BUAD 5082

Machine Learning II

Maximal Margin Classifiers
And
Support Vector Classifiers

(ISLR, Chapter 9)

BUAD 5082

Machine Learning II

Maximal Margin Classifiers
And
Support Vector Classifiers

(ISLR, Chapter 9)

Agenda

- Quick review of the mathematics of hyperplanes
- Three Approaches to binary classification (often collectively called Support Vector Machines, or SVM's)
  - A Separating Hyperplane
  - The Maximal Margin Classifier
  - The Support Vector Classifier
  - Support Vector Machines
- This talk addresses the first two (today and Monday).
- We'll then discuss the final two topic on Wednesday and Friday next week and conclude this sequence with some hands-on problems.

MAXIMAL MARGIN CLASSIFIER

Hyperplanes

- In a p-dimensional space, a *hyperplane* is a flat subspace of dimension $p - 1$ not necessarily passing through the origin.
  - In two dimensions, a hyperplane is a flat one-dimensional subspace (i.e. a line).
  - In three dimensions, a hyperplane is a flat two-dimensional subspace (i.e. a plane).
  - In $p > 3$ dimensions, it is impossible to visualize a hyperplane, but the notion of a $(p - 1)$-dimensional flat subspace still applies.

Hyperplanes

- Mathematically,
  - In two dimensions, a hyperplane is a line defined by the equation:

$$\beta_0 + \beta_1 x_1 + \beta_2 x_2 = 0$$

    - Another way to express this: any point $X = (x_1, x_2)^T$ for which this expression holds is a point on the hyperplane.
      - If this isn't obvious to you, re-write this expression as:

$x_2 = \frac{-\beta_0 - \beta_1 x_1}{\beta_2} = -\frac{\beta_1}{\beta_2} x_1 + \frac{-\beta_0}{\beta_2}$, and this X is a point on the straight line with intercept $-\frac{\beta_0}{\beta_2}$ and slope $\frac{-\beta_1}{\beta_2}$.

  - In a *p*-dimensional setting, a hyperplane is defined by:

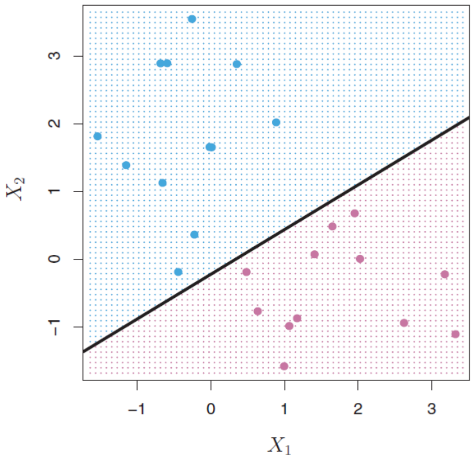
$$\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p = 0$$

    - Any point $x = (x_1, x_2, \ldots x_p)^T$ in *p*-dimensional space (i.e. a vector in p-dimensional space) that satisfies this expression lies on the hyperplane.
    - Equivalently, if a vector $x = \langle x_1, x_2, \ldots x_p, \rangle$ in "position representation" (initial point at the origin) has a terminal point on the plane, then $\beta_0 + x^T \beta = 0$

Hyperplanes

- Suppose that X does not satisfy this expression; rather,
$$\beta_0 + \beta_0 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p > 0$$
- Then this tells us that X lies to one side of the hyperplane, say “above”.
- Alternatively, if
$$\beta_0 + \beta_0 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p < 0$$
then X lies on the other side of the hyperplane, say “below”.
- 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.

Hyperplanes



- The hyperplane $1 + 2x_1 + 3x_2 = 0$ is shown.
  - The blue region is the set of points for which $1+2x_1 +3x_2 > 0$
  - The purple region is the set of points for which $1 + 2x_1 +3x_2 < 0$

Classification Using a Separating Hyperplane

- Consider our familiar setting in which we have an $n \times p$ data matrix $\mathbf{X}$ that consists of n training observations in p-dimensional space,

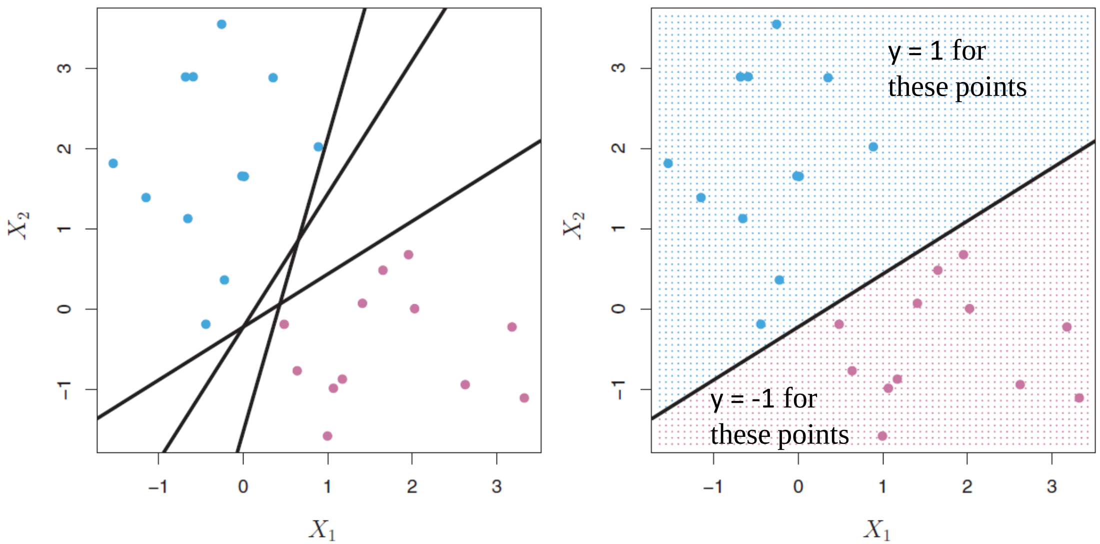
$$x_1 = \begin{pmatrix} x_{11} \\ \vdots \\ x_{1p} \end{pmatrix}, \ldots, x_n = \begin{pmatrix} x_{n1} \\ \vdots \\ x_{np} \end{pmatrix}$$

  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.
  - This is a notational departure from our previous (arbitrary) practice of denoting a binary response variable as 0/1. This is done for mathematical convenience, as we'll see shortly.
- We also have a test observation, a p-vector of observed features $x^* = (x_1^*, \ldots, x_p^*)^T$ that we wish to correctly classify using its feature measurements x_i^*.

Classification Using a Separating Hyperplane

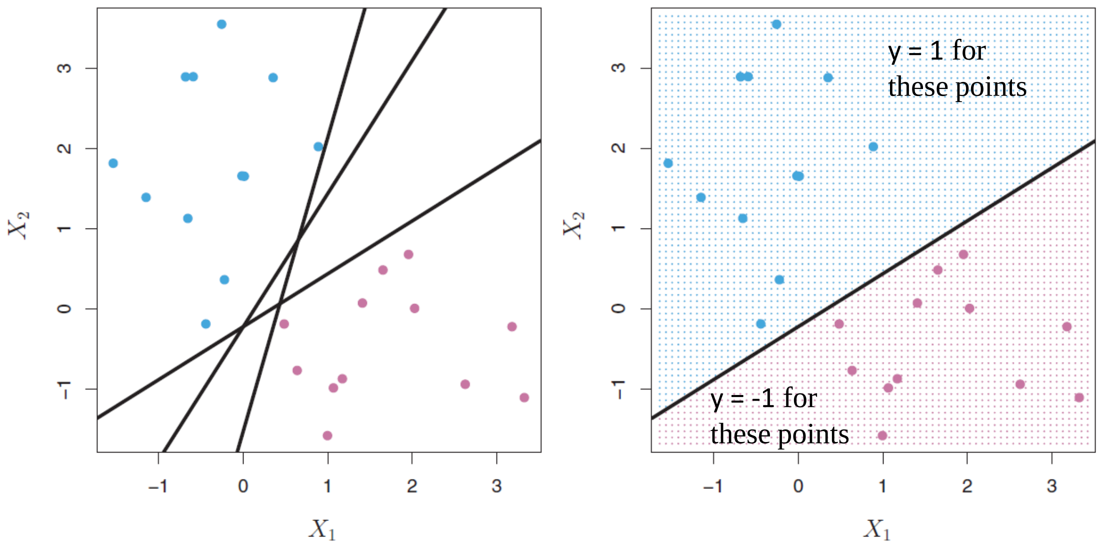
- We have seen a number of approaches for this task, such as KNN, Bernoulli Naïve Bayes, Logistic Regression , LDA, QDA, and Ridge and Lasso Classification.
- We will now see a new approach that is based upon the concept of a *separating hyperplane*.

Classification Using a Separating Hyperplane



- Suppose that it is possible to construct a hyperplane that separates the training observations perfectly according to their class labels.
  - Examples of three such *separating hyperplanes* are shown in the left-hand panel above.
- We have labeled the observations from the blue class as $y_i = 1$ and those from the purple class as $y_i = -1$.

Classification Using a Separating Hyperplane



- Then a separating hyperplane has the properties

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

and

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

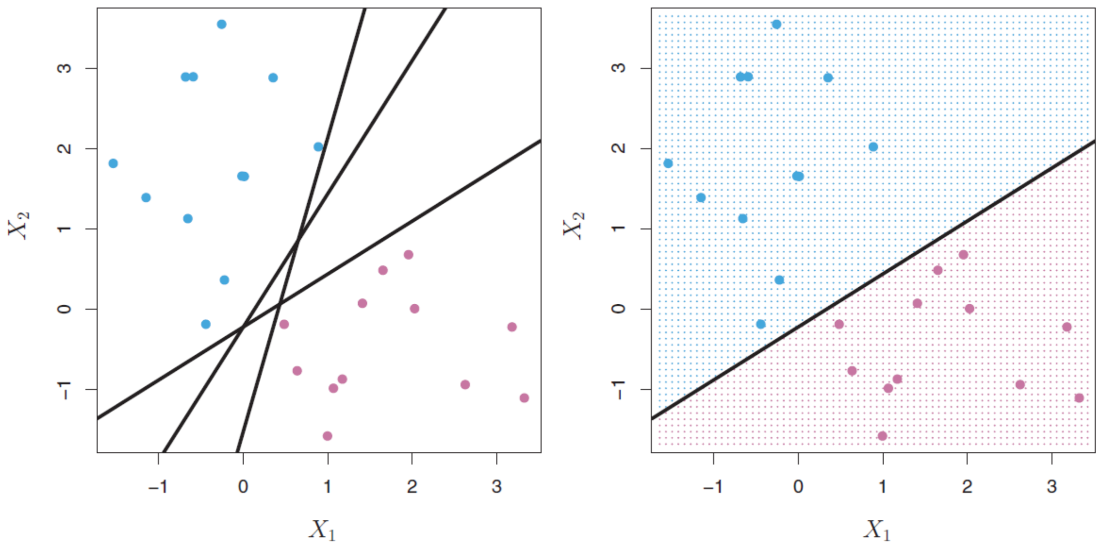
- Equivalently, a **Separating Hyperplane** has the property that

$$y_i(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip}) > 0$$

for all i in 1,…, n with x_i not on the hyperplane.

This is where the mathematical convenience comes in.

Classification Using a Separating Hyperplane

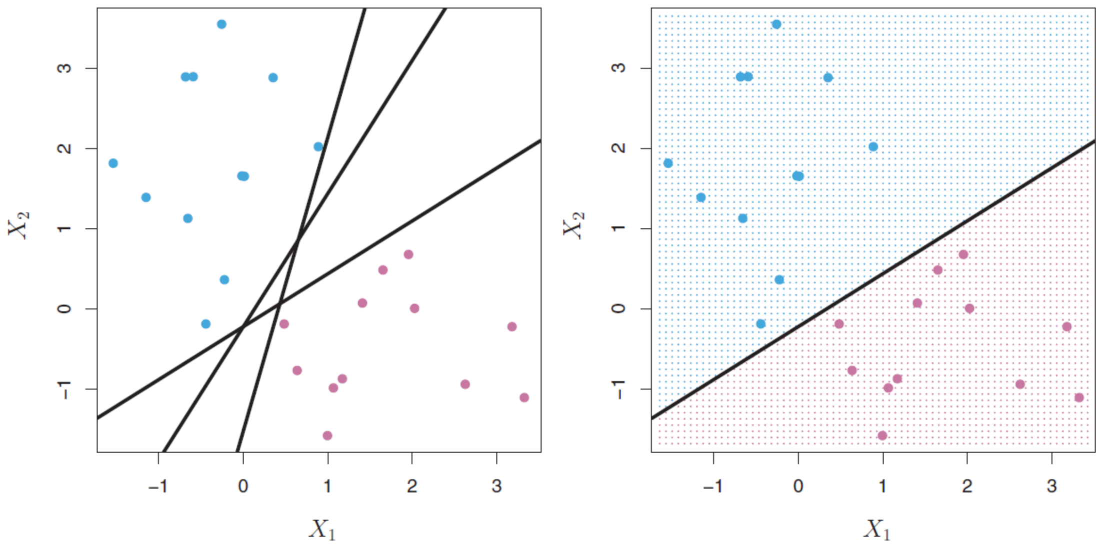


- If a separating hyperplane exists, we can use it to construct a very natural classifier:
- A test observation is assigned a class depending on which side of the hyperplane it is located.
- We classify x^* based on the sign of

$$f(x^*) = \beta_0 + \beta_1 x_1^* + \beta_2 x_2^* + \cdots + \beta_p x_p^*$$

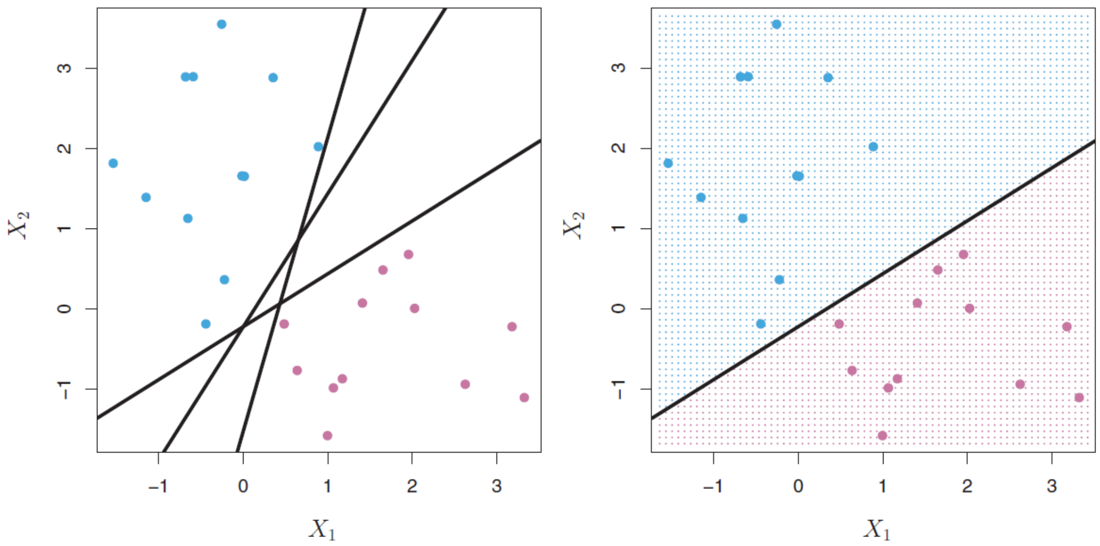
  - If $f(x^*)$ is positive, then we assign the test observation to class 1, and if $f(x^*)$ is negative, then we assign it to class -1.

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^*.
  - If $f(x^*)$ is close to zero, then x^* is located near the hyperplane, and we are less certain about the class assignment for x^*.
- As shown in the right-hand panel above, a classifier that is based on a separating hyperplane leads to a linear decision boundary.

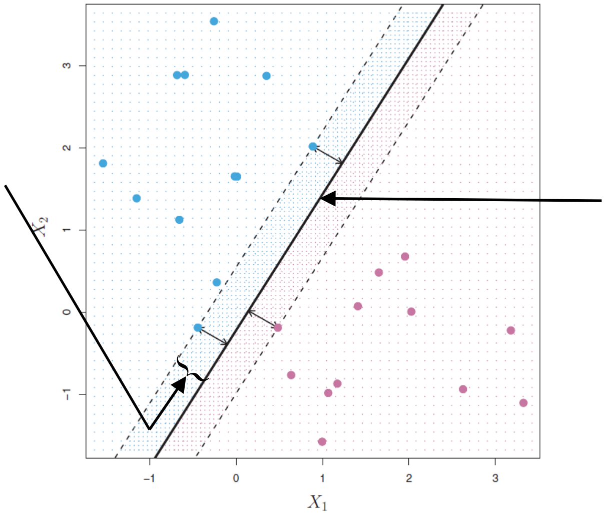
Classification Using a Separating Hyperplane



- In general, if our data can be perfectly separated using a hyperplane, then there will in fact exist an <u>infinite</u> 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.
- Three possible separating hyperplanes are shown in the left-hand panel above.
- In order to construct a classifier based upon a separating hyperplane, we must have a reasonable way to decide <u>which of the infinite possible separating hyperplanes to use</u>.

The Maximal Margin Classifier

The ***margin*** is the distance from the solid line to either of the dashed lines.



The ***Maximal Margin Hyperplane*** is the mid-line of the widest "slab" that we can insert between the two classes

- Intuitively, the ***maximal margin hyperplane*** (also known as the ***optimal separating hyperplane***), is the separating hyperplane that is farthest from the training observations.
- To determine which of the candidate separating hyperplanes is the maximal margin hyperplane, we proceed as follows:
  - For each separating hyperplane, compute the distance from each training observation to that hyperplane; the <u>smallest</u> such distance is known as the ***margin*** for that hyperplane.
  - The maximal margin hyperplane is the separating hyperplane for which the margin is <u>largest</u> - that is, it is the hyperplane that has the <u>farthest minimum distance to the training observations</u>.

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***.
- So in summary, 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

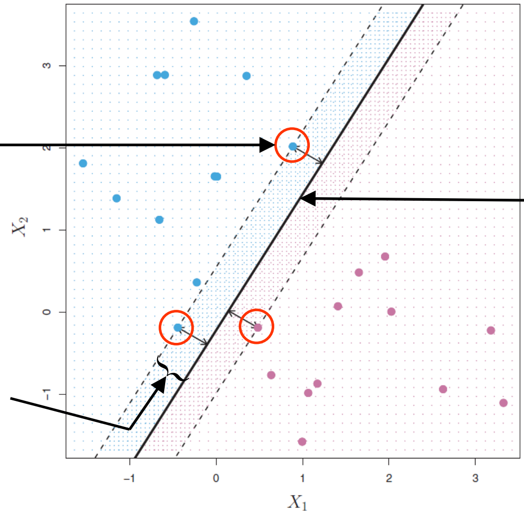
$$f(x^*) = \beta_0 + \beta_1 x_1^* + \beta_2 x_2^* + \cdots + \beta_p x_p^*$$

- **Important**: Our expectation (hope?) is that if the classifier has a large margin on the training data, it will also have a large margin on the test data, and hence will classify the test observations correctly.

The Maximal Margin Classifier

The ***Support Vectors***

The ***margin*** is the distance from the solid line to either of the dashed lines



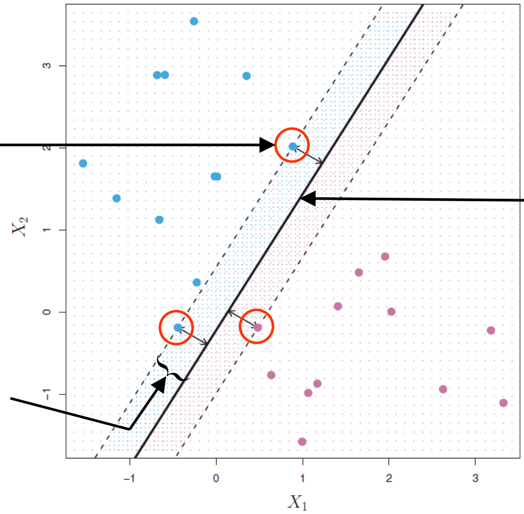
The ***Maximal Margin Hyperplane*** is the mid-line of the widest "slab" that we can insert between the two classes

- Observe that 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 (here $p = 2$) 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

The ***Support Vectors***

The ***margin*** is the distance from the solid line to either of the dashed lines



The ***Maximal Margin Hyperplane*** is the mid-line of the widest "slab" that we can insert between the two classes

- Note that 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 that will arise later when we discuss the support vector classifier and support vector machines.

Construction of the Maximal Margin Classifier

- To find the maximal margin hyperplane, we solve the following optimization problem:

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

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

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

- This is perhaps simpler than it looks:
  - The last constraint guarantees that each observation will be on the correct side of the hyperplane with some “cushion” (i.e. margin) *M*, provided that *M* is positive.

Construction of the Maximal Margin Classifier

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

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

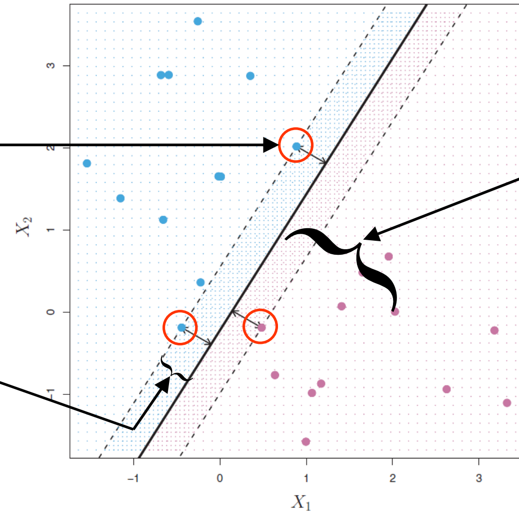
$$y_i(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \ldots + \beta_p x_{ip}) \geq M \quad \forall\, i = 1, \ldots, n$$

- The first constraint 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, one can show that with this constraint the perpendicular distance from the i th observation to the hyperplane is given by
  $$y_i(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip})$$

Construction of the Maximal Margin Classifier

The ***Support Vectors***

The ***margin***



The perpendicular distance from the i th observation is $y_i(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip})$ when $\|\beta\|=1$

- Therefore, taken together, 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.

Solution Note for the Maximal Margin Classifier

$$\max_{\beta,\beta_0} M$$

subject to

$$\|\beta\| = 1, \text{and}$$

$$y_i\left(x_i^T\beta + \beta_0\right) \geq M, i = 1, \dots, n.$$

- Above is our earlier expression of the optimization problem, expressed in vector form for convenience.
- The two constraints ensure that all the points are at least a distance M from the decision boundary defined by β and β_0, and we seek the largest such M.
- There is another slightly different formulation of this problem, which is more commonly seen in the literature.
- We can get rid of the first constraint by replacing the second constraint with
$$\frac{1}{\|\beta\|} y_i\left(x_i^T\beta + \beta_0\right) \geq M, i = 1, \dots, n.$$
- …or equivalently
$$y_i\left(x_i^T\beta + \beta_0\right) \geq M\|\beta\|, i = 1, \dots, n.$$

Solution Note for the Maximal Margin Classifier

$$\max_{\beta,\beta_0} M$$

subject to

$$y_i\left(x_i^T\beta + \beta_0\right) \geq M\|\beta\|, i = 1, \dots, n.$$

- We now have the problem expressed above.
- Since for any β and β_0 satisfying these inequalities, any positively scaled multiple satisfies them too, we can arbitrarily set $\|\beta\| = 1/M$. This, our problem can be expressed as

$$\min_{\beta,\beta_0} \|\beta\|$$

subject to

$$y_i\left(x_i^T\beta + \beta_0\right) \geq 1, i = 1, \dots, n.$$

Any β and β_0 that satisfies this expression also satisfies the following:

$$\min_{\beta,\beta_0} \frac{1}{2}\|\beta\|^2$$

subject to

$$y_i\left(x_i^T\beta + \beta_0\right) \geq 1, i = 1, \dots, n.$$

Solution Note for the Maximal Margin Classifier

$$\min_{\beta,\beta_0} \frac{1}{2}\|\beta\|^2$$

subject to

$$y_i\left(x_i^T\beta + \beta_0\right) \geq 1, i = 1, \ldots, N.$$

- Recall that earlier we showed that the distance from any point not on the hyperplane to the hyperplane is

$$\frac{1}{\|\beta\|} y_i\left(x_i^T\beta + \beta_0\right)$$

- The constraint therefore insures that there will be an empty "slab" surrounding the maximal margin hyperplane of thickness $1/\|\beta\|$. Hence we choose β and β_0 to maximize this thickness.

Solution Note for the Maximal Margin Classifier

- This is a convex quadratic optimization problem. The Lagrangian for the primal function to be minimized w.r.t. α, β and β_0, is

$$L_P = \frac{1}{2}||\beta||^2 - \sum_{i=1}^{N} \alpha_i [y_i (x_i^T \beta + \beta_0) - 1].$$

- Setting the partial derivatives equal to 0, solving for β, and substituting back into L_P, we obtain we obtain the dual optimization problem

$$\max_\alpha \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{k=1}^{N} \alpha_i \alpha_k y_i y_k x_i^T x_k$$

$$\text{subject to } \alpha_i \geq 0 \text{ and } \sum_{i=1}^{N} \alpha_i y_i = 0.$$

Observe that the solution depends only on the dot products of the training data. It can also be shown that the only points for which $\alpha_i \neq 0$ are the support vectors.

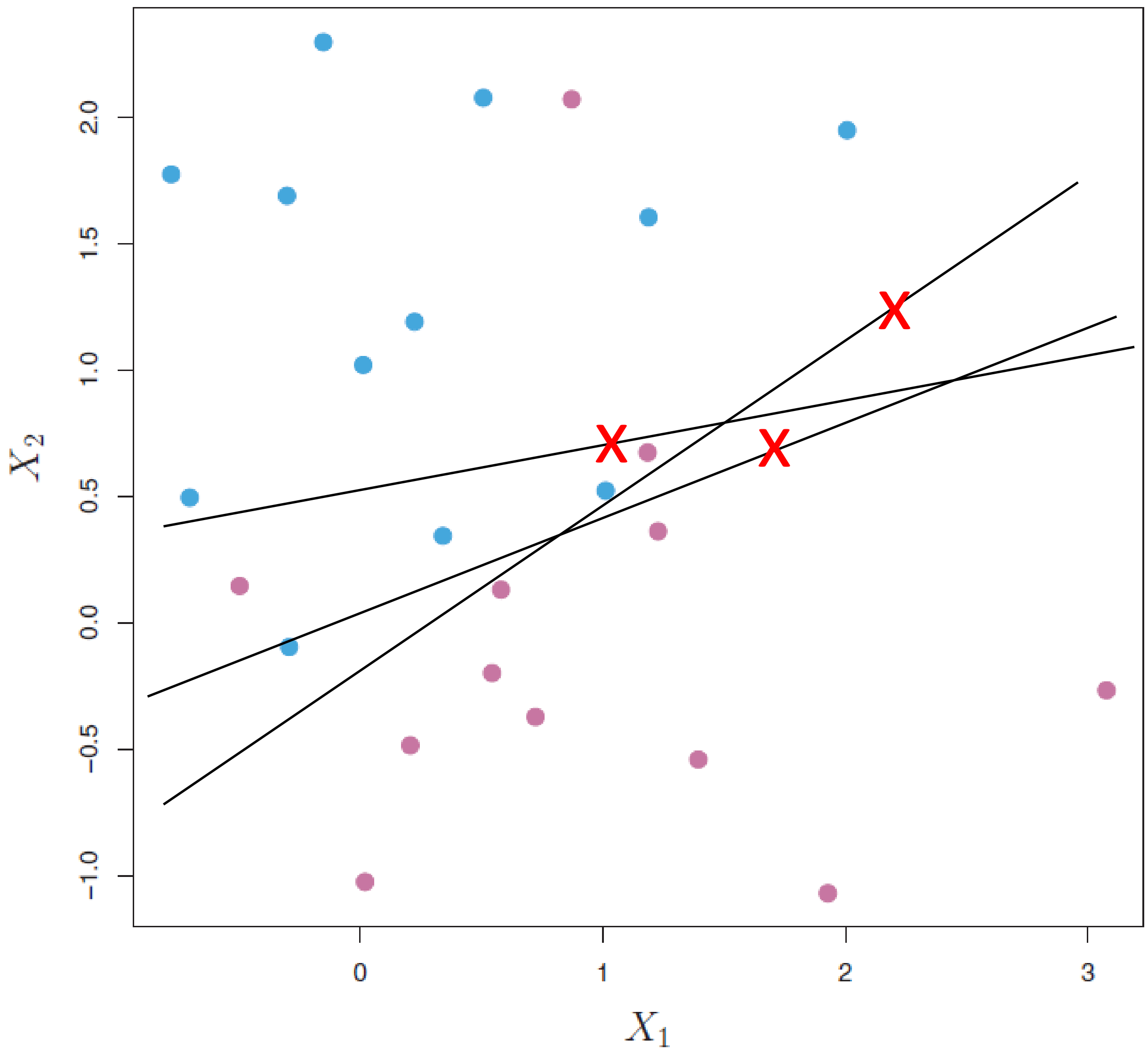
Solution Note for the Maximal Margin Classifier

$$\max_{\alpha} \sum_{i=1}^{N} \alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{k=1}^{N} \alpha_i \alpha_k y_i y_k x_i^T x_k$$

$$\text{subject to } \alpha_i \geq 0 \text{ and } \sum_{i=1}^{N} \alpha_i y_i = 0.$$

- In the dual problem above, we have a maximization problem with the parameters the α_i's, which is much easier to solve than the earlier quadratic optimization problem.
- Once we have found the α_i's, we can go back and substitute these into the partial derivative expressions to find the optimal betas as a function of the α_i's.

But What If Our Data Looks Like This?

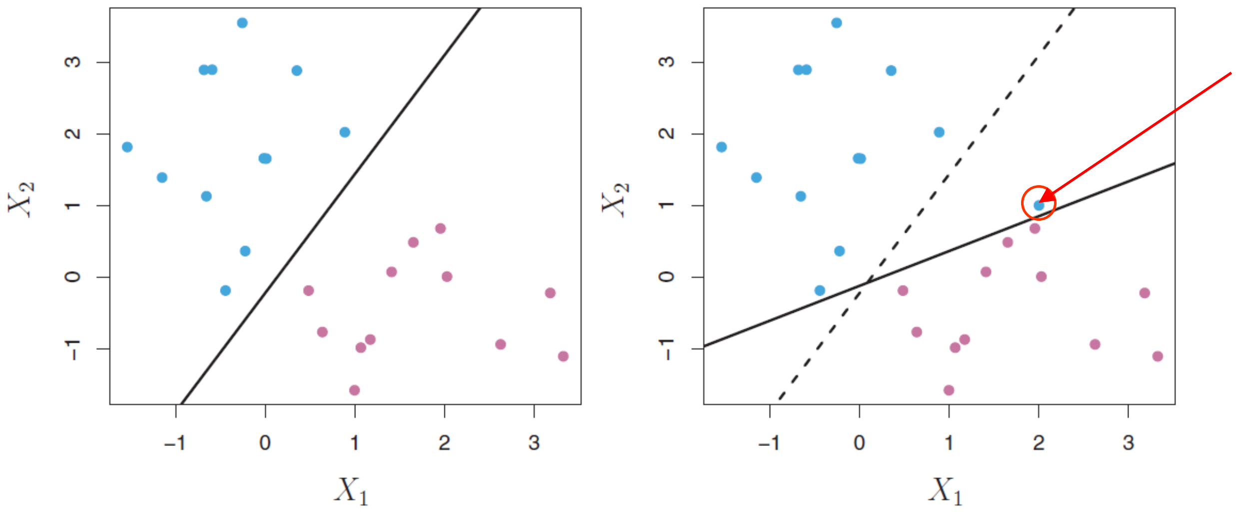


The Non-separable Case

- The maximal margin classifier is a very natural way to perform classification, *if a separating hyperplane exists*.
- However, in most cases no separating hyperplane exists, and so there is no maximal margin classifier (and the optimization problem just discussed has no solution).
- 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 CLASSIFIER

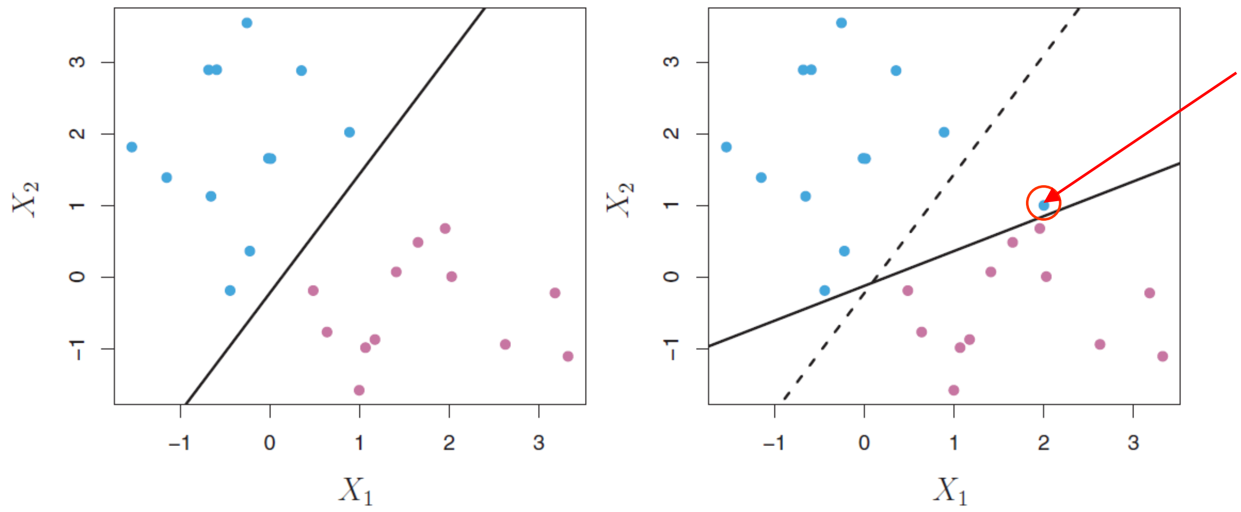
Overview



The addition of one additional point dramatically changes the maximal margin hyperplane

- There are instances in which a classifier based on a separating hyperplane, even if it exists, 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.
- In the example above, the addition of a single observation in the right-hand panel leads to a dramatic change in the maximal margin hyperplane.

Overview



The addition of one additional point dramatically changes the maximal margin hyperplane

- The resulting maximal margin hyperplane is not satisfactory for two reasons:
  - It has only a tiny margin. This is problematic because as discussed previously, the distance of an observation from the hyperplane can be seen as a measure of our confidence that the observation was correctly classified.
  - The fact that the maximal margin hyperplane is extremely sensitive to a change in a single observation suggests that it may have overfit the training data.

Overview

- In this case, 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
  - 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.

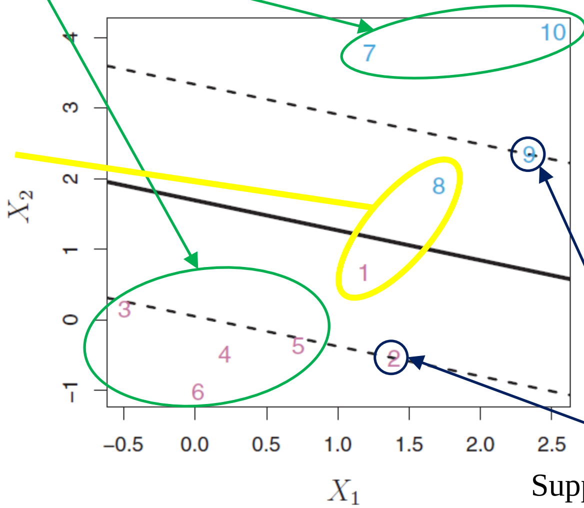
Overview

- 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 both on the correct side of the hyperplane and also on the correct side of the margin, we instead <u>allow some observations to be on the incorrect side of the margin</u>, or even the <u>incorrect side of the hyperplane</u> (thus the term “soft”).

Overview

Correct side of the hyperplane, correct side of the margin

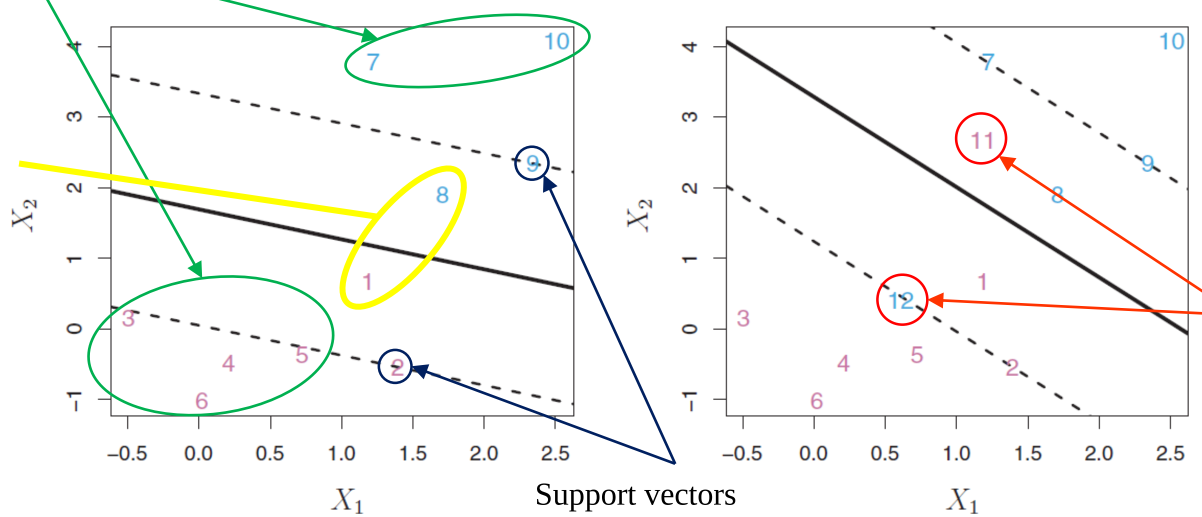
Correct side of the hyperplane, wrong side of the margin



Support vectors (on the margin)

- Example:
  - Left panel: All points are on the correct side of the hyperplane, and all but two are on the correct side of the margin.
  - However, an observation can be not only on the wrong side of the margin, but also on the wrong side of the hyperplane.
    - In fact, when there is no separating hyperplane, such a situation is inevitable.

Overview



- Example:
  - Left panel: All points are on the correct side of the hyperplane, and all but two are on the correct side of the margin.
  - However, an observation can be not only on the wrong side of the margin, but also on the wrong side of the hyperplane.
  - In fact, when there is no separating hyperplane, such a situation is inevitable.
  - Right panel: Observations on the wrong side of the hyperplane correspond to training observations that are misclassified by the support vector classifier
    - Points 11 and 12 are misclassified.

Details of the Support Vector Classifier

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

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

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

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

- We wish to choose a hyperplane that correctly separates *most* of the training observations into the two classes, but may misclassify a few observations.
- To find such a hyperplane, we solve the above optimization problem.

Details of the Support Vector Classifier

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

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

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

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

Changes to the maximal margin classifier constraints

- C is a nonnegative tuning parameter
- The $\epsilon_1, \ldots, \epsilon_n$ are *slack variables* that allow individual observations to be on the wrong side of the margin or the hyperplane (more later).
- As before, we classify the test observation x^* based on the sign of

$$f(x^*) = \beta_0 + \beta_1 x_1^* + \beta_2 x_2^* + \cdots + \beta_p x_p^*$$

Details of the Support Vector Classifier

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

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

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

$$\epsilon_i \geq 0, \quad \sum_{i=1}^{n} \epsilon_i \leq C, \qquad \text{(Constraint 2)}$$

- The slack variable ϵ_i tells us where the i^{th} observation is located relative to the hyperplane and relative to the margin.
  - If $\epsilon_i = 0$ then the RHS of Constraint 1 for the i^{th} observation is M, so the distance from the i^{th} observation to the hyperplane is at least M, and the observation is correctly classified and on the correct side of the margin.
  - If $0 < \epsilon_i < 1$ then the RHS of Constraint 1 for the i^{th} observation is less than M but still greater than 0, so the distance from the i^{th} observation to the hyperplane is greater than 0 but less than M, and the observation is correctly classified but on the wrong side of the margin.
  - If $\epsilon_i = 1$ then the RHS of Constraint 1 for the i^{th} observation is 0, so the distance from the i^{th} observation to the hyperplane is 0 and the observation is on the hyperplane, where its classification is inconclusive and it is still on the wrong side of the margin
  - If $\epsilon_i > 1$ then the observation is on the wrong side of the hyperplane, and so is misclassified.

Details of the Support Vector Classifier

- Consider the constraint $\sum_i^n \varepsilon_i \leq C$.
- *Since C* bounds the sum of the ε_i's, 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 all the ε_i's are 0, in which case the hyperplane we seek is just the maximal margin hyperplane optimization, which may not exist.
  - 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 ε_i > 1, and the ε_i's must sum to no more than *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.

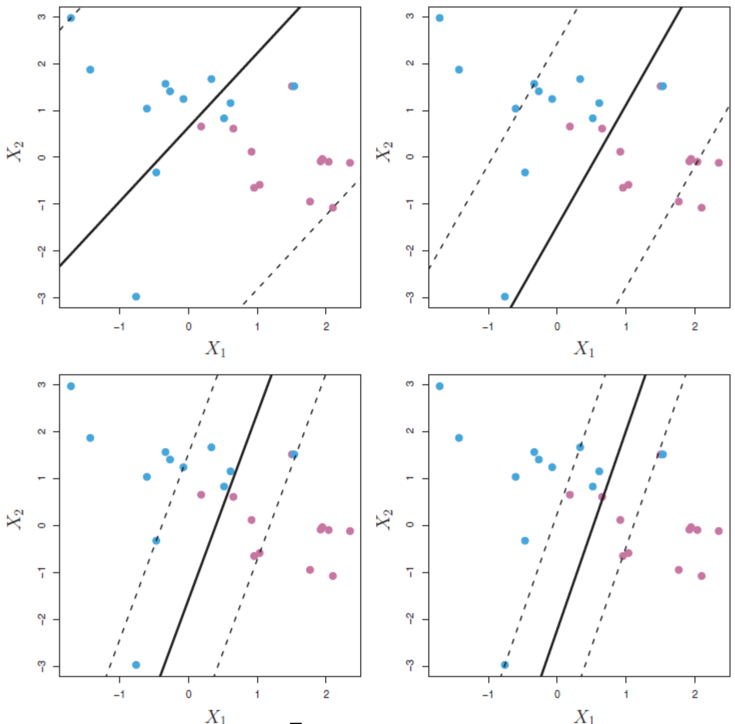
Details of the Support Vector Classifier

- Note 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 implies 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.

Details of the Support Vector Classifier



- The largest value of C was used in the top left panel, and smaller values were used in the other panels.
- When C is large, there is a high tolerance for observations being on the wrong side of the margin, a wider margin, more bias, relatively more support vectors, and lower variance.
- As C decreases, the tolerance for observations being on the wrong side of the margin decreases, leading to narrower margins, lower bias, fewer support vectors, and higher variance.

Details of the Support Vector Classifier

- 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.
- This property is distinct from some of the other classification methods that we have seen in preceding topics, such as linear discriminant analysis.
  - Recall that the LDA classification rule depends on the means μ_k of all of the observations within each class k, as well as the within-class covariance matrix $\sum$ computed using all of the observations.
    - That is, for more than one predictor, we computed

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

- In contrast, logistic regression has very low sensitivity to observations far from the decision boundary.
- In fact we will see that the support vector classifier and logistic regression are closely related.