DATA 572: Supervised Learning

2023W2

Shan Du

Support Vector Machines

- One of the most influential approaches to supervised learning is the support vector machine (SVM). SVM was developed in the 1990s and has grown in popularity since then.
- SVMs have been shown to perform well in a variety of settings, and are often considered one the best "out of the box" classifiers.

Support Vector Machines

- The support vector machine is a generalization of a simple and intuitive classifier called the *maximal margin classifier.*
- The *maximal margin classifier* is elegant and simple, but cannot be applied to most data sets, since it requires that the classes be separable by a linear boundary.
- The *support vector classifier* is an extension of the maximal margin classifier that can be applied in a broader range of cases.

Support Vector Machines

- The *support vector machine* is a further extension of the support vector classifier in order to accommodate non-linear class boundaries.
- Support vector machines are intended for the binary classification setting in which there are two classes.
- We can extend support vector machines further to the case of more than two classes.

Support Vector Machines

- SVM is similar to logistic regression in that it is driven by a linear function $w^T x + b$.
- Unlike logistic regression, the support vector machine does not provide probabilities, but only outputs a class identity.

Maximal Margin Classifier

- The maximal margin classifier is first to find a maximal margin hyperplane (also known as the optimal separating hyperplane), and then the classifier is constructed based upon this hyperplane and an observation is classified by on which side of the hyperplane it lies.

Hyperplane

- In a p-dimensional space, a hyperplane is a flat affine subspace of dimension $p - 1$.
- For instance, in two dimensions, a hyperplane is a flat one-dimensional subspace—in other words, a line.
- In three dimensions, a hyperplane is a flat two-dimensional subspace—that is, a plane.
- In $p > 3$ dimensions, it can be hard to visualize a hyperplane, but the notion of a $(p - 1)$-dimensional flat subspace still applies.

Hyperplane

- The mathematical definition of a hyperplane is quite simple. In two dimensions, a hyperplane is defined by the equation

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

Equation of a line

for parameters β_0, β_1, and β_2.

- Any $X = (X_1, X_2)^T$ fits the equation is a point on the hyperplane.
- It is simply the equation of a line, since indeed in two dimensions a hyperplane is a line.

Hyperplane

- If we extend to the p-dimensional setting:

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

defines a p-dimensional hyperplane.

- Any $X = (X_1, X_2, \ldots, X_p)^T$ satisfies the equation is a point on the hyperplane.
- Now, suppose that X does not satisfy the equation; rather,

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

Hyperplane

- Then this tells us that X lies to one side of the hyperplane. On the other hand, if

$$\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p < 0$$

then X lies on the other side of the hyperplane.

- So we can think of the hyperplane as dividing p-dimensional space into two halves.
- One can easily determine on which side of the hyperplane a point lies by simply calculating the sign of the left-hand side of the equation.

Hyperplane

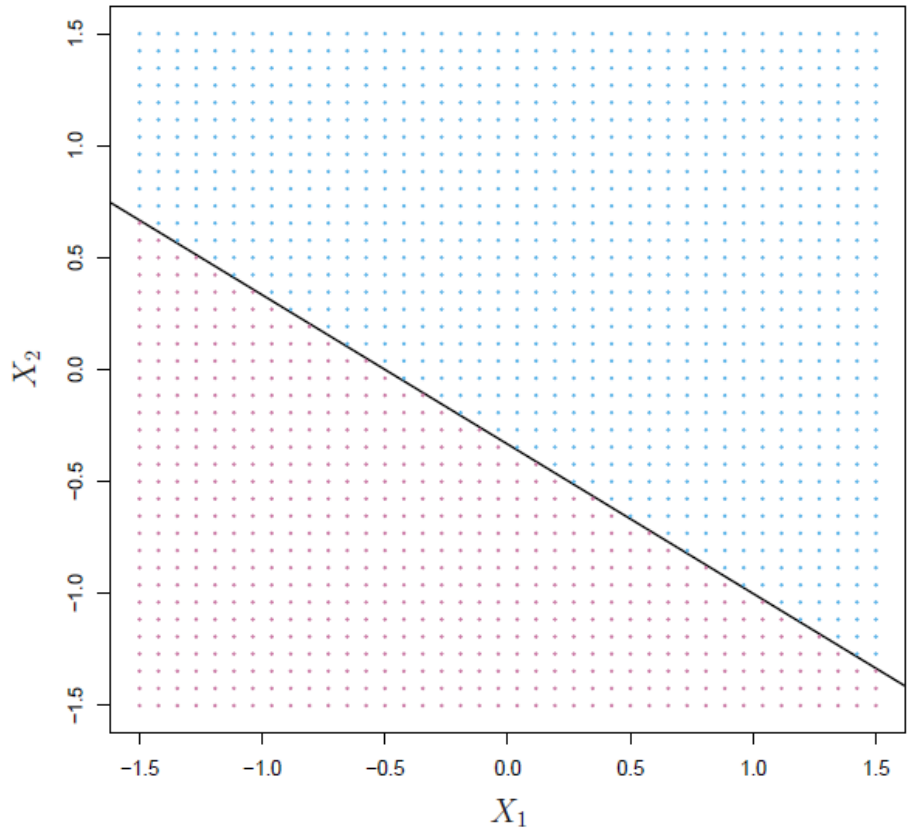

FIGURE 9.1. *The hyperplane* $1 + 2X_1 + 3X_2 = 0$ *is shown. The blue region is the set of points for which* $1 + 2X_1 + 3X_2 > 0$*, and the purple region is the set of points for which* $1 + 2X_1 + 3X_2 < 0$*.*

Classification Using a Separating Hyperplane

- Now suppose that we have an $n \times p$ data matrix $\mathbf{X}$ that consists of n training observations in p-dimensional space,

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

and that these observations fall into two classes—that is, $y_1, \ldots y_n \in \{-1, 1\}$ where -1 represents one class and 1 the other class.

Classification Using a Separating Hyperplane

- We also have a test observation, a p-vector of observed features $x^* = (x_1^*, \ldots, x_p^*)^T$. Our goal is to develop a classifier based on the training data that will correctly classify the test observation using its feature measurements.
- Suppose that it is possible to construct a hyperplane that separates the training observations perfectly according to their class labels.

Classification Using a Separating Hyperplane

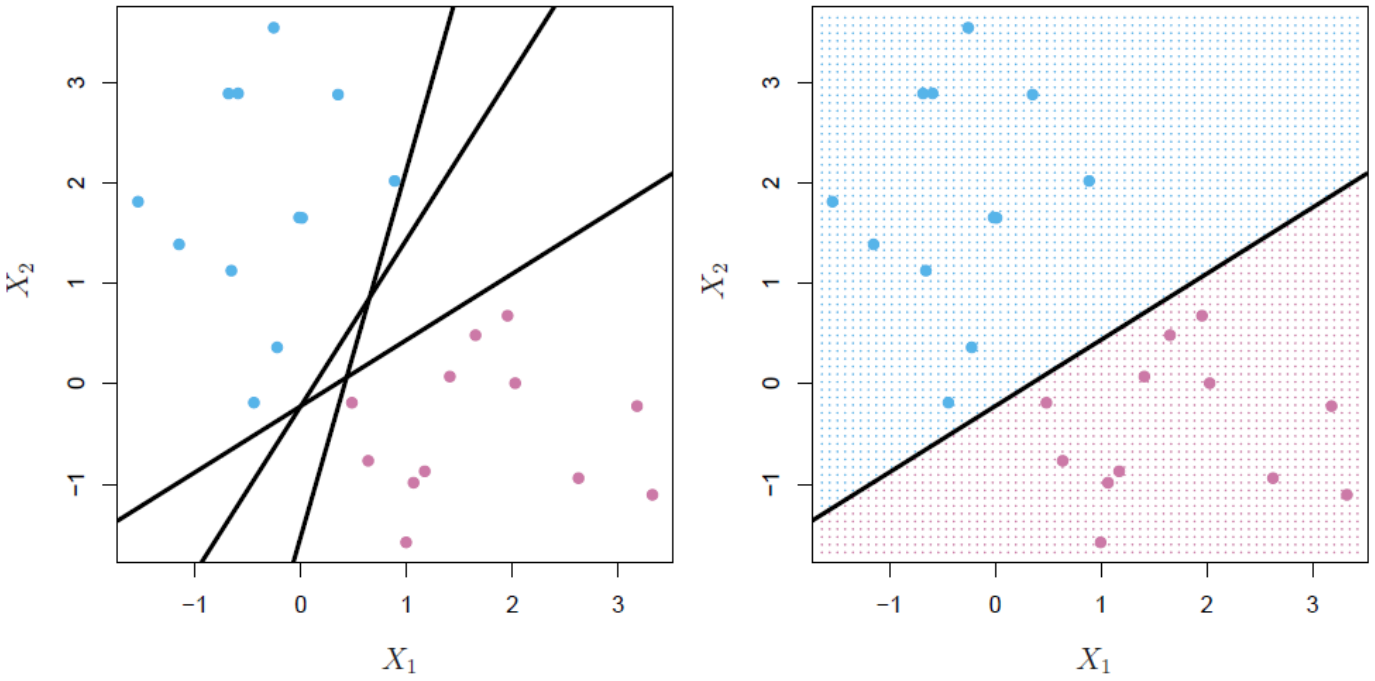

FIGURE 9.2. Left: *There are two classes of observations, shown in blue and in purple, each of which has measurements on two variables. Three separating hyperplanes, out of many possible, are shown in black.* Right: *A separating hyperplane is shown in black. The blue and purple grid indicates the decision rule made by a classifier based on this separating hyperplane: a test observation that falls in the blue portion of the grid will be assigned to the blue class, and a test observation that falls into the purple portion of the grid will be assigned to the purple class.*

Classification Using a Separating Hyperplane

- We can label the observations from the blue class as $y_i = 1$ and those from the purple class as $y_i = -1$. Then a separating hyperplane has the property that

$\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip} > 0$ if $y_i = 1$,

and

$\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip} < 0$ if $y_i = -1$

- Equivalently, a separating hyperplane has the property that

$y_i(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip}) > 0$ for all $i = 1, \ldots, n$.

Classification Using a Separating Hyperplane

- If a separating hyperplane exists, we can use it to construct a very natural classifier: a test observation is assigned a class depending on which side of the hyperplane it is located.
- We classify the test observation x^* based on the sign of $f(x^*) = \beta_0 + \beta_1 x_1^* + \beta_2 x_2^* + \cdots + \beta_p x_p^*$.
- If $f(x^*)$ is positive, then we assign the test observation to class 1, and if $f(x^*)$ is negative, then we assign the test observation to class -1.

Classification Using a Separating Hyperplane

- We can also make use of the *magnitude* of $f(x^*)$. If $f(x^*)$ is far from zero, then this means that x^* lies far from the hyperplane, and so we can be confident about our class assignment for x^*.
- On the other hand, if $f(x^*)$ is close to zero, then x^* is located near the hyperplane, and so we are less certain about the class assignment for x^*.

The Maximal Margin Classifier

- In general, if our data can be perfectly separated using a hyperplane, then there will in fact exist an infinite number of such hyperplanes. This is because a given separating hyperplane can usually be shifted a tiny bit up or down, or rotated, without coming into contact with any of the observations.

The Maximal Margin Classifier

- In order to construct a classifier based upon a separating hyperplane, we must have a reasonable way to decide which of the infinite possible separating hyperplanes to use.
- A natural choice is the *maximal margin hyperplane* (also known as the *optimal separating hyperplane*), which is the separating hyperplane that is farthest from the training observations.

The Maximal Margin Classifier

- That is, we can compute the (perpendicular) distance from each training observation to a given separating hyperplane; the smallest such distance is the minimal distance from the observations to the hyperplane, and is known as the *margin*.
- The maximal margin hyperplane is the separating hyperplane for which the margin is largest—that is, it is the hyperplane that has the farthest minimum distance to the training observations.

The Maximal Margin Classifier

- We can then classify a test observation based on which side of the maximal margin hyperplane it lies. This is known as the *maximal margin classifier*.
- We hope that a classifier that has a large margin on the training data will also have a large margin on the test data, and hence will classify the test observations correctly. Although the maximal margin classifier is often successful, it can also lead to overfitting when p is large.

The Maximal Margin Classifier

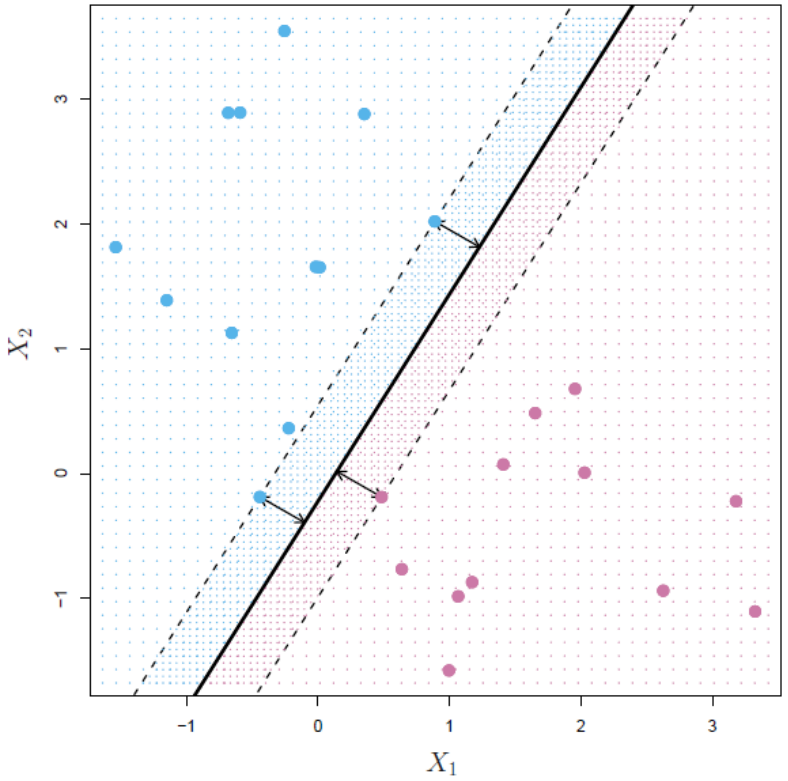

FIGURE 9.3. *There are two classes of observations, shown in blue and in purple. The maximal margin hyperplane is shown as a solid line. The margin is the distance from the solid line to either of the dashed lines. The two blue points and the purple point that lie on the dashed lines are the support vectors, and the distance from those points to the hyperplane is indicated by arrows. The purple and blue grid indicates the decision rule made by a classifier based on this separating hyperplane.*

The Maximal Margin Classifier

- If $\beta_0, \beta_1, \ldots, \beta_p$ are the coefficients of the maximal margin hyperplane, then the maximal margin classifier classifies the test observation x^* based on the sign of $f(x^*) = \beta_0 + \beta_1 x_1^* + \beta_2 x_2^* + \cdots + \beta_p x_p^*$.
- The maximal margin hyperplane represents the mid-line of the widest "slab" that we can insert between the two classes.

The Maximal Margin Classifier

- Three training observations are equidistant from the maximal margin hyperplane and lie along the dashed lines indicating the width of the margin.
- These three observations are known as *support vectors*, since they are vectors in p-dimensional space and they "support" the maximal margin hyperplane in the sense that if these points were moved slightly then the maximal margin hyperplane would move as well.

The Maximal Margin Classifier

- Interestingly, the maximal margin hyperplane depends directly on the support vectors, but not on the other observations: a movement to any of the other observations would not affect the separating hyperplane, provided that the observation's movement does not cause it to cross the boundary set by the margin.
- The fact that the maximal margin hyperplane depends directly on only a small subset of the observations is an important property.

Construction of the Maximal Margin Classifier

- We now consider the task of constructing the maximal margin hyperplane based on a set of n training observations $x_1, \ldots, x_n \in R^n$ and associated class labels $y_1, \ldots y_n \in \{-1, 1\}$.
- Briefly, the maximal margin hyperplane is the solution to the optimization problem

$$\underset{\beta_0, \beta_1, \ldots, \beta_p, M}{maximize} M$$

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

$$y_i\left(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip}\right) \geq M \quad \forall i = 1, \ldots, n.$$

Construction of the Maximal Margin Classifier

- The constraint $y_i\left(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip}\right) \geq M \;\; \forall i = 1, \ldots, n$ guarantees that each observation will be on the correct side of the hyperplane, provided that M is positive. (Actually, for each observation to be on the correct side of the hyperplane we would simply need $y_i\left(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip}\right) > 0$, so the constraint in fact requires that each observation be on the correct side of the hyperplane, with some cushion, provided that M is positive.)

Construction of the Maximal Margin Classifier

- $\sum_{j=1}^{p} \beta_j^2 = 1$ is not really a constraint on the hyperplane, since if $\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip} = 0$ defines a hyperplane, then so does $k(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip}) = 0$ for any $k \neq 0$.
- However, with this constraint the perpendicular distance from the ith observation to the hyperplane is given by $y_i\left(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip}\right)$.

Construction of the Maximal Margin Classifier

- Therefore, the two constraints ensure that each observation is on the correct side of the hyperplane and at least a distance M from the hyperplane.
- Hence, M represents the margin of our hyperplane, and the optimization problem chooses $\beta_0, \beta_1, \ldots, \beta_p$ to maximize M. This is exactly the definition of the maximal margin hyperplane!

The Non-separable Case

- The maximal margin classifier is a very natural way to perform classification, if a separating hyperplane exists. However, in many cases no separating hyperplane exists, and so there is no maximal margin classifier.
- In this case, the optimization problem has no solution with $M > 0$.

The Non-separable Case

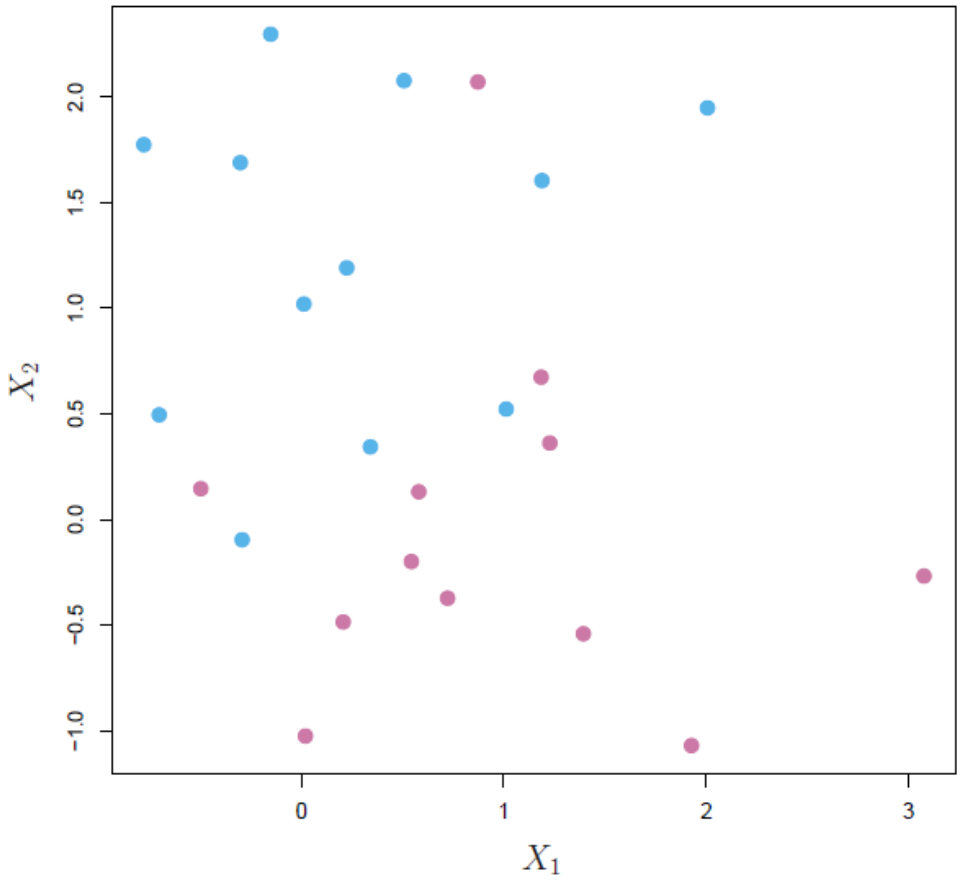

FIGURE 9.4. *There are two classes of observations, shown in blue and in purple. In this case, the two classes are not separable by a hyperplane, and so the maximal margin classifier cannot be used.*

The Non-separable Case

- In this case, we cannot *exactly* separate the two classes. However, we can extend the concept of a separating hyperplane in order to develop a hyperplane that *almost* separates the classes, using a so-called *soft margin*.
- The generalization of the maximal margin classifier to the non-separable case is known as the *support vector classifier*.

Support Vector Classifiers

- In fact, even if a separating hyperplane does exist, then there are instances in which a classifier based on a separating hyperplane might not be desirable.
- A classifier based on a separating hyperplane will necessarily perfectly classify all of the training observations; this can lead to sensitivity to individual observations.

Support Vector Classifiers

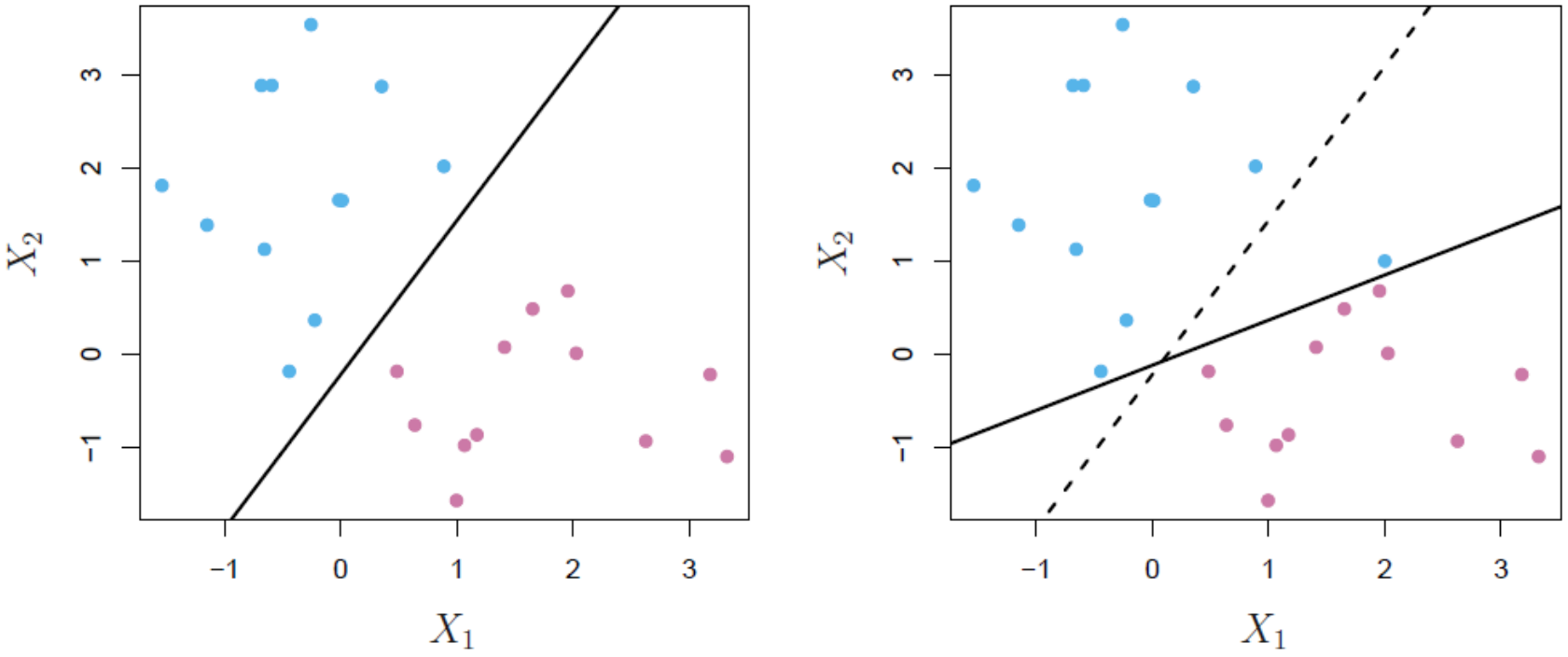

FIGURE 9.5. Left: *Two classes of observations are shown in blue and in purple, along with the maximal margin hyperplane.* Right: *An additional blue observation has been added, leading to a dramatic shift in the maximal margin hyperplane shown as a solid line. The dashed line indicates the maximal margin hyperplane that was obtained in the absence of this additional point.*

Support Vector Classifiers

- We might be willing to consider a classifier based on a hyperplane that does not perfectly separate the two classes, in the interest of
  - Greater robustness to individual observations, and
  - Better classification of most of the training observations.
- That is, it could be worthwhile to misclassify a few training observations in order to do a better job in classifying the remaining observations.

Support Vector Classifiers

- The *support vector classifier*, sometimes called a *soft margin classifier*, does exactly this.
- Rather than seeking the largest possible margin so that every observation is not only on the correct side of the hyperplane but also on the correct side of the margin, we instead allow some observations to be on the incorrect side of the margin, or even the incorrect side of the hyperplane. (The margin is *soft* because it can be violated by some of the training observations.)

Support Vector Classifiers

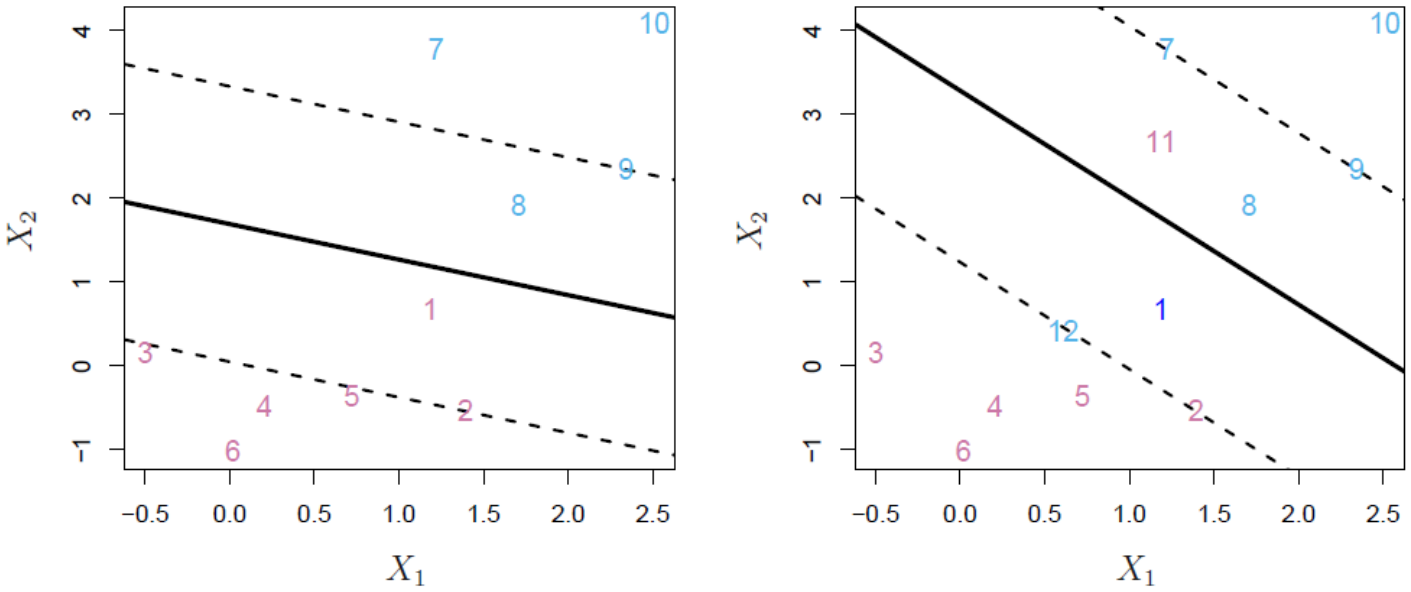

FIGURE 9.6. Left: *A support vector classifier was fit to a small data set. The hyperplane is shown as a solid line and the margins are shown as dashed lines.* Purple observations: *Observations* $3, 4, 5,$ *and* 6 *are on the correct side of the margin, observation* 2 *is on the margin, and observation* 1 *is on the wrong side of the margin.* Blue observations: *Observations* 7 *and* 10 *are on the correct side of the margin, observation* 9 *is on the margin, and observation* 8 *is on the wrong side of the margin. No observations are on the wrong side of the hyperplane.* Right: *Same as left panel with two additional points,* 11 *and* 12*. These two observations are on the wrong side of the hyperplane and the wrong side of the margin.*

Details of the Support Vector Classifier

- The support vector classifier classifies a test observation depending on which side of a hyperplane it lies.
- The hyperplane is chosen to correctly separate *most* of the training observations into the two classes, but may misclassify a few observations. It is the solution to the optimization problem

$$\underset{\beta_0, \beta_1, \ldots, \beta_p, \epsilon_1, \ldots, \epsilon_n, M}{maximize} M$$

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

$$y_i\left(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip}\right) \geq M(1 - \epsilon_i), \epsilon_i$$

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

Details of the Support Vector Classifier

- M is the width of the margin; we seek to make this quantity as large as possible.
- $\epsilon_1, \ldots, \epsilon_n$ are *slack variables* that allow individual observations to be on the wrong side of the margin or the hyperplane.
- C is a nonnegative tuning parameter.

Details of the Support Vector Classifier

- The slack variable ϵ_i tells us where the ith observation is located, relative to the hyperplane and relative to the margin.
- If $\epsilon_i = 0$, then the ith observation is on the correct side of the margin. If $\epsilon_i > 0$, then the ith observation is on the wrong side of the margin, and we say that the ith observation has violated the margin. If $\epsilon_i > 1$, then the ith observation is on the wrong side of the hyperplane.

Details of the Support Vector Classifier

- C bounds the sum of the $\epsilon_i's$, and so it determines the number and severity of the violations to the margin (and to the hyperplane) that we will tolerate.
- We can think of C as a budget for the amount that the margin can be violated by the n observations.
- If $C = 0$, then there is no budget for violations to the margin, and it must be the case that $\epsilon_1 = \cdots = \epsilon_n = 0$. It is just the maximal margin hyperplane optimization problem.

Details of the Support Vector Classifier

- For $C > 0$, no more than C observations can be on the wrong side of the hyperplane, because if an observation is on the wrong side of the hyperplane then $\epsilon_i > 1$ and we require $\sum_{i=1}^{n} \epsilon_i < C$.
- As the budget C increases, we become more tolerant of violations to the margin, and so the margin will widen.
- Conversely, as C decreases, we become less tolerant of violations to the margin and so the margin narrows.

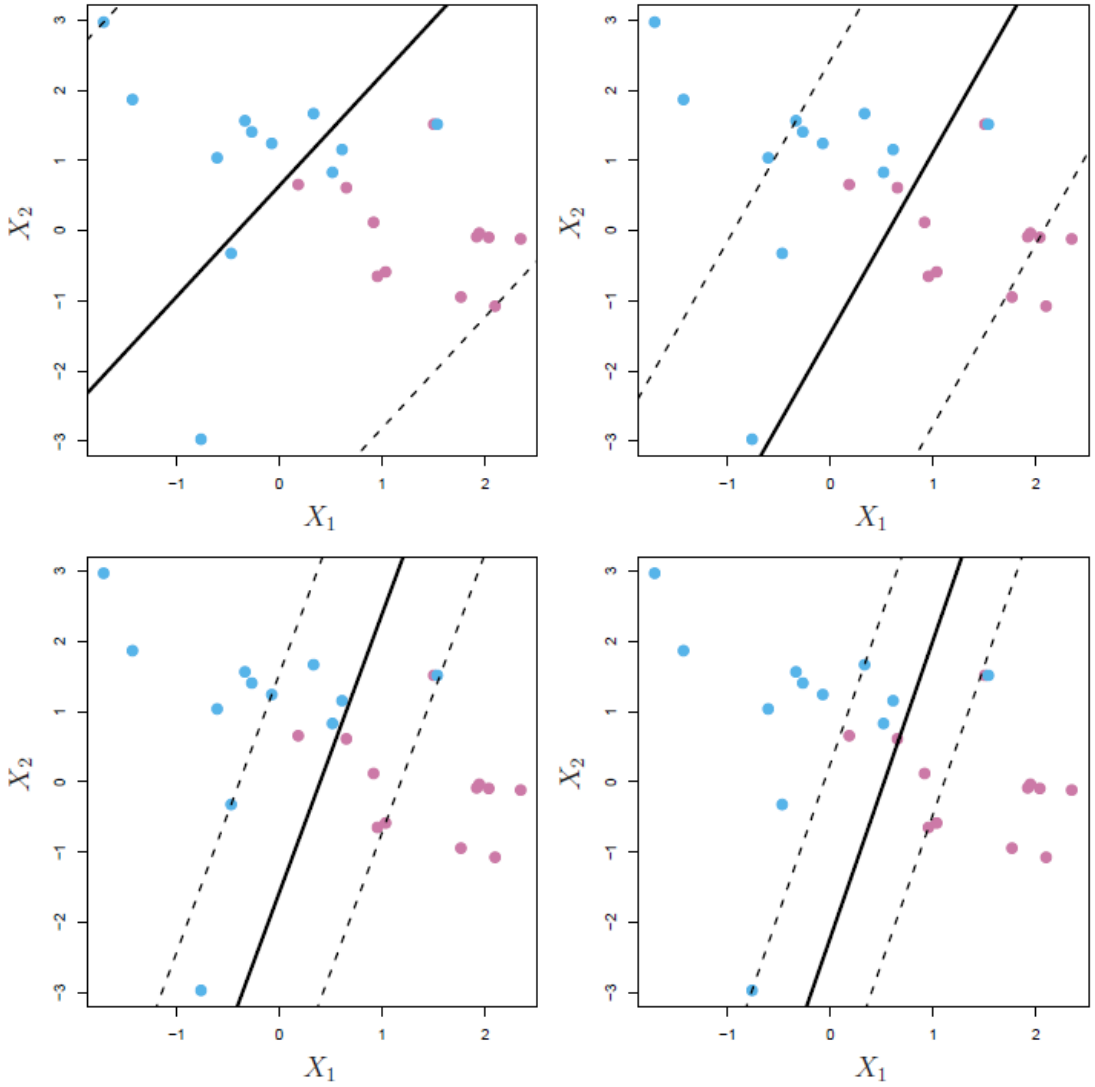

FIGURE 9.7. *A support vector classifier was fit using four different values of the tuning parameter C in (9.12)–(9.15). The largest value of C was used in the top left panel, and smaller values were used in the top right, bottom left, and bottom right panels. When C is large, then there is a high tolerance for observations being on the wrong side of the margin, and so the margin will be large. As C decreases, the tolerance for observations being on the wrong side of the margin decreases, and the margin narrows.*

Details of the Support Vector Classifier

- In practice, C is treated as a tuning parameter that is generally chosen via cross-validation.
- C controls the bias-variance trade-off of the statistical learning technique. When C is small, we seek narrow margins that are rarely violated; this amounts to a classifier that is highly fit to the data, which may have low bias but high variance.
- On the other hand, when C is larger, the margin is wider and we allow more violations to it; this amounts to fitting the data less hard and obtaining a classifier that is potentially more biased but may have lower variance.

Details of the Support Vector Classifier

- The optimization problem has a very interesting property: it turns out that only observations that either lie on the margin or that violate the margin will affect the hyperplane, and hence the classifier obtained.
- In other words, an observation that lies strictly on the correct side of the margin does not affect the support vector classifier! Changing the position of that observation would not change the classifier at all, provided that its position remains on the correct side of the margin.
- Observations that lie directly on the margin, or on the wrong side of the margin for their class, are known as support vectors. These observations do affect the support vector classifier.

Details of the Support Vector Classifier

- The fact that only support vectors affect the classifier is in line with our previous assertion that C controls the bias-variance trade-off of the support vector classifier.
- When the tuning parameter C is large, then the margin is wide, many observations violate the margin, and so there are many support vectors. In this case, many observations are involved in determining the hyperplane. This classifier has low variance (since many observations are support vectors) but potentially high bias.

Details of the Support Vector Classifier

- In contrast, if C is small, then there will be fewer support vectors and hence the resulting classifier will have low bias but high variance.
- The fact that the support vector classifier's decision rule is based only on a potentially small subset of the training observations (the support vectors) means that it is quite robust to the behavior of observations that are far away from the hyperplane.