STA 5820
Chapter 9
Support Vector Machines

Kazuhiko Shinki

Wayne State University

9.1 Maximal Margin Classifier

9.1.1 What is a hyperplane?

In a $\boldsymbol{p}$-dimensional space ($\boldsymbol{p} \in \mathbb{N}$), a flat affine subspace of dimension $\boldsymbol{p} - \mathbf{1}$ is called a hyperplane.

- If $\boldsymbol{p} = \mathbf{1}$, any point is a hyperplane.
- If $\boldsymbol{p} = \mathbf{2}$, any straight line is a hyperplane.
- If $\boldsymbol{p} = \mathbf{3}$, any flat plane is a hyperplane.
- If $\boldsymbol{p} = \mathbf{4}$, any 3-dimentional *flat* space is a hyperplane.
- ....

Mathematically, a hyperplane is defined by

$$\beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p = 0 \tag{1}$$

where $\boldsymbol{X} = (X_1, \cdots, X_p)^T$ is a point in $\mathbb{R}^p$.

9.1 Maximal Margin Classifier

9.1.2 Classification Using a separating hyperplane

Suppose that there are $\boldsymbol{n}$ observations in a $\boldsymbol{p}$-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 each observation $\boldsymbol{x_i}$ has a binary label $\boldsymbol{y_i \in \{-1, 1\}}$.

9.1 Maximal Margin Classifier

If there is a hyperplane (1) which perfectly separates the cases with $\boldsymbol{y_i = -1}$ and the cases with $\boldsymbol{y_i = 1}$, we can write

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

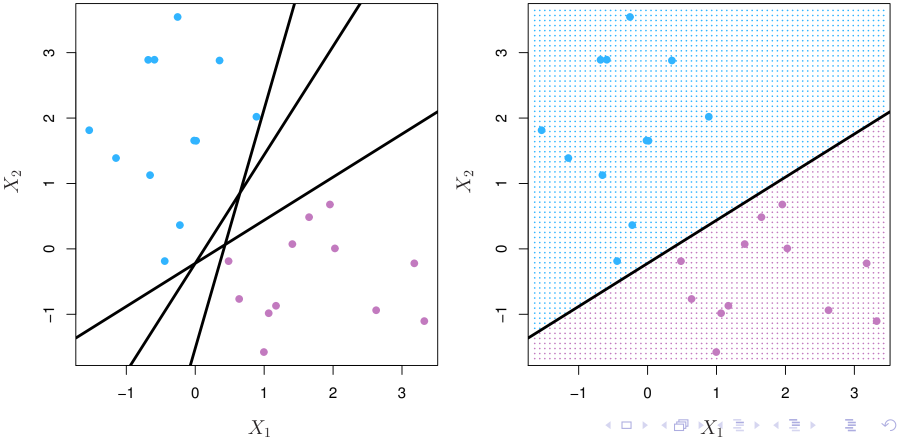
$$\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \ldots + \beta_p x_{ip} < 0 \text{ if } y_i = -1.$$

We can also write

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

9.1 Maximal Margin Classifier

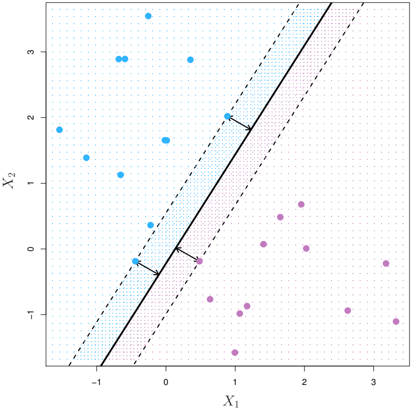
Figure 9-2: When observations are separable by a hyperplane in the feature space, the separating hyperplane is usually not unique. (Note: $p = 2$ in this figure for visualization purpose, but p is very large in practice!)



9.1 Maximal Margin Classifier

9.1.3 Maximal margin classifier

When the cases are separable, the separating hyperplane that is farthest from the training observations is called the maximal margin classifier or the optimal separating hyperplane. (Figure 9.3).



9.1 Maximal Margin Classifier

The observations closest to the maximal margin classifier are called support vectors.

(In separable cases, there are usually $p + 1$ support vectors.)

9.1 Maximal Margin Classifier

9.1.4 Construction of the Maximal Margin Classifier

Mathematically, the maximal margin classifier is obtained by

$$\begin{aligned}
&\underset{\beta_0,\beta_1,\ldots,\beta_p,M}{\text{maximize}} \quad M \\
&\text{subject to } \sum_{j=1}^{p} \beta_j^2 = 1, \\
&y_i(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \ldots + \beta_p x_{ip}) \geq M \quad \forall\, i = 1, \ldots, n.
\end{aligned}$$

As $\boldsymbol{\beta_0 + \beta_1 x_{i,1} + \cdots + \beta_p x_{i,p}}$ is the signed distance between the separating hyperplane (1) and the observation $\boldsymbol{x_i}$. $\boldsymbol{M}$ is positive, and is equal to the minimum margin between the hyperplane and the observations.

9.1 Maximal Margin Classifier

Figure: A review on geometric interpretation of inner products

9.1 Maximal Margin Classifier

Figure: Distance between the hyperplane and the observation

9.1 Maximal Margin Classifier

9.1.5 The non-separable case

For non-separable cases, a few observations are allowed to go beyond the dashed lines (cf. Figure 9.3) with some penalty.

The maximal margin classifier for this non-separable case is called the support vector classifier.

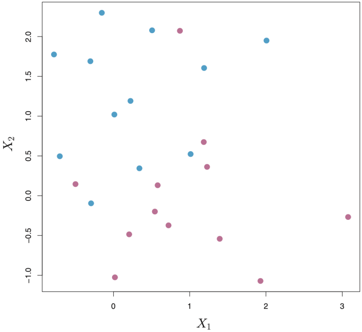


Figure 9.4: non-separable cases

9.2 Support Vector Classifiers

9.2.1 An overview of the Support Vector Classifier

The support vector classifier (also called soft margin classifier is robust against randomness of observations. Namely,

- The maximal margin rule makes the test performance better.
- As a few obsevations are allowed to go beyond the dashed lines (cf (1), the hyperplane is not too sensitive to a few observations.

9.2 Support Vector Classifiers

Robustness of Support Vector classifier

9.2 Support Vector Classifiers

9.2.2 Details of the Support Vector Classifier

Mathematically, the support vector classifier is determined by the set of following conditions:

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

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

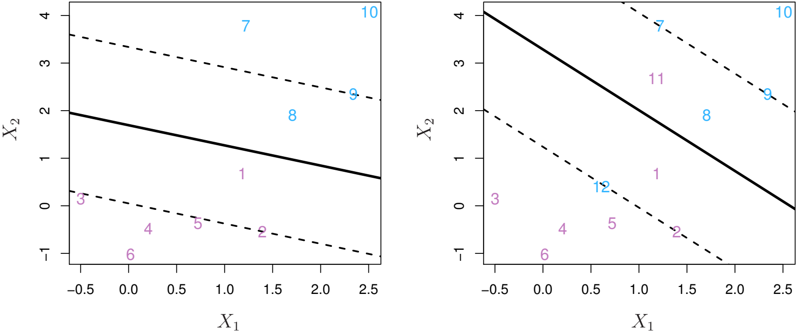
$$y_i(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \ldots + \beta_p x_{ip}) \geq M(1 - \epsilon_i),$$

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

where $C > 0$ is a tuning parameter. $\epsilon_i > 0$ only when the observation goes beyond the dashed lines in Figure 9.3, and represent how far it violates the classification rule.

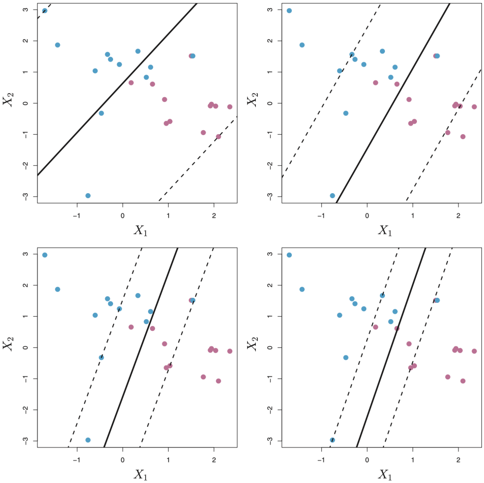
9.2 Support Vector Classifiers

Figure 9-6: Observations may not be separable by a hyper-plane in the feature space. We allow small classification errors in such a case.



9.2 Support Vector Classifiers

Figure 9-7: Support vector classifier for different ***C***'s.



9.3.1 Classification with non-linear decision boundaries

To make complicated boundaries linearly separable, map all observations into a higher dimensional space (called a feature space).

An example is:

$$\phi : (X_1, X_2) \to (X_1, X_1^2, X_2, X_2^2, X_1X_2). \tag{2}$$

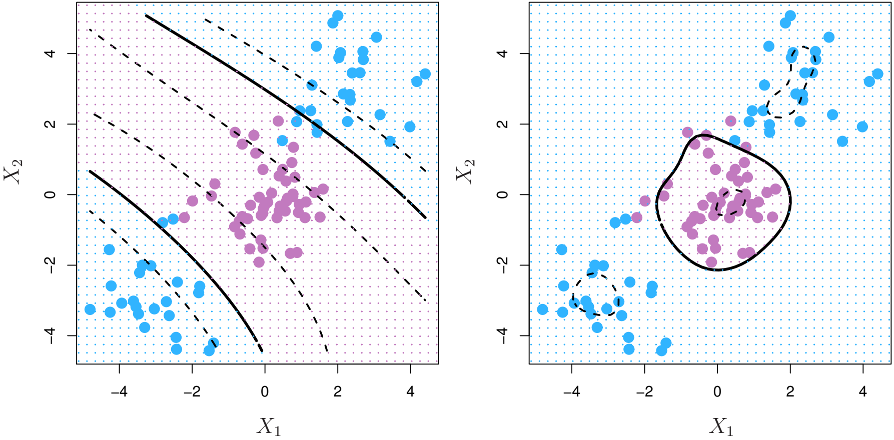
In the end set of the feature space, a support vector classifier is applied.

9.3 Support Vector Machines

Idea of classification with non-linear decision boundaries

9. Support Vector Machines

Figure 9-9: SVM estimates non-linear decision boundaries (in the original space of predictors).



9.3 Support Vector Machines

Feature maps and kernels

In the actual algorithm of SVM, $\boldsymbol{\phi}$ is not explicitly defined. Instead, a kernel $\boldsymbol{K} : \mathbb{R}^{\boldsymbol{p}} \times \mathbb{R}^{\boldsymbol{p}} \rightarrow \mathbb{R}$ such that

$$\boldsymbol{K}(\boldsymbol{x}, \tilde{\boldsymbol{x}}) = \boldsymbol{\phi}(\boldsymbol{x})' \boldsymbol{\phi}(\tilde{\boldsymbol{x}})$$

is defined, and all estimations are done through $\boldsymbol{K}$.

The inner product determines the distance between any two observations, and it is sufficient to classify observations. (Imagine that there are 50 observations in an infinite dimensional feature space. We do not need the information of the entire feature space.)

9.3 Support Vector Machines

Example 1 - Linear kernel

If ϕ is an identity map (i.e., the original space and the feature space are the same), the kernel is

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

The decision boundary is given in the form of

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

Once the boundary and the margin are obtained, the decision boundary only depends on support vectors. It is known that we can write

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

where $\mathcal{S}$ is the set of support vectors.

9.3 Support Vector Machines

Generally, the inner product $< \boldsymbol{x_i}, \boldsymbol{x_{i'}} >$ is replaced by an inner product $\boldsymbol{K(x_i, x_{i'})}$ in the feature space. ($\boldsymbol{K}$ may not be an inner product in the original space.)

Recall that $\boldsymbol{K}$ is an inner product in $\boldsymbol{V}$ if for any $\boldsymbol{x, y, z \in V}$ and $\boldsymbol{a} \in \mathbb{R}$,

$$\begin{aligned} \boldsymbol{K(ax, y)} &= \boldsymbol{aK(x, y)} && (3) \\ \boldsymbol{K(x + y, z)} &= \boldsymbol{K(x, z) + K(y, z)} && (4) \\ \boldsymbol{K(x, y)} &= \overline{\boldsymbol{K(y, x)}}, \text{(overline means conjugate.)} && (5) \\ \boldsymbol{K(x, x)} &> \boldsymbol{0}, \quad \text{if} \quad \boldsymbol{x \neq 0.} && (6) \end{aligned}$$

9.3 Support Vector Machines

Example 2 - Polynomial kernel

The polynomial kernel is defined by

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

This gives a feature space including up to the ***d***-th degree polynomials of the original predictors. (***d*** is a hyper-parameter.) The decision boundary can be written in the form of

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

where $\mathcal{S}$ is the set of support vectors.

Example 3 - Gaussian Kernel

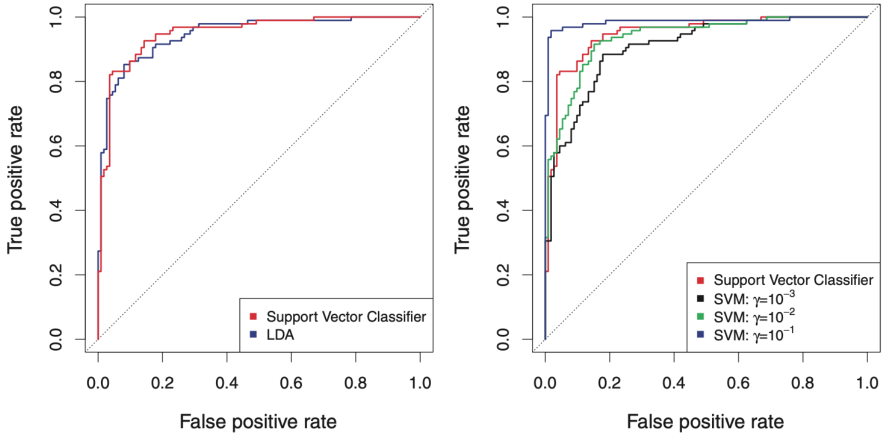
The Gaussian kernel is defined by

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

(γ is a hyper-parameter.) This gives an infinite dimensional feature space.

9.3 Support Vector Machines

Comparison of performance



9.4 SVMs for more than two classes

There are two possible approaches.

- ‘One-versus-one’ classification.
- ‘One-versus-the rest’ classification.

9.X Support vector regression

Support vector machine has a version for regression analysis, while it is not discussed in the textbook.

Suppose that $\boldsymbol{x} \in \mathbb{R}^p$ is the predictor, and $y \in \mathbb{R}$ is a response, and there are n observations $(x_1, y_1), \cdots, (x_n, y_n)$. Let $\phi : \mathbb{R}^p \to \mathbb{R}^M$ be a feature map defined in the same way as above.

Consider a model:

$$y = \beta_0 + \beta'\phi(x) + e$$

where $\beta \in \mathbb{R}^M$ and e is an error term.

9. Support Vector Machines

The support vector regression minimizes

$$\sum_{i=1}^{N} V(y_i - \beta_0 - \beta' \phi(x_i)) + \lambda \cdot ||\beta||^2$$

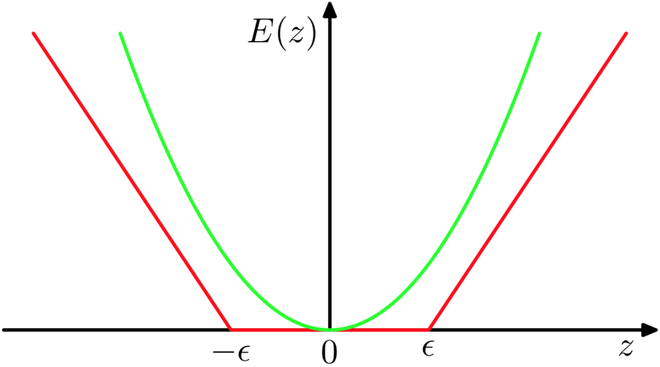
with respect to β_0, β, where $V(z)$ is an ϵ-sensitive error function:

$$\begin{aligned} V(z) &= 0 \text{ if } |z| < \epsilon \\ &= |z| - \epsilon \text{ otherwise} \end{aligned}$$

where λ and ϵ are hyperparameters.

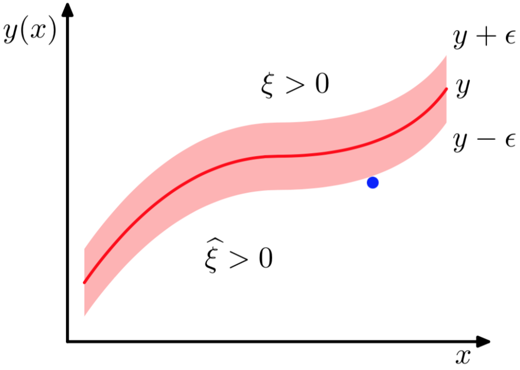
9. Support Vector Machines

$\boldsymbol{V(z)}$ is the red function below. The green curve represents the usual squared error loss, which is used in linear (least-square) regression.



The idea of Support Vector Regression

Since small errors are ignored by $\boldsymbol{V(z)}$, only a few observations determine the fitted curve, making the fitted curve relatively simpler.


$y(x)$
$y+\epsilon$
$\xi > 0$
y
$y-\epsilon$
$\widehat{\xi} > 0$
x