COM 525000 – Statistical Learning

Lecture 9 – Support Vector Machines

Y.-W. Peter Hong

Overview

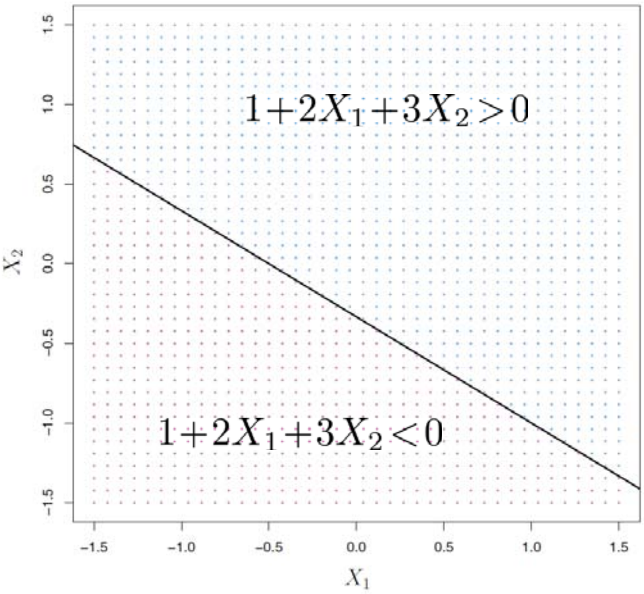
- **Maximal Margin Classifier:** Finds a hyperplane that separates the data points from two different classes.
- **Support Vector Classifier:** An extension of maximal margin classifier to incorporate non-separable data.
- **Support Vector Machine:** An extension of supper vector classifier to allow for nonlinear decision boundaries.

➔SVC and SVM are now generally referred to as "support vector machines".

Hyperplane

- In a p-dimensional space, a hyperplane is an affine subspace of dimension $p-1$ described by
$$\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p = 0.$$

- For example, for 2-dim. spaces, a hyperplane is a "line" described by
$$\beta_0 + \beta_1 X_1 + \beta_2 X_2 = 0$$
for 3-dim. spaces, it is a "plane" described by
$$\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 = 0.$$

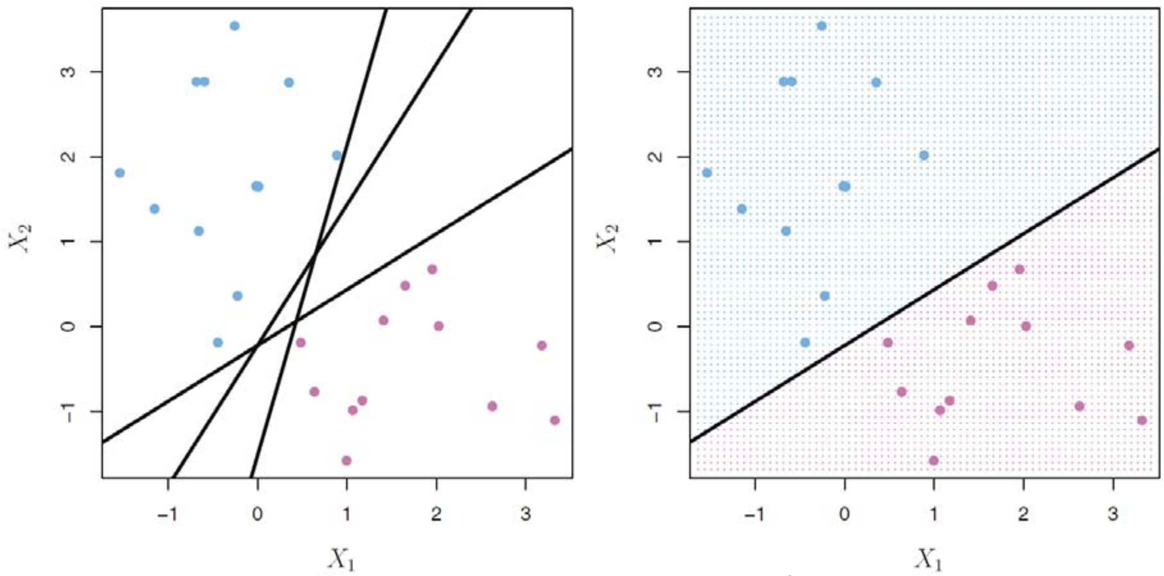


Separating Hyperplane

- Let $\mathbf{X}$ be an $n \times p$ data matrix consisting of n p-dim. training observations
$$x_1 = (x_{11}, \cdots, x_{1p})^T, \ldots, x_n = (x_{n1}, \cdots, x_{np})^T$$
and responses $y_1, \ldots, y_n \in \{-1, +1\}$.

- A separating hyperplane is a hyperplane that separates the training observations perfectly according to their class labels. That is, for all i,
$$\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip} > 0, \text{ if } y_i = 1,$$
$$\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip} < 0, \text{ if } y_i = -1.$$
(Or, equivalently, $y_i(\beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}) > 0$.)

Example



- Using separating hyperplane $f(x) \triangleq \beta_0 + \cdots + \beta_p x_{ip} = 0$, we can obtain classifier

$$\hat{y}^* = \begin{cases} 1, & \text{if } f(x^*) > 0, \\ -1, & \text{if } f(x^*) < 0. \end{cases}$$

Maximal Margin Hyperplane

- The optimal separating hyperplane is defined as the *maximal margin hyperplane*, i.e., the separating hyperplane that has the farthest minimum distance to the training observations (i.e., margin).
- It is the solution to the optimization problem:

$$\max_{\beta_0, \ldots, \beta_p, M} M$$

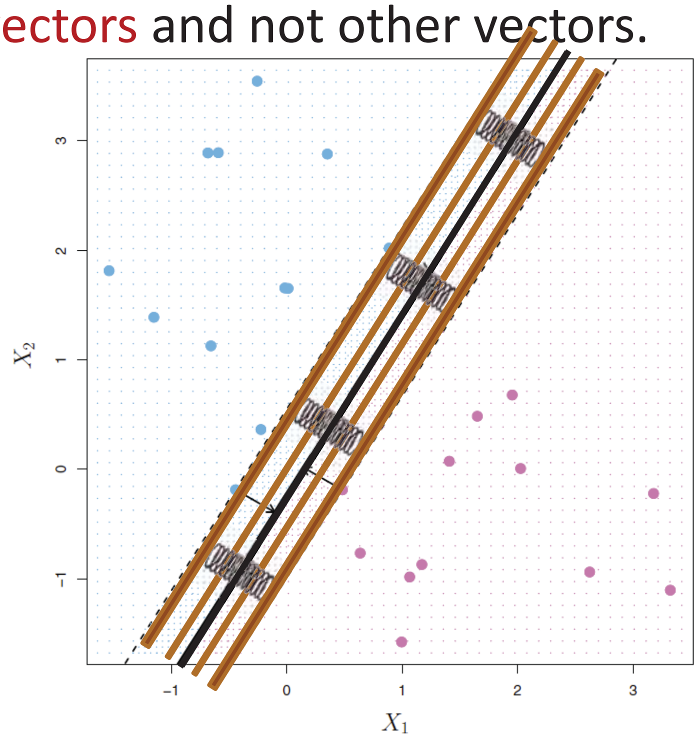
$$\text{subject to } \sum_{j=1}^{p} \beta_j^2 = 1$$

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

- The maximal margin classifier classifies observations based on the maximal margin hyperplane.

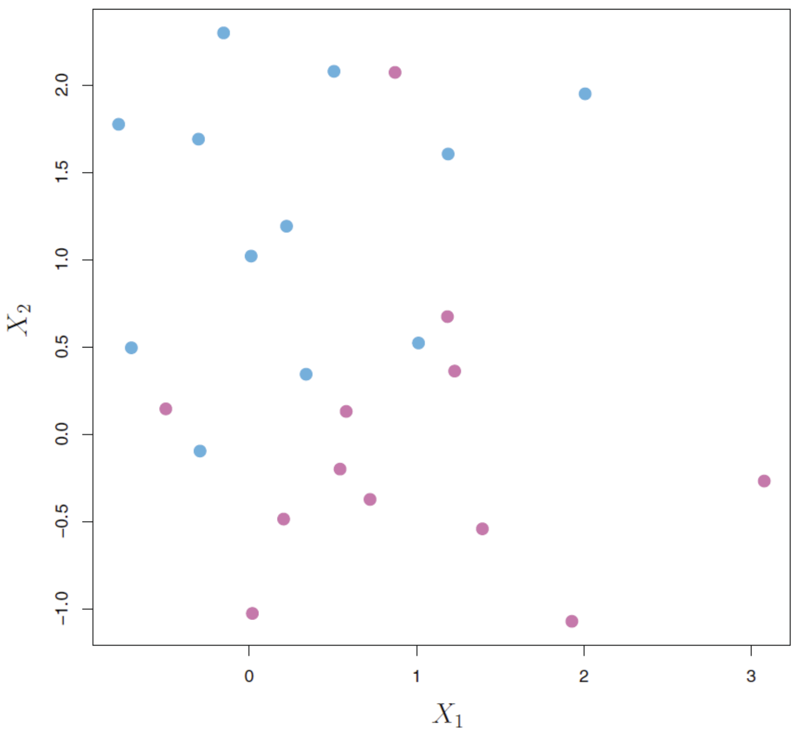
Support Vectors

- The maximal margin hyperplane depends only on the support vectors and not other vectors.



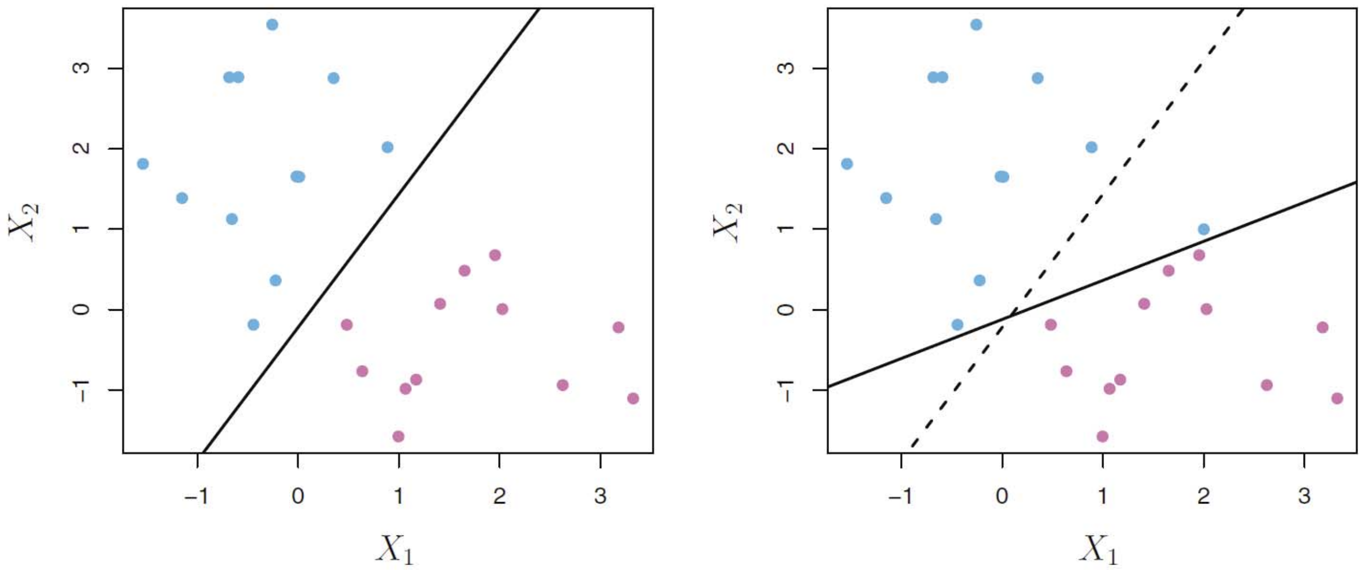
Example

- Example of non-separable data points.



Example

- Example with small or sensitive margins.



Support Vector Classifier

- The support vector classifier (or *soft margin classifier*) that utilizes a hyperplane that does NOT perfectly separate the two classes for (i) better robustness and (ii) better classification of *most* training observations.
- With tuning parameter C, find the hyperplane:

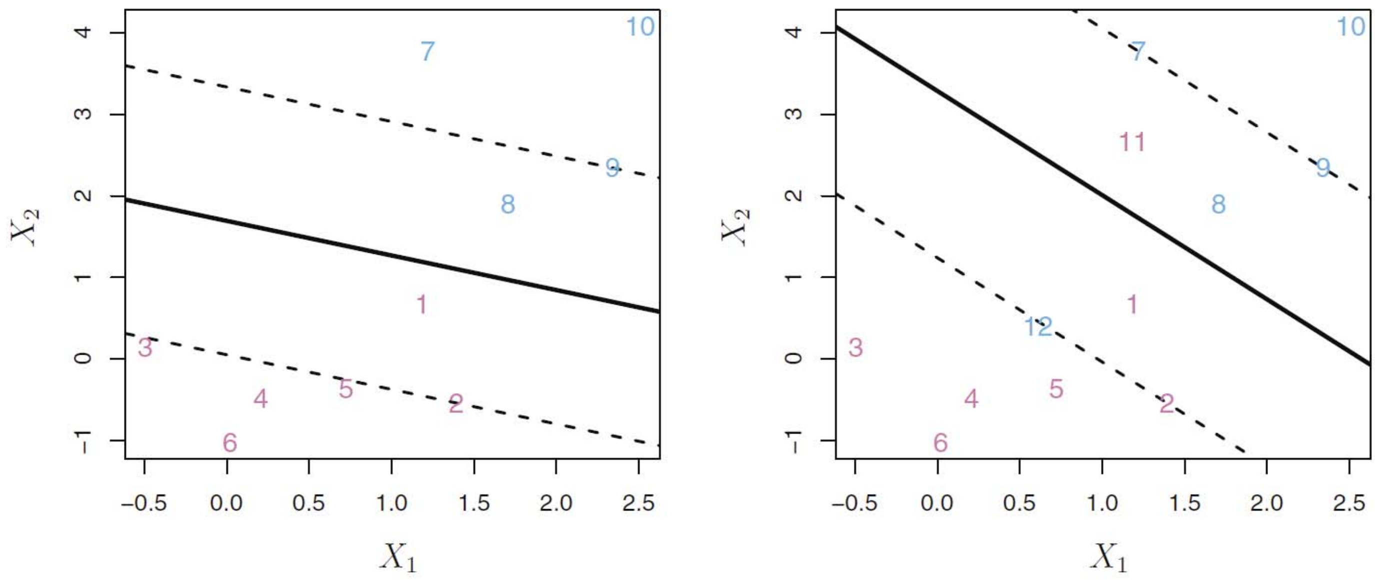
$$\max_{\beta_0,\ldots,\beta_p,\epsilon_1,\ldots,\epsilon_n,M} M$$

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

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

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

Example

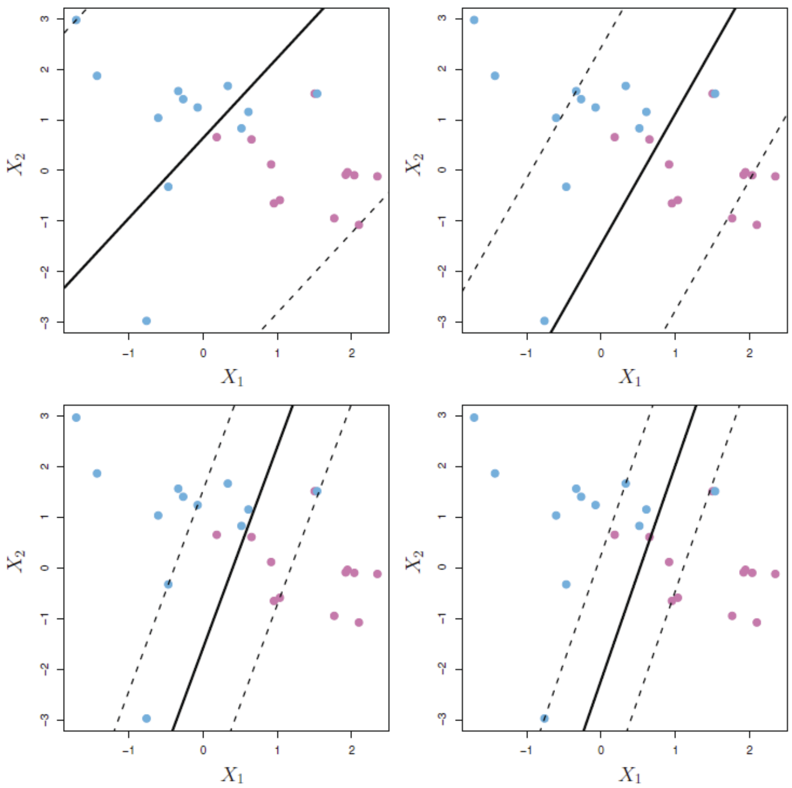


Interpretation

- Slack variable ϵ_i tells us the location of i-th point.
  - $\epsilon_i = 0$: i-th point is on the correct side of the margin.
  - $\epsilon_i > 0$: it is on the wrong side of the margin.
  - $\epsilon_i > 1$: it is on the wrong side of the hyperplane.
- Parameter C represents the budget of violations.
  - $C = 0$: yields the maximal margin hyperplane.
  - C increases: more tolerant of violations to the margin.

  ➔Controls bias-variance tradeoff, and can be chosen via CV.
- **Remark:** Only observations that lie directly on the margin, or on the wrong side of it, affect the hyperplane and, thus, are called *support vectors*.

Examples



Solving Support Vector Classifiers (1/5)

- Let us reformulate the problem as

$$\max_{\beta_0', \beta, \epsilon_1, \ldots, \epsilon_n, M} M$$

$$\text{subject to } \frac{1}{\|\beta\|} y_i(\beta_0' + \beta^T x_i) \geq M(1 - \epsilon_i),$$

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

where $\beta = (\beta_1, \ldots, \beta_p)^T$ and $\beta_0' = \beta_0 \|\beta\|^2$.

- W.l.o.g., set $\|\beta\| = 1/M$ to get

$$\min_{\beta_0', \beta, \epsilon_1, \ldots, \epsilon_n, M} \frac{1}{2}\|\beta\|^2 + \gamma \sum_{i=1}^{n} \epsilon_i$$

$$\text{subject to } y_i(\beta_0' + \beta^T x_i) \geq (1 - \epsilon_i), \ \epsilon_i \geq 0, \ \forall i.$$

Solving Support Vector Classifiers (2/5)

- The Lagrange function is

$$L = \frac{1}{2}\|\beta\|^2 + \gamma\sum_{i=1}^{n}\epsilon_i - \sum_{i=1}^{n}\alpha_i[y_i(\beta_0' + \beta x_i) - (1-\epsilon_i)] - \sum_{i=1}^{n}\mu_i\epsilon_i.$$

where $\alpha_i \geq 0,\ \mu_i \geq 0.$

- The Lagrange dual optimization problem is

$$\max_{\alpha_i \geq 0, \mu_i \geq 0}\ \min_{\beta, \beta_0', \epsilon_i} L(\beta, \beta_0', \epsilon_i, \alpha_i, \mu_i)$$

- By setting the derivatives w.r.t. $\beta, \beta_0', \epsilon_i$ to 0, we get

Solving Support Vector Classifiers (3/5)

- Solve numerically the dual optimization problem

$$\begin{aligned}&\max_{\alpha_i, \mu_i}\ L_D(\alpha_i, \mu_i)\\ &\text{subject to } \alpha_i \geq 0, \mu_i \geq 0\\ &\qquad \alpha_i = \gamma - \mu_i,\\ &\qquad \sum_{i=1}^{n}\alpha_i y_i = 0\end{aligned} \quad\Rightarrow\quad \begin{aligned}&\max_{\alpha_i, \mu_i}\ L_D(\alpha_i, \mu_i)\\ &\text{subject to } 0 \leq \alpha_i \leq \gamma\\ &\qquad \sum_{i=1}^{n}\alpha_i y_i = 0\end{aligned}$$

Solving Support Vector Classifiers (4/5)

- By KKT conditions, we know that

$$\begin{aligned}\alpha_i[y_i(\beta^T x_i + \beta_0) - (1 - \epsilon_i)] &= 0\\ \mu_i \epsilon_i &= 0\\ y_i(\beta^T x_i + \beta_0) - (1 - \epsilon_i) &\geq 0.\end{aligned}$$

- This implies that α_i is nonzero only if

$$y_i(\beta^T x_i + \beta_0) = 1 - \epsilon_i$$

(i.e., if x_i is a support vector).

- By denoting the set of support vectors by $\mathcal{S}$, we have

$$\hat{\beta} = \sum_{i \in \mathcal{S}} \hat{\alpha}_i y_i x_i.$$

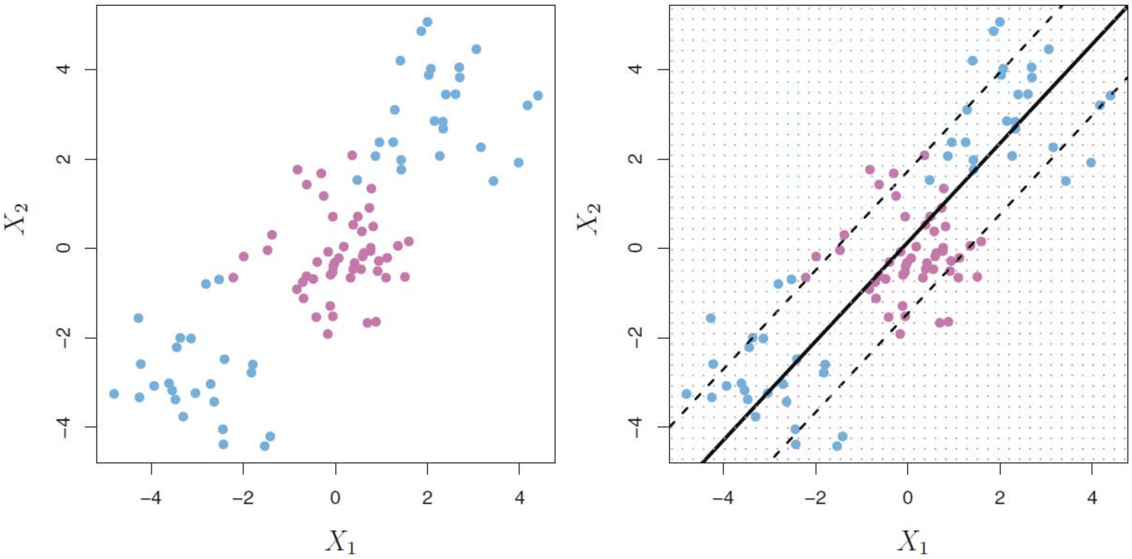
Solving Support Vector Classifiers (5/5)

- In fact, for $i \in \mathcal{S}$,

$$\begin{aligned}\hat{\alpha}_i &= \gamma \text{ if } \epsilon_i > 0 \text{ (i.e., } x_i \text{ is not on the margin)}\\ \epsilon_i &= 0 \text{ if } \hat{\alpha}_i \in (0, \gamma) \text{ (i.e., } x_i \text{ is on the margin)}\end{aligned}$$

- Let $\mathcal{S}_M \subset \mathcal{S}$ be the set of points on the margin such that $\hat{\alpha}_i \in (0, \gamma)$. Then, for $i \in \mathcal{S}_M$,

Extend to Nonlinear Decision Boundaries



$$\max_{\beta_0,\ldots,\beta_p,\epsilon_1,\ldots,\epsilon_n,M} M$$

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

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

Support Vector Machines (1/4)

- Support vector machine (SVM) is an extension of support vector classifiers that enlarges the feature space using *kernels*.
- Recall that the Lagrange dual function for SVC is

$$L_D=\sum_{i=1}^{n}\alpha_i-\frac{1}{2}\sum_{i=1}^{n}\sum_{i'=1}^{n}\alpha_i\alpha_{i'}y_iy_{i'}\underbrace{x_i^Tx_{i'}}_{\langle x_i,x_{i'}\rangle}.$$

and the solution function (decision boundary) is

$$f(x)=x^T\beta+\beta_0=\sum_{i=1}^{n}\alpha_iy_i\langle x,x_i\rangle+\beta_0.$$

➔ α_i can be solved numerically, and β_0 can be found by solving $y_if(x_i)=1$ for all x_i for which $0<\alpha_i<\gamma$.

Support Vector Machines (1/4)

- By considering the enlarged feature space

$$\phi(x_i) \triangleq [\phi_1(x_i), \ldots, \phi_m(x_i)]^T$$

where m may be large, the Lagrange dual function is

$$L_D = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{i'=1}^{n} \alpha_i \alpha_{i'} y_i y_{i'} \langle \phi(x_i), \phi(x_{i'}) \rangle$$

and the solution function can be written as

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

➔We are only concerned with the kernel

$$K(x, x') = \langle \phi(x), \phi(x') \rangle$$

not the actual enlarged feature space $\phi(x)$.

Support Vector Machines (3/4)

- SVM replaces the inner product $\langle x_i, x_{i'} \rangle$ with a more general kernel function $K(x_i, x_{i'})$, which should be a symmetric positive (semi-)definite function.
- Examples:
  – linear kernel: $K(x_i, x_{i'}) = \langle x_i, x_{i'} \rangle$
  – d-degree polynomial kernel:

$$K(x_i, x_{i'}) = (1 + \langle x_i, x_{i'} \rangle)^d$$

  – radial kernel: $K(x_i, x_{i'}) = \exp(-\|x_i - x_{i'}\|^2 / c)$
- The solution function can be written as

$$f(x) = \sum_{i=1}^{n} \alpha_i y_i K(x, x_i) + \beta_0.$$

Support Vector Machines (4/4)

- For example, with two input features x_{i1} and x_{i2}, a polynomial kernel of degree 2 yields

$$K(x_i, x_{i'}) = (1 + \langle x_i, x_{i'} \rangle)^2$$
$$= 1 + 2x_{i1}x_{i'1} + 2x_{i2}x_{i'2} + (x_{i1}x_{i'1})^2$$
$$+ (x_{i2}x_{i'2})^2 + 2x_{i1}x_{i'1}x_{i2}x_{i'2}.$$

This is equivalent to considering the 6 input features

$$\phi(x_i) \triangleq [1, \sqrt{2}x_{i1}, \sqrt{2}x_{i2}, (x_{i1})^2, (x_{i2})^2, \sqrt{2}x_{i1}x_{i2}]^T$$

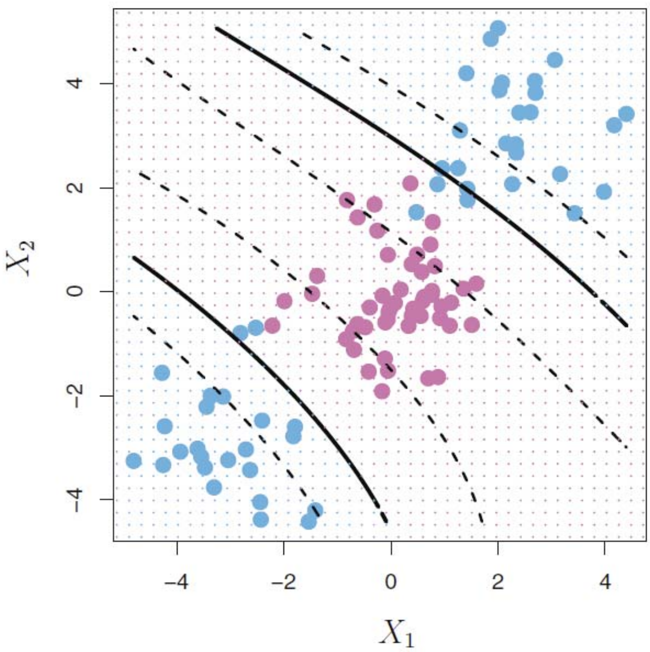
and view $K(x_i, x_{i'})$ as a linear kernel on the enlarged feature space $\phi(x_i)$, i.e.,

$$K(x_i, x_{i'}) = \langle \phi(x_i), \phi(x_{i'}) \rangle.$$

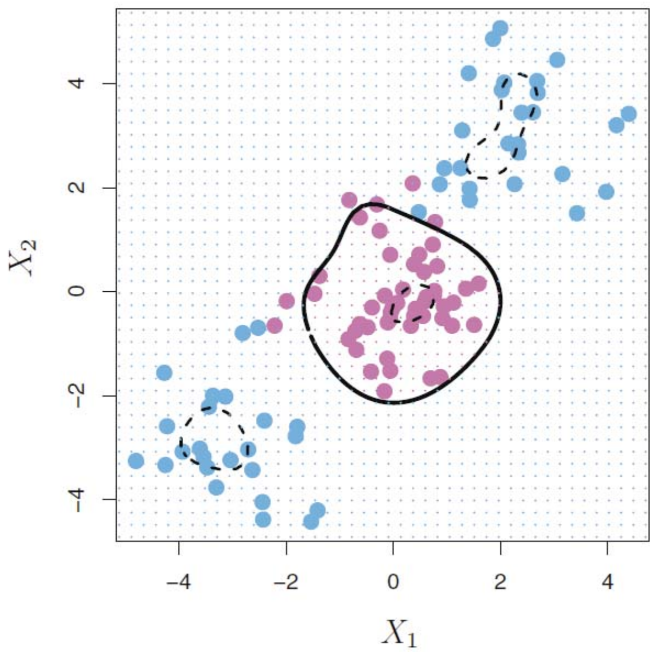
Polynomial vs Radial Kernel

Example:

Polynomial kernel of degree 3



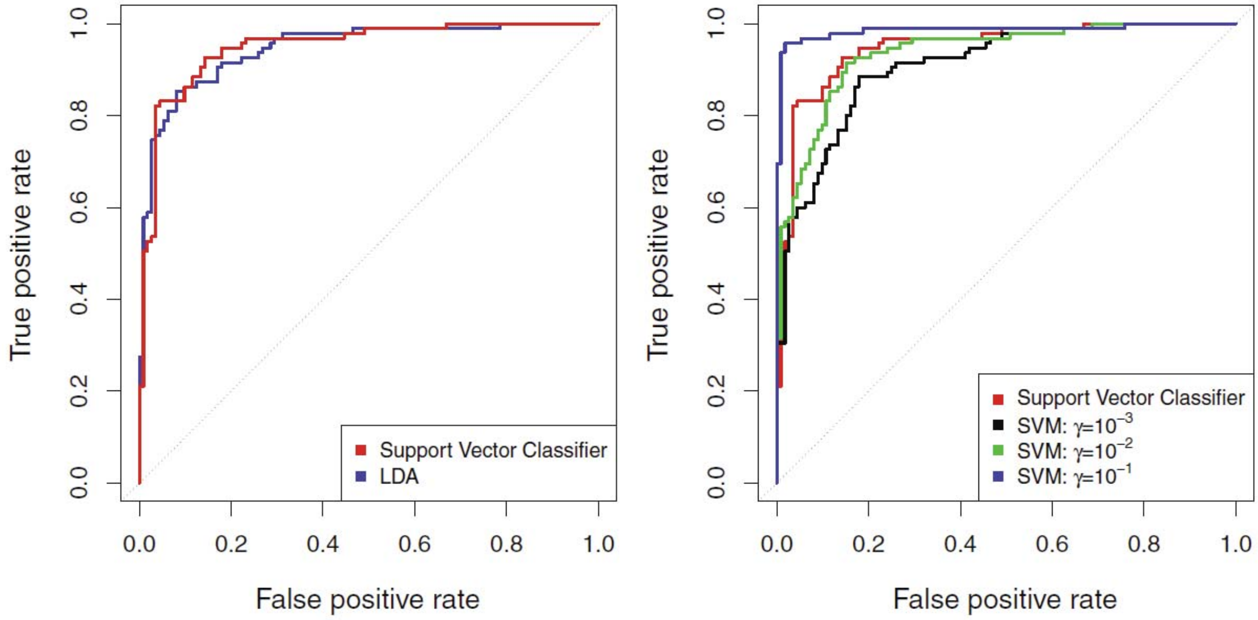
Radial kernel



Example (1/2)

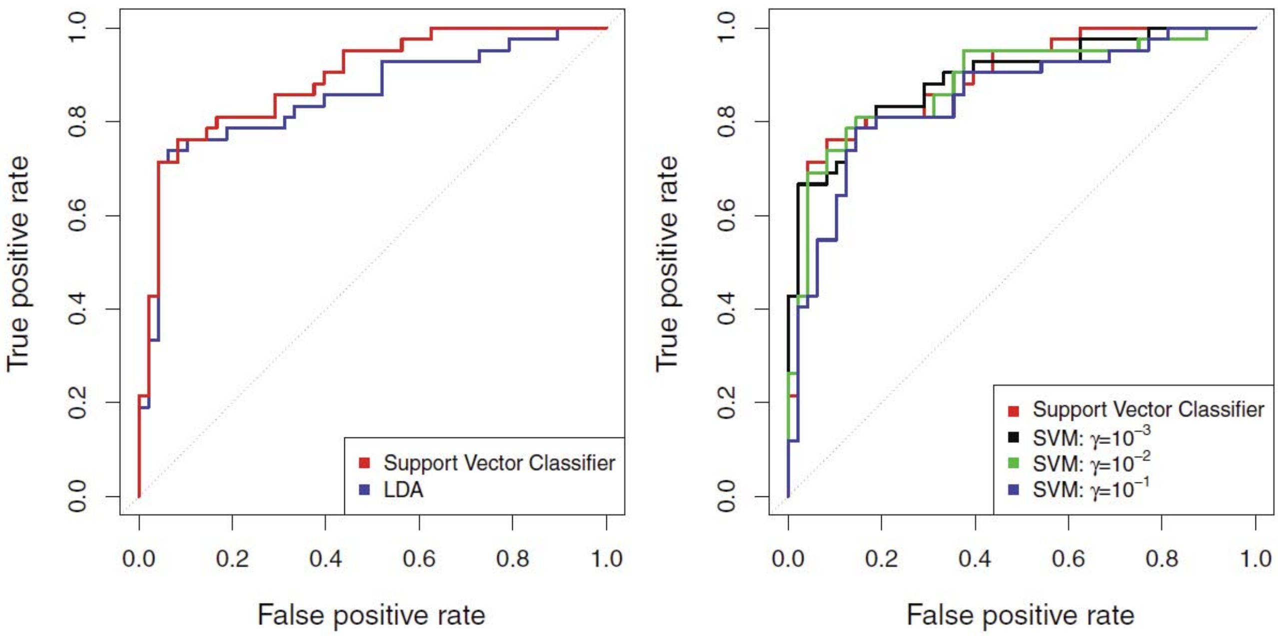
Example (Heart disease data set):

- 13 predictors (e.g., Age, Sex, Chol etc) to predict heart disease.
- 297 subjects: 207 for training and 90 for testing.



Example (2/2)

- Test error results



SVMs with More than Two Classes

- **Method 1:** One-Versus-One Classification
  - Suppose that there are K classes.
  - Construct $\binom{K}{2}$ SVMs, each comparing a pair of classes.
  - Final classification made by majority vote.
- **Method 2:** One-Versus-All Classification
  - Fit K SVMs, each comparing one of the K classes to the remaining $K-1$ classes.

    ➔ The resulting solution for class k is

$$f_k(x) = \beta_{0k} + \beta_{1k}x_1 + \cdots + \beta_{pk}x_p.$$

  - Final classification is chosen as

$$k^* = \arg\max_k f_k(x).$$

Loss + Penalty Formulation

- It can be shown that the problems on slide 14 can be equivalently reformulated as

$$\min_{\beta_0',\beta} \sum_{i=1}^{n} \max\{0, 1 - y_i f(x_i)\} + \lambda\|\beta\|^2$$

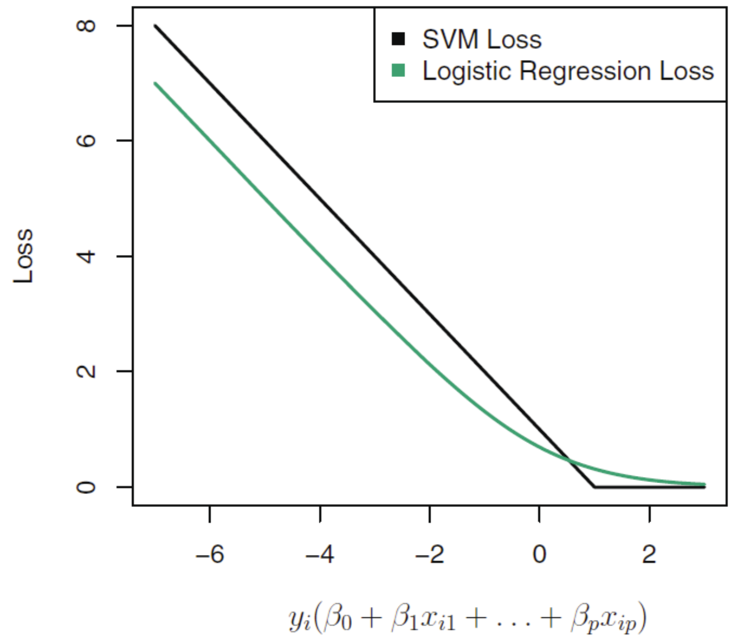
with $\lambda = 1/(2\gamma)$. (I.e., the loss + penalty form.)

  - λ large: more margin violations, less variance, higher bias.
  - λ small: fewer violations, higher variance, low bias.
- By using the hinge loss, only points x_i such that $y_i(\beta_0 + \beta^T x_i) < 1$ (i.e., points on the wrong side of the margin) affect the objective value.

SVM versus Logistic Regression

- Logistic regression aims to minimize loss function

$$-\log \ell(\beta) = \sum_{i=1}^{n} \log\left(1 + e^{-y_i(\beta_0 + \beta^T x_i)}\right), \text{ for } y_i \in \{1, -1\}.$$



SVM for Regression

- With the linear regression model $f(x) = x^T\beta + \beta_0$, we seek to minimize

$$\sum_{i=1}^{n} V_\epsilon(y_i - x_i^T\beta - \beta_0) + \frac{\lambda}{2}\|\beta\|^2$$

where the support vector error measure is defined as

$$V_\epsilon(r) = \begin{cases} 0, & \text{if } |r| < \epsilon, \\ |r| - \epsilon, & \text{otherwise.} \end{cases}$$

- The solution can be shown to have the form

$$\hat{f}(x) = \sum_{i=1}^{n} (\hat{\alpha}_i^* - \hat{\alpha}_i)\langle x, x_i\rangle + \beta_0,$$

where $\hat{\alpha}_i^*, \hat{\alpha}_i$ are solutions to the problem

SVM for Regression

$$\min_{\alpha_i,\alpha_i^*} \epsilon \sum_{i=1}^{n}(\alpha_i^* + \alpha_i) - \sum_{i=1}^{n} y_i(\alpha_i^* - \alpha_i)$$

$$+ \frac{1}{2}\sum_{i=1}^{n}\sum_{i'=1}^{n}(\alpha_i^* - \alpha_i)(\alpha_{i'}^* - \alpha_{i'})\langle x_i, x_{i'}\rangle$$

$$\text{subject to } 0 \le \alpha_i, \alpha_i^* \le \frac{1}{\lambda}, \ \sum_{i=1}^{n}(\alpha_i^* - \alpha_i) = 0, \ \alpha_i\alpha_i^* = 0.$$

➔ $\hat{\alpha}_i^* - \hat{\alpha}_i$ are nonzero only for a subset of observations (i.e., support vectors).

➔ The solution depends only on inner product $\langle x_i, x_{i'}\rangle$.

- SVM for regression replaces $\langle x_i, x_{i'}\rangle$ with $K(x_i, x_{i'})$.