer atalho t, classificamos observações entre as categorias *doença cardíaca* e *sem doença cardíaca* dependendo se $\hat{f}(X) < t$ ou $\hat{f}(X) \geq t$. A curva ROC é obtida ao formar estas predições e ao computar as taxas de falso positivo e os verdadeiro positivo para um intervalo de valores de t. Um classificador ótimo alcançará a extremidade acima à esquerda do gráfico ROC. Neste exemplo ambos a LDA e o classificador de vetor de suporte possuem boa performance, apesar de haver uma sugestão de que o classificador de vetor de suporte seja um pouco superior.

O painel à direita da Figura 9.10 mostra curvas ROC para SVMs utilizando um kernel radial, com vários valores de γ. À medida que γ aumenta e o ajuste se torna menos linear, as curvas ROC melhoram. $\gamma = 10^{-1}$ parece retornar uma curva ROC quase perfeita. No entanto, estas curvas representam taxas de erro de treino, que podem ser enganosas em termos da performance nos novos dados de teste. A figura 9.11 mostra curvas ROC computadas nas 90 observações de teste. Podemos observar algumas diferenças nas curvas ROC de treino. No painel à esquerda da Figura 9.11, o classificador de vetor de suporte parece ter uma pequena vantagem sobre o LDA (apesar destas diferenças não serem significantes estatisticamente). No painel à direita, o SVM utilizando $\gamma = 10^{-1}$, que mostrou os melhores resultados nos dados de treino, produz as piores estimativas nos dados de teste. Isto mostra mais uma vez que um método mais flexível frequentemente produzirá menores taxas de erro de treino, mas isto não necessariamente leva a uma melhor performance nos dados de teste. Os SVMS com $\gamma = 10^{-2}$ e $\gamma = 10^{-3}$ possuem performances comparáveis à do classificador de vetor de suporte, e todas possuem performance melhor que o SVM com $\gamma = 10^{-1}$.

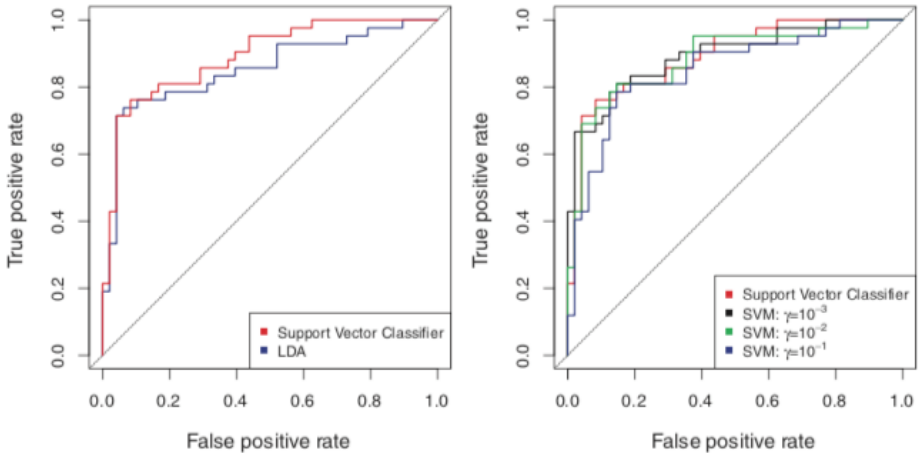

FIGURA 9.11.
Curvas ROC para o conjunto de teste para os dados $Heart$. À esquerda: *O classificador de vetor de suporte e LDA são comparados*. À direita: *O classificador de vetor de suporte é comparado a um SVM utilizando um núcleo de base radial com* $\gamma = 10^{-3}, 10^{-2}, e\ 10^{-1}$.

9.4 SVMs com mais de duas classes

Até agora, nossa discussão foi limitada ao caso da classificação binária: isto é, classificação no contexto de duas classes. Como podemos extender SVMs para um caso mais geral onde temos algum número de classes arbitrário? O conceito de hiperplanos separadores nos quais SVMs são baseadas não se extende naturalmente a mais de duas classes. Apesar de algumas propostas para a extensão de SMVs para um caso de K-classes terem sido feitas, as duas mais populares são as abordagens um-versus-um e a um-versus-todos.

9.4.1 Classificação um-versus-um

Suponha que gostaríamos de realizar classificação utilizando SVMS, e há $K > 2$ classes. Uma abordagem um-versus-um ou todos-os-pares constrói $\binom{K}{2}$ SVMs, cada qual comparando um par de classes. Por exemplo, um SVM pode comparar a k-ésima classe, codificada como +1, à k'-ésima classe, codificada como -1. Classificamos uma observação de teste utilizando cada um dos $\binom{K}{2}$ classificadores, e calculamos o número de vezes que a observação de teste é atribuída a cada uma das K classes. A classificação final é realizada ao atribuir a observação de teste à classe a qual foi mais frequentemente atribuída nestas classificações par a par $\binom{K}{2}$.

9.4.2 Classificação um-versus-todos

A abordagem um-versus-todos é um procedimento alternativo para aplicar SVMs no caso de K > 2 classes. Ajustamos K SVMs, cada vez comparando uma das K classes às K-1 classes restantes. Seja $\beta_{0k},\beta_{1k},...,\beta_{pk}$ os parâmetros que resultam do ajuste de um SVM comparando a k-ésima classe (codificada como +1) às outras restantes (codificadas como -1). Seja x^* uma observação de teste. Atribuímos a observação à classe para qual $\beta_{0k},\beta_{1k}x_1^*,...,\beta_{pk}x_p^*$ é a maior, e isto leva a um grau alto de confiança que a observação de teste pertence a k-ésima classe.

9.5 Relação com Regressão Logística

Desde a introdução das SVMs, conexões profundas entre SVMs e outros métodos estatísticos clássicos emergiram. Pode-se reescrever (9.12)-(9.15) para o ajuste de $f(X) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_p X_p$ como

$$\underset{\beta_0,\beta_1,\ldots,\beta_p}{\text{minimize}} \left\{ \sum_{i=1}^{n} \max\left[0, 1 - y_i f(x_i)\right] + \lambda \sum_{j=1}^{p} \beta_j^2 \right\}, \qquad (9.25)$$

onde λ é um parâmetro de afinação não-negativo. Quando λ é grande então $\beta_1,...,\beta_p$ são pequenas, mais violações à margem são toleradas, e um teremos um classificador com baixa variância e alta polarização. Quando λ é pequeno, então poucas violações à margem ocorrerão; isto leva a um classificador de alta variância mas baixa polarização. Então, um pequeno valor de λ em (9.25) leva a um pequeno valor de C em (9.15). O termo $\lambda\Sigma\beta_j^2$ possui um papel em controlar o trade-off entre polarização e variância.

Agora, (9.25) toma a seguinte forma de "Perda + Penalidade":

$$\underset{\beta_0,\beta_1,\ldots,\beta_p}{\text{minimize}} \{L(\mathbf{X}, \mathbf{y}, \beta) + \lambda P(\beta)\} . \qquad (9.26)$$

Em (9.26), $L(\mathbf{X},\mathbf{y},\beta)$ é alguma função de perda quantificando a extensão para qual o modelo, parametrizado por β, ajusta os dados ($\mathbf{X}$,$\mathbf{y}$), e $P(\beta)$ é uma função penalidade no vetor de parâmetro β cujo efeito é controlado por um parâmetro de afinação não-negativo λ. No caso de (9.25), a função perda toma a forma

$$L(\mathbf{X}, \mathbf{y}, \beta) = \sum_{i=1}^{n} \max [0, 1 - y_i(\beta_0 + \beta_1 x_{i1} + \ldots + \beta_p x_{ip})] .$$

Isto é conhecido como *perda de toque*, e é retratado na figura 9.12. No entanto, a função de perda de toque mostra-se relacionada de forma próxima com a função de perda utilizada na regressão logística, também mostrada na figura 9.12.

Uma caracteristicca importante do classificador de vetor de suporte é que apenas vetores de suporte possuem um papel no classificador obtido; observações no lado correto da margem não a afetam. Isto acontece devido ao fato de que a função de perda mostrada na Fgiura 9.12 é exatamente zero para observações para as quais $y_i(\beta_0 + \beta_1 X_{i1} + ... + \beta_p X_{ip}) \geq 1$; elas correspondem às observações que estão no lado correto da margem. Em contraste, a função de perda para regressão logística mostrada na Figura 9.12 não é exatamente zero em nenhum local. Mas, é muito pequena para observações que estão longe da fronteira de decisão. Devido às similaridades entre suas funções de perda, a regressão logística e o classificador de vetor de suporte frequentemente dão resultados bastante similares. Quando as classes são bem separadas, SVMs tendem a se comportar de forma melhor que a regressão logística; em regimes com sobreposição, a regressão logística é frequentemente preferida.

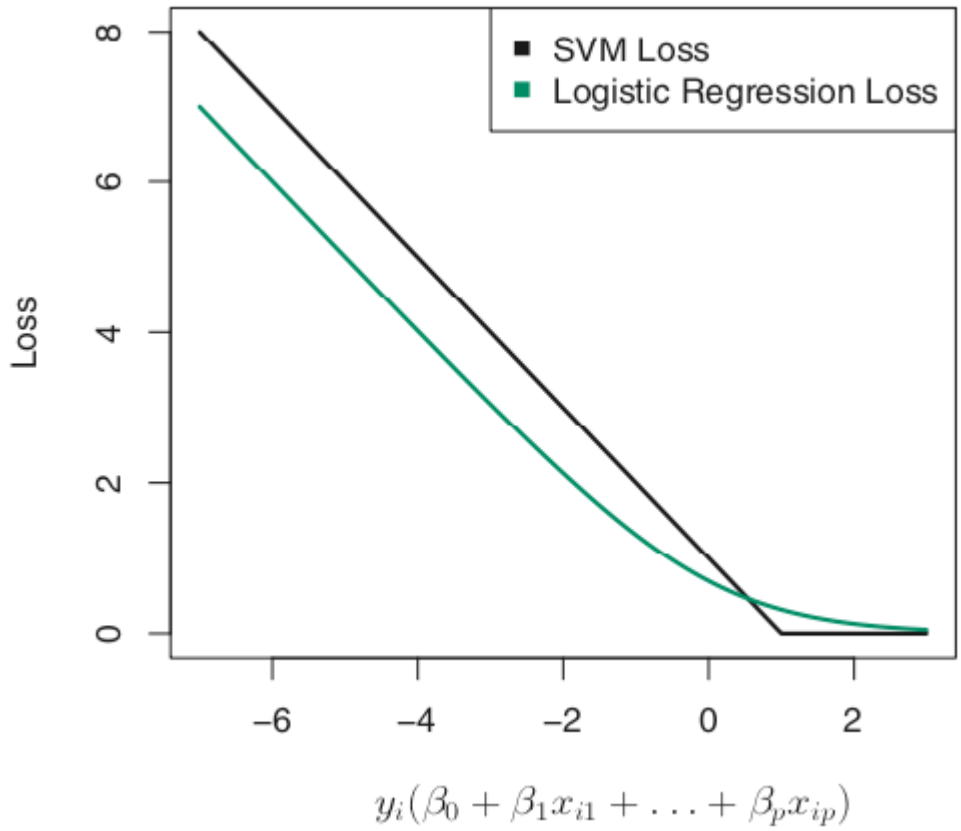

FIGURA 9.12.
As funções de perda do SVM e da regressão logística são comparadas, como função de $y_i(\beta_0 + \beta_1 X_{i1} + ... + \beta_p X_{ip})$. *Quando* $y_i(\beta_0 + \beta_1 X_{i1} + ... + \beta_p X_{ip})$ *for maior que 1, então a perda do SVM é nula, já que isto corresponde a uma observação que está no lado correto da margem. Em geral, as duas funções de perda possuem comportamentos bem similares.*

Quando o classificador de vetor de suporte e SVM foram introduzidos, foi pensando que o parâmetro de afinação C em (9.15) era um parâmetro sem importância e incômodo que poderia ser estabelecido como um valor padrão, como 1. No entanto, a formulação "Perda + Penalidade" (9.25) para o classificador de vetor de suporte indica que isto não é o caso. A escolha do parâmetro de afinação é muito importante e determina a extensão pra qual o modelo se subajusta ou sobreajusta aos dados, como mostrado na Figura 9.7.

Estabelecemos que o classificador de vetor de suporte é proximamente relacionado à regressão logística e a outros métodos estatísticos clássicos. Seria o SVM único no uso de núcleos para aumentar o espaço de recurso para que sejam acomodadas fronteiras de classe não-lineares? A resposta é **não**. Poderíamos realizar regressão logística ou muitos outros métodos de classificação utilizando núcleos não-lineares. No entanto, por razões históricas, o uso de núcleos não-lineares é melhor difundido no contexto de SVM.

Apesar de não ser mostrado aqui, há uma extensão do SVM para regressão chamada *regressão de vetor de suporte*. No Capítulo 3, vimos que a regressão por mínimos quadrados procura coeficientes $\beta_0,\beta_1,...,\beta_p$ tais que a soma do resíduos elevados ao quadrado é a menor possível. A regressão por vetor de suporte procura coeficientes que minimizam um tipo diferente de perda, onde apenas resíduos maiores em valor absoluto que alguma constante positiva contribuem para a função de perda. Esta é uma extensão da margem utilizada em classificadores de vetor de suporte para o contexto da regressão.