Lecture 11 Support Vector Machine I

ECE 625: Data Analysis and Knowledge Discovery

Di Niu

Department of Electrical and Computer Engineering
University of Alberta

March 2, 2021

Outline

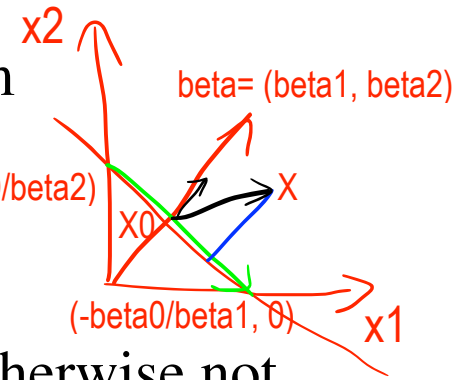
Separable Hyperplanes

- Imagine a situation where you have a two class classification problem with two predictors x_1 and x_2.
- Suppose that the two classes are linearly separable i.e. one can draw a straight line in which all points on one side belong to the first class and points on the other side to the second class.
- Then a natural approach is to find the straight line that gives the biggest separation between the classes i.e. the points are as far from the line as possible
- This is the basic idea of a support vector classifier.

Hyperplane

- A hyperplane in p dimensions is a flat affine subspace of dimension $p-1$.
- In general the equation for a hyperplane has the form

$$\beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p = 0.$$

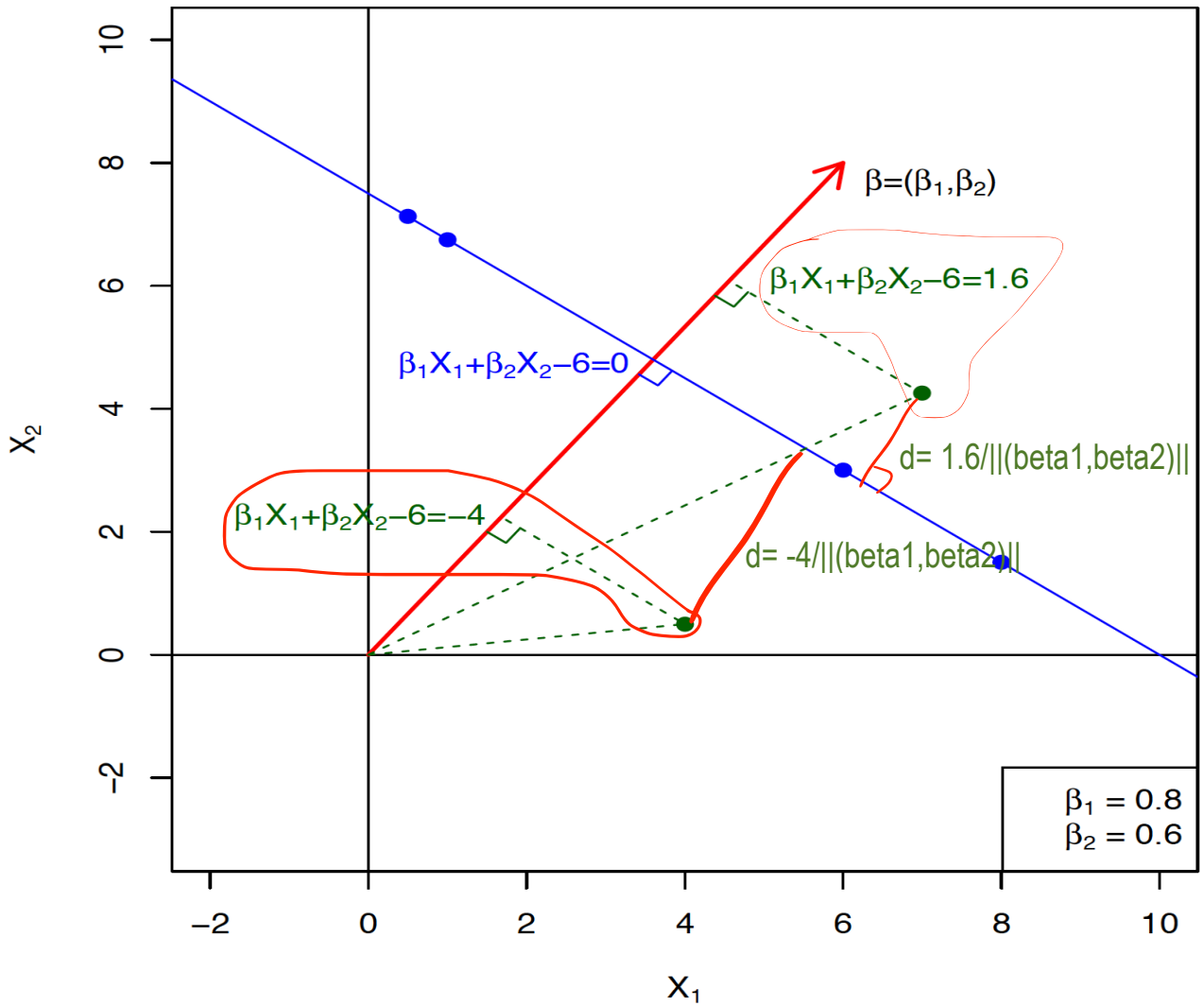


- In $p = 2$ dimensions a hyperplane is a line.
- If $\beta_0 = 0$, the hyperplane goes through the origin, otherwise not.
- The vector $\beta = (\beta_1, \cdots, \beta_p)$ is called the normal vector — it points in a direction orthogonal to the surface of a hyperplane.
- The signed distance of any point $X = (x_1, \ldots, x_p)$ to the hyperplane is

$$\frac{\beta^T}{\|\beta\|} \cdot (X - X_0) = \frac{1}{\|\beta\|}(\beta^T X + \beta_0),$$

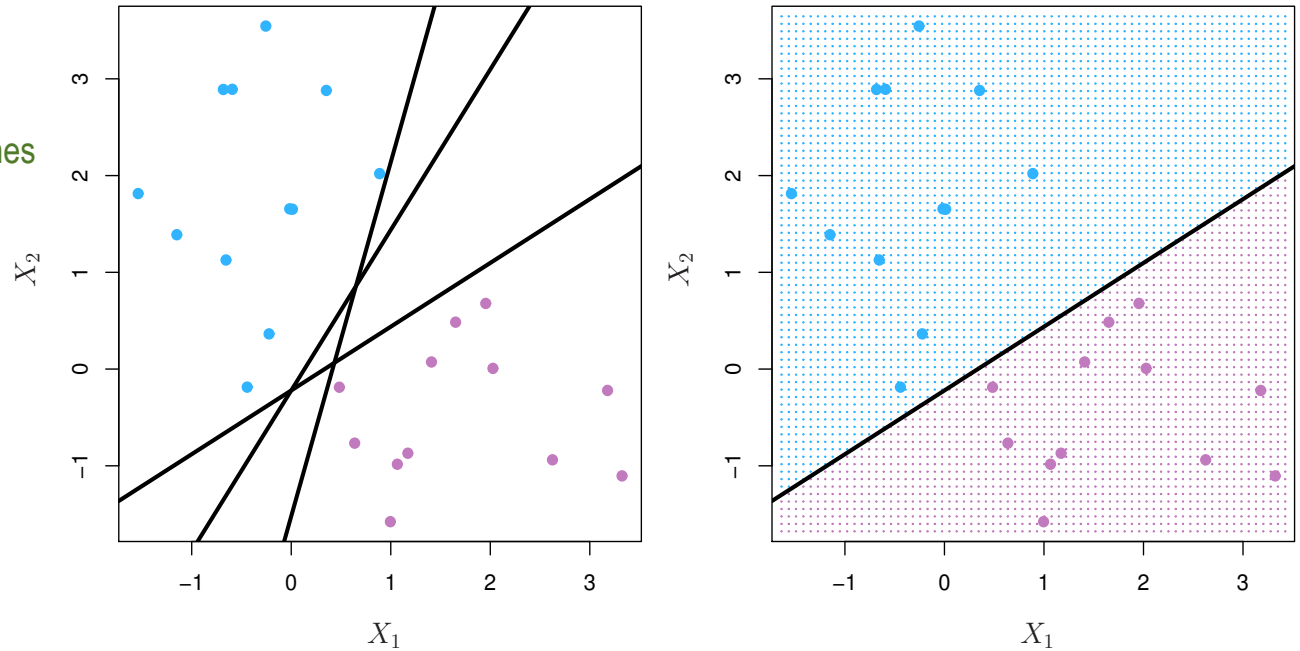
where X_0 is the projection of the zero vector onto the hyperplane.

Hyperplane in 2 Dimensions



Separating Hyperplane

All three lines are separating hyperplanes



- If $f(X) = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p$, then $f(X) > 0$ for points on one side of the hyperplane, and $f(X) < 0$ for points on the other.
- If we code the colored points as $Y_i = +1$ as blue and $Y_i = -1$ as purple, then if $Y_i \cdot f(X_i) > 0$ for all training samples $i = 1, \ldots, n$, $f(X) = 0$ defines a Separating Hyperplane.

Hard Margin

- Among all separating hyperplanes, find the one that makes the biggest gap or margin between the two classes.
- Constrained optimization problem

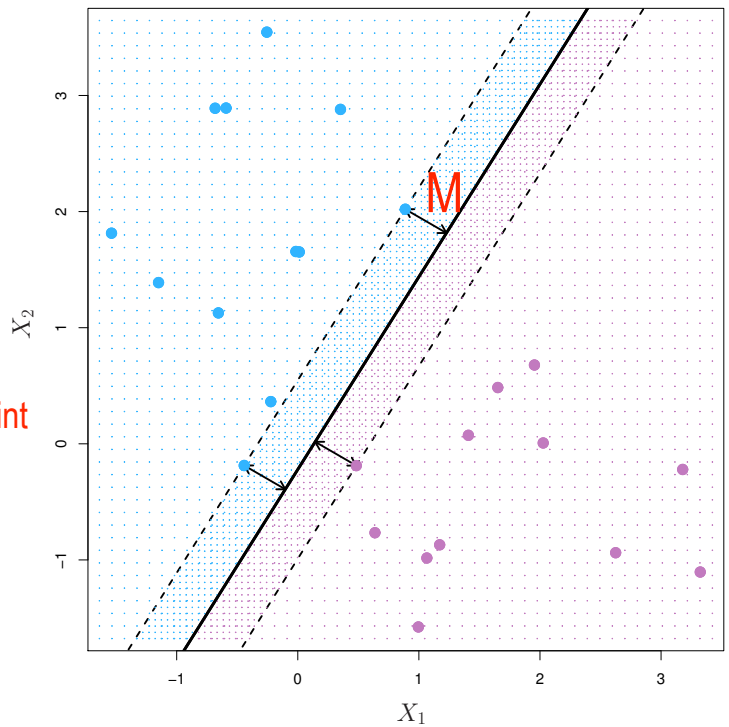
$$\text{maximize}_{\beta_0,\beta_1,\cdots,\beta_p} M$$

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

$$Y_i(\beta_0 + \beta^T X_i) \geq M$$

$$\text{for } i = 1, \cdots, n.$$

a nonlinear constraint



maximize M
subject to Yi (beta0 + beta^T Xi)/||beta|| >= M, for i = 1, ..., n

maximize M
subject to Yi (beta0/||beta|| + beta^T Xi/||beta||) >= M, for i = 1, ..., n

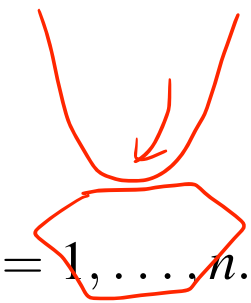
maximize M
subject to Yi (beta0' + (beta')^T Xi) >= M, for i = 1, ..., n
||beta'|| = 1

Hard Margin

maximize M
subject to Yi (beta0' + (beta')^T Xi) >= M, for i = 1, ..., n
||beta'|| = 1
maximize M
subject to Yi (beta0'/M + (beta'/M)^T Xi) >= 1, for i = 1, ..., n
||beta'|| = 1
maximize M <===> minimize ||beta''|| (because beta'' = beta'/M)
subject to Yi (beta0'' + (beta'')^T Xi) >= 1, for i = 1, ..., n
||beta'|| = 1

- Let $\beta = (\beta_1, \cdots, \beta_p)$. The problem above can be rewritten as a convex quadratic program:

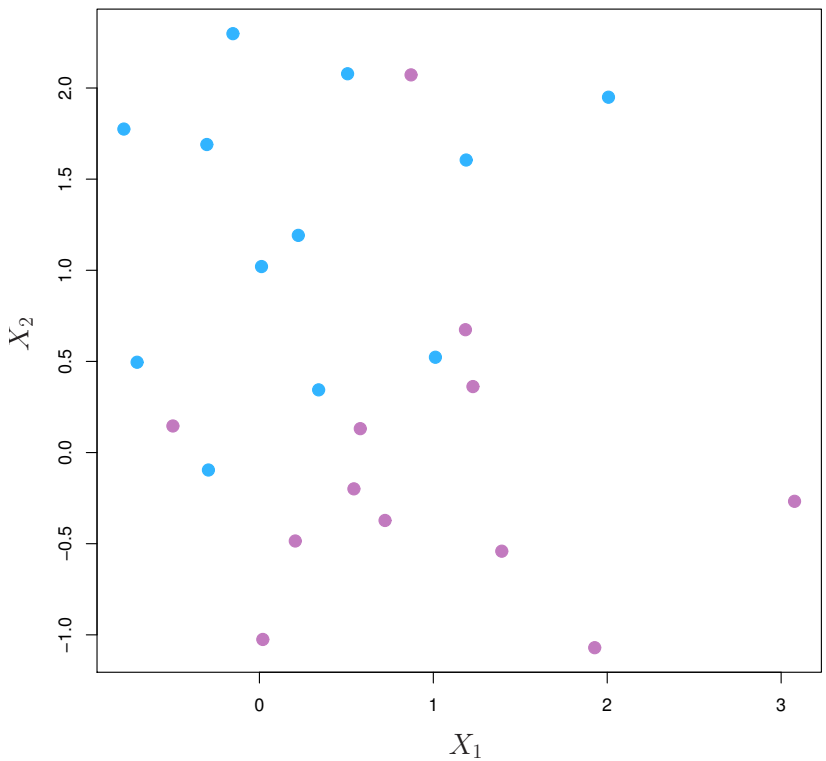
$$\begin{aligned} \underset{\beta_0, \beta}{\text{minimize}} \quad & \frac{1}{2}\|\beta\|^2 \\ \text{subject to} \quad & Y_i(\beta_0 + \beta^{\mathsf{T}} X_i) \geq 1, \quad \text{for } i = 1, \ldots, n. \end{aligned}$$



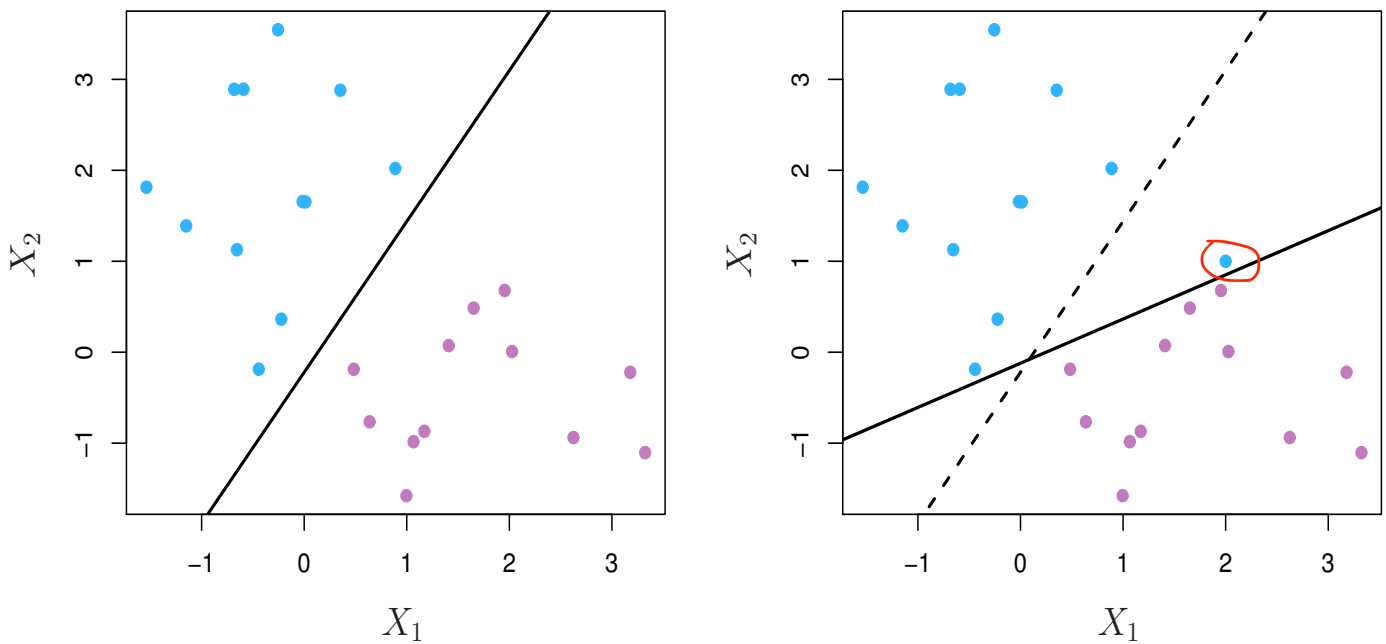
- The function `svm()` in package `e1071` solves this problem efficiently.

Hard Margin

- The data on the left are not separable by a linear boundary.
- In general it is true for $n \ll p$.
- The hard separating constraint doesn't work.

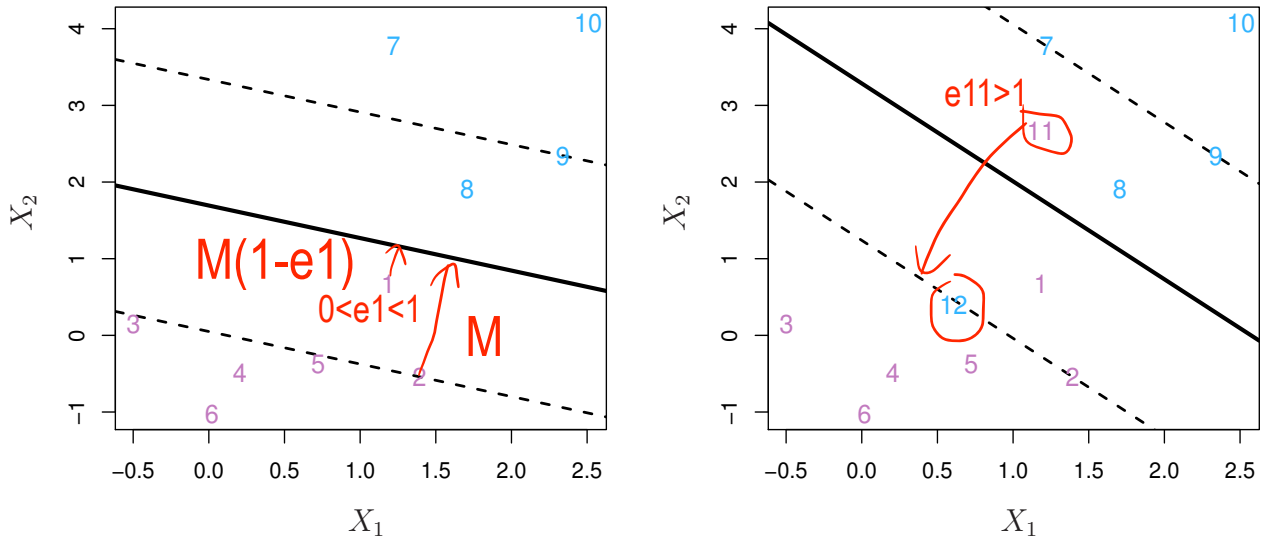


Hard Margin



- Sometimes the data are separable, but noisy. This can lead to a poor solution (overfitting) for the maximal-margin (hard margin) classifier. boundary.

Soft Margin



- The support vector classifier maximizes a soft margin.

$$\underset{\beta_0,\beta_1,\cdots,\beta_p;\epsilon_1,\cdots\epsilon_n}{\text{maximize}} \; M; \text{ subject to } \sum_{j=1}^{p} \beta_j^2 = 1$$

$$Y_i(\beta_0 + \beta^{\mathsf{T}} X_i) \geq M(1 - \epsilon_i)$$

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

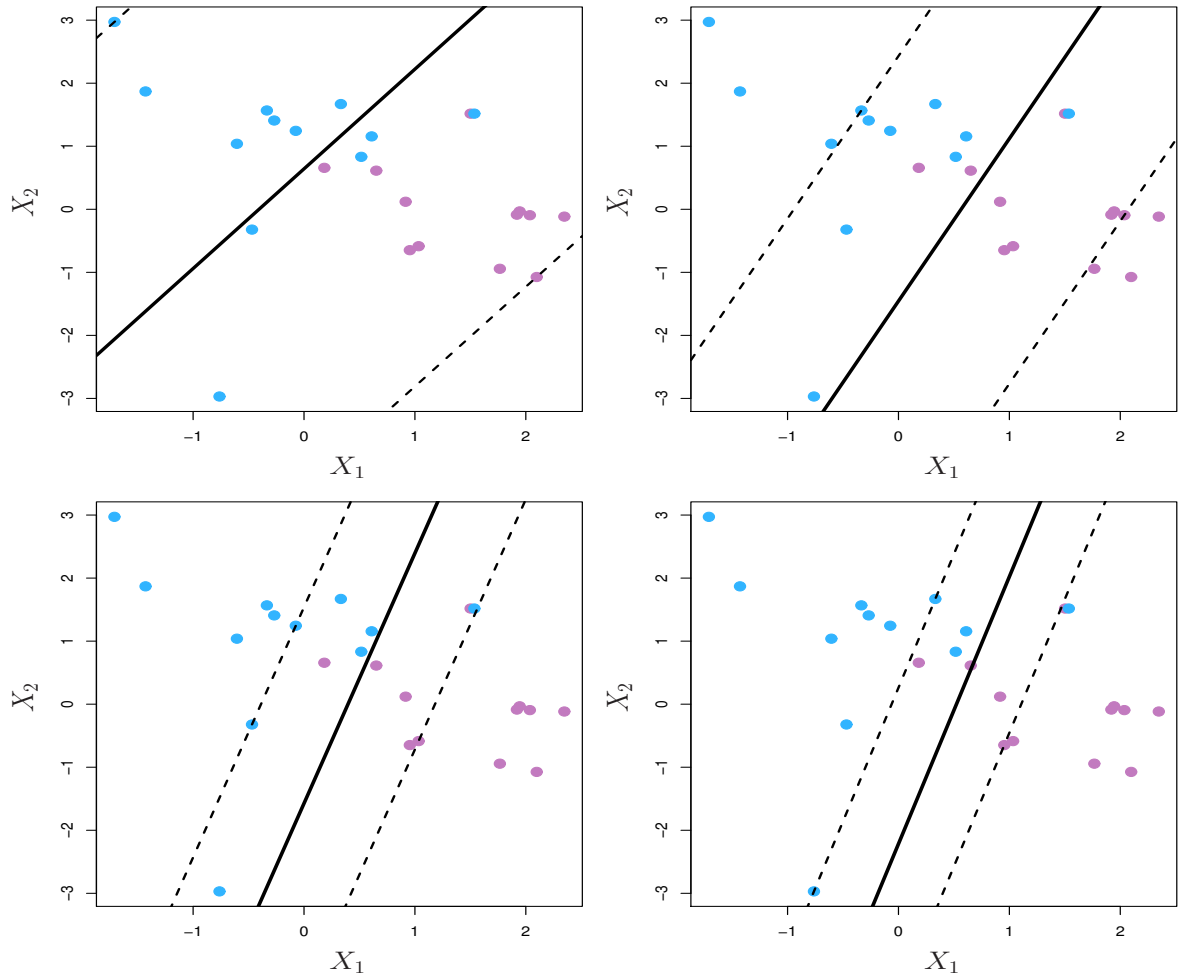
The slack variable ϵ_i (in the solution) tells us where the ith observation is located:

- If $\epsilon_i = 0$, it is on the correct side of the margin; (margin is between the dashed lines)
- If $\epsilon_i > 0$, it is on the wrong side of the margin (*violating* the margin);
- If $\epsilon_i > 1$, it is on the wrong side of the hyperplane;

C is a budget for the amount that the margin can be violated:

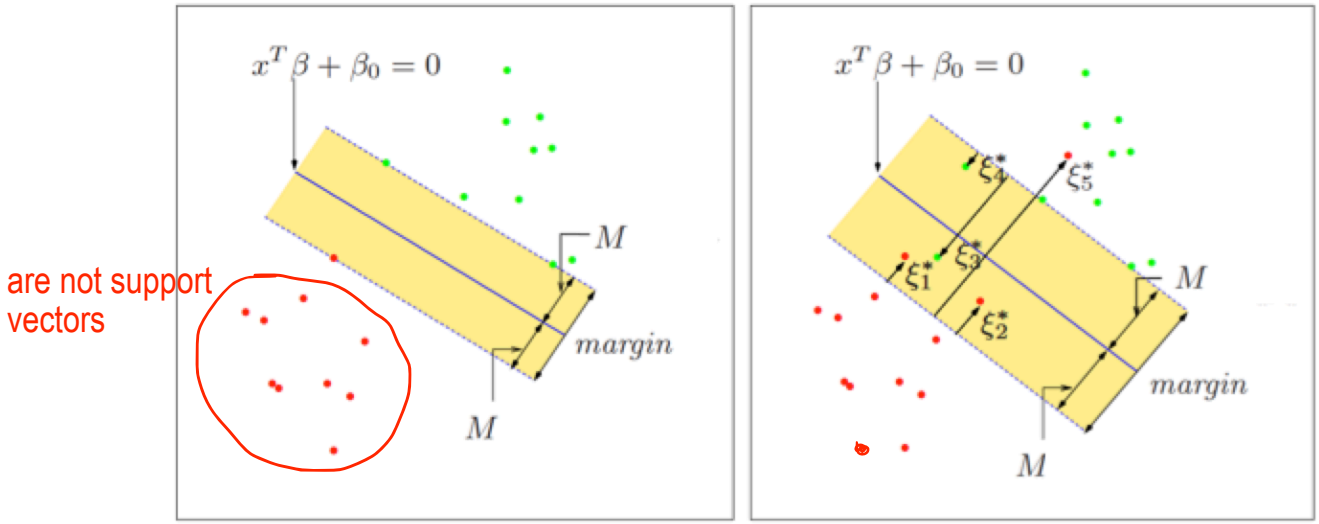
- $C = 0$: $\epsilon_i = 0$, hard margin;
- $C > 0$: no more than C observations can be on the wrong side of the hyperplane;

C is a regularization parameter



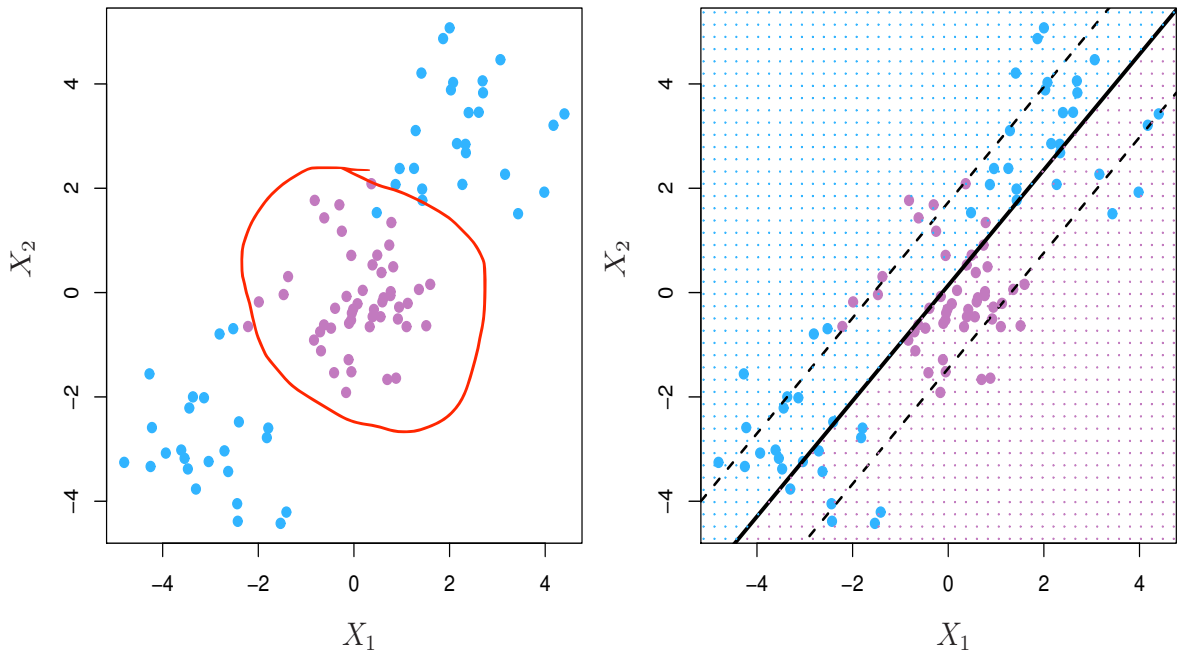
- As C increases, the margin will widen; controls the bias-variance tradeoff.

Support Vectors



- Support vectors: observations that lie directly on the margin, or on the wrong side of the margin for their class.
- Only the support vectors determine the optimization solution for both hard margin and soft margin.
- Similar to logistic regression with low sensitivity to observations far from the decision boundary; Different from LDA, where all the observations can affect the decision boundary.

Linear boundary can fail



- Sometimes a linear boundary simply won't work, no matter what value of C.
- For example, in the situation shown above.
- What do we do? the kernel trick!!!

Summary and Remark

- Hyperplane
- Support vector classifier
- Read textbook Chapter 12 and `R` code
- Do `R` lab