Support Vector Machines

Feng Li

feng.li@cufe.edu.cn

School of Statistics and Mathematics
Central University of Finance and Economics

Support Vector Machines *(SVM) method produces nonlinear boundaries by constructing a linear boundary in a large, transformed version of the feature space.*

Support vector classifiers

- consider the datasets $\{y_i, x_i\}$ for $i = 1, ..., N$ where $y_i \in \{-1, 1\}$ and x_i is a p dimensional vector.
- Define the **hyperplane**

$$\{x : f(x) = x^T\beta + \beta_0 = 0\}$$

where β is a unit vector $||\beta|| = 1$

- The classification rule

$$G(x) = sign(X^T\beta + \beta_0)$$

- Find a function $f(x) = x'\beta + \beta_0$ with $y_i f(x_i) > 0$ for all $i = 1, ..., N$. Hence we are able to find the hyperplane that creates the biggest margin (M) between the training points for class 1 and -1.

$$max_{||\beta||=1, \beta, \beta_0} M$$

subject to

$$y_i f(x_i) \geqslant M \text{ for all } i = 1, ..., N$$

The support vector classifier optimization

- The previous optimization is equivalent of

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

  subject to $y_i(x_i'\beta + \beta_0) \geqslant 1$ for all $i = 1, ..., N$
- The situation we have discussed is the non overlapping case (next figure, left)

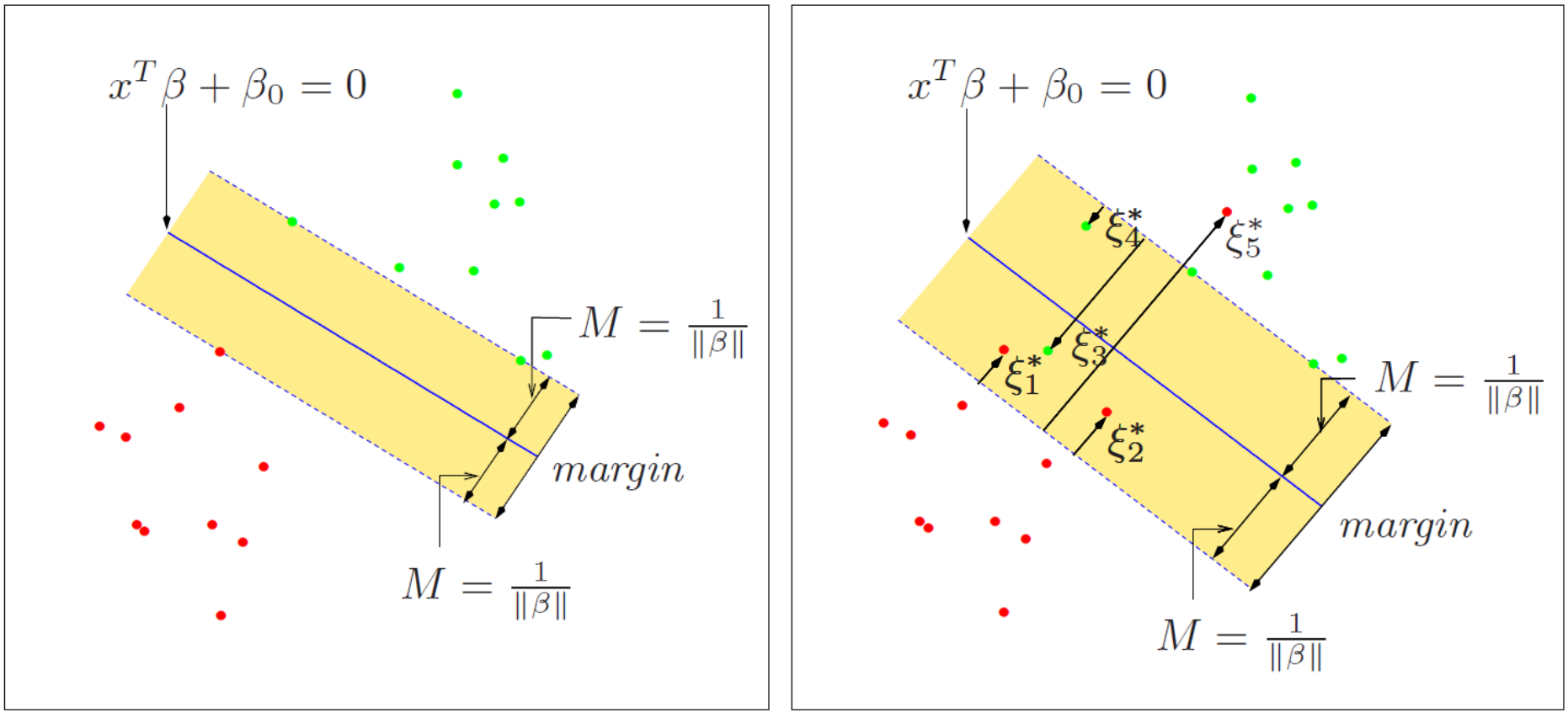


Figure: Support vector classifiers. The left panel shows the separable case. The decision boundary is the solid line. The right panel shows the nonseparable (overlap) case. The margin is maximized subject to a total budget constant.

The support vector classifier overlapped case

- Overlapping in feature space (previous figure, right)
- Still maximize M, but allow for some points to be on the wrong side of the margin.
  - Define the **slack variables** $\xi = (\xi_1, ..., \xi_N)$.
  - The constrain is now as

$$y_i(x_i'\beta + \beta_0) \geqslant M - \xi_i$$

or

$$y_i(x_i'\beta + \beta_0) \geqslant M(1 - \xi_i)$$

for all $\xi_i \geqslant 0$ and $\sum \xi_i \leqslant c$

- The two choices lead to different solutions.
  - The first measures overlap in actual distance from the margin; the second choice measures the overlap in relative distance.
  - the first choice results in a nonconvex optimization problem, while the second is convex; the second leads to the "standard" support vector classifier.
  - The second one is used from here on.

Computing the Support Vector Classifier

- The previous optimization can be re-expressed using the Lagrange multipliers

$$mim_{\beta,\beta_0}\frac{1}{2}||\beta||^2 + C\sum_{i=1}^{N}\xi_i$$

subject to $\xi_i \geqslant 0$, $y_i(x_i'\beta + \beta_0) \geqslant 1 - \xi_i$ where C is the constrain. Furthermore, when $C = \infty$, it reduces to the separable(non-overlapping) case.

- The Lagrange function is then

$$L_p = \frac{1}{2}||\beta||^2 + C\sum_{i=1}^{N}\xi_i - \sum_{i=1}^{N}\alpha_i(y_i(x_i'\beta + \beta_0) - (1 - \xi_i)) - \sum_{i=1}^{N}\mu_i\xi_i$$

- Then let $\partial L_p/\partial\beta = 0$ and solve for β which yields

$$\beta = \sum_{i=1}^{N}\alpha y_i x_i,\ \sum_{i=1}^{N}\alpha y_i = 0,\ \text{and}\ \alpha_i = C - \mu_i$$

Computing the Support Vector Classifier

- Put the solution back to the Lagrange function, we obtain the Lagrangian dual objective function

$$L_D = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{i'=1}^{N} \alpha_i \alpha_{i'} y_i y_{i'} x_i^T x_{i'}$$

  which is the lower bound of the optimized function and $x_i^T x_{i'}$ is the **inner product** of x_i and $x_{i'}$, denoting as $< x_i, x_{i'} >$
- Now maximize L_D with the constrain $0 \leqslant \alpha_i \leqslant C$ and $\sum_{i=1}^{N} \alpha_i y_i = 0$

  $\hat{\alpha}_i = C$

  $\hat{\beta} = \sum_{i=1}^{N} \hat{\alpha} y_i x_i \hat{\alpha}_i = C$

  $\hat{\beta}_0$ is from the solution of $\alpha_i(y_i(x_i'\beta + \beta_0) - (1 - \xi_i)) = 0$
- Those observations that meet the constrain

$$y_i(x_i'\beta + \beta_0) - (1 - \xi_i) \geqslant 0$$

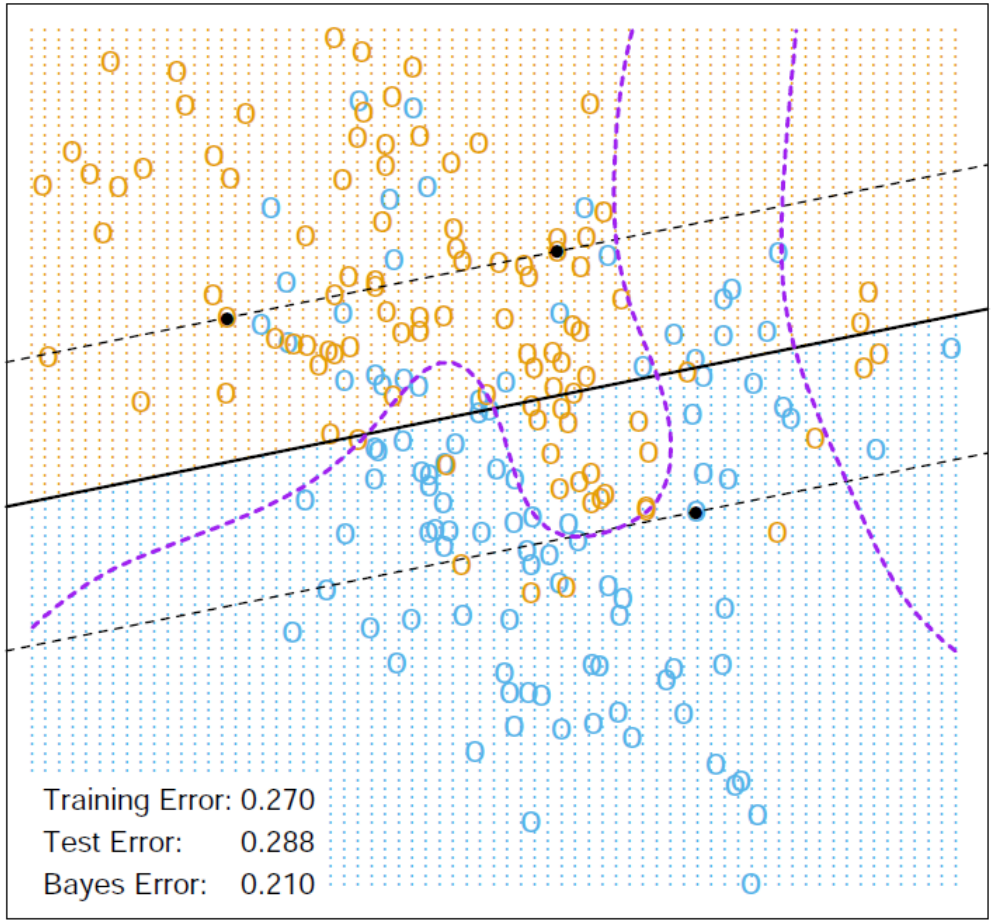
  are called the **support vectors**.

The Support Vector Classifier decision function

- The decision function

$$\hat{G}(x) = sing(\hat{f}(x)) = sign(x'\hat{\beta} + \hat{\beta}_0)$$

- $\hat{G}(x)$ depends on the tuning parameter C.



$C = 10000$

Figure: The **linear support vector boundary** for the mixture data exam- ple with two overlapping classes. The broken lines indicate the margins, where $f(x) = \pm 1$. The support points ($\xi > 0$) are all the points on the wrong side of their margin. The black solid dots are those support points falling exactly on the margin ($\xi = 0$, $\alpha_i > 0$). The broken purple curve in the background is the Bayes decision boundary.

The Support Vector Machines

- We have discussed finding linear boundaries in the input feature space.
- We can make the procedure more flexible by enlarging the feature space using basis expansions such as polynomials or splines
- The **support vector machine** classifier allows the dimension of the enlarged space is allowed to get very large, infinite in some cases.
  - The computations would become prohibitive.
  - With sufficient basis functions, the data would be separable, and overfitting would occur.

Compute the SVM for Classification

- We now replace the feature x_i with nonlinear function $h(x_i)$. Then we have $\hat{f}(x) = h(x)^T\hat{\beta} + \hat{\beta}_0$.
- The Lagrange dual function now has the form

$$L_D = \sum_{i=1}^{N} \alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{i'=1}^{N} \alpha_i\alpha_{i'}y_iy_{i'} < h(x_i), h(x_{i'}) >$$

- If we let $K(x, x') =< h(x_i), h(x_{i'}) >$, we may select "good" $K(x, x')$ that finds the best boundary. Remember K should be symmetric and semi-positive definite. Some popular choices

$$K(x, x') = (1+ < x, x' >)^d$$
$$K(x, x') = exp(-\gamma||x - x'||^2)$$
$$K(x, x') = tanh(\kappa_1 < x, x' > +\kappa_2)$$

- There is still a tuning parameter C. A large value of C will discourage any positive ξ, and lead to an overfit wiggly boundary in the original feature space; a small value of C will encourage a small value of $||\beta||$, which in turn causes $f(x)$ and hence the boundary to be smoother.

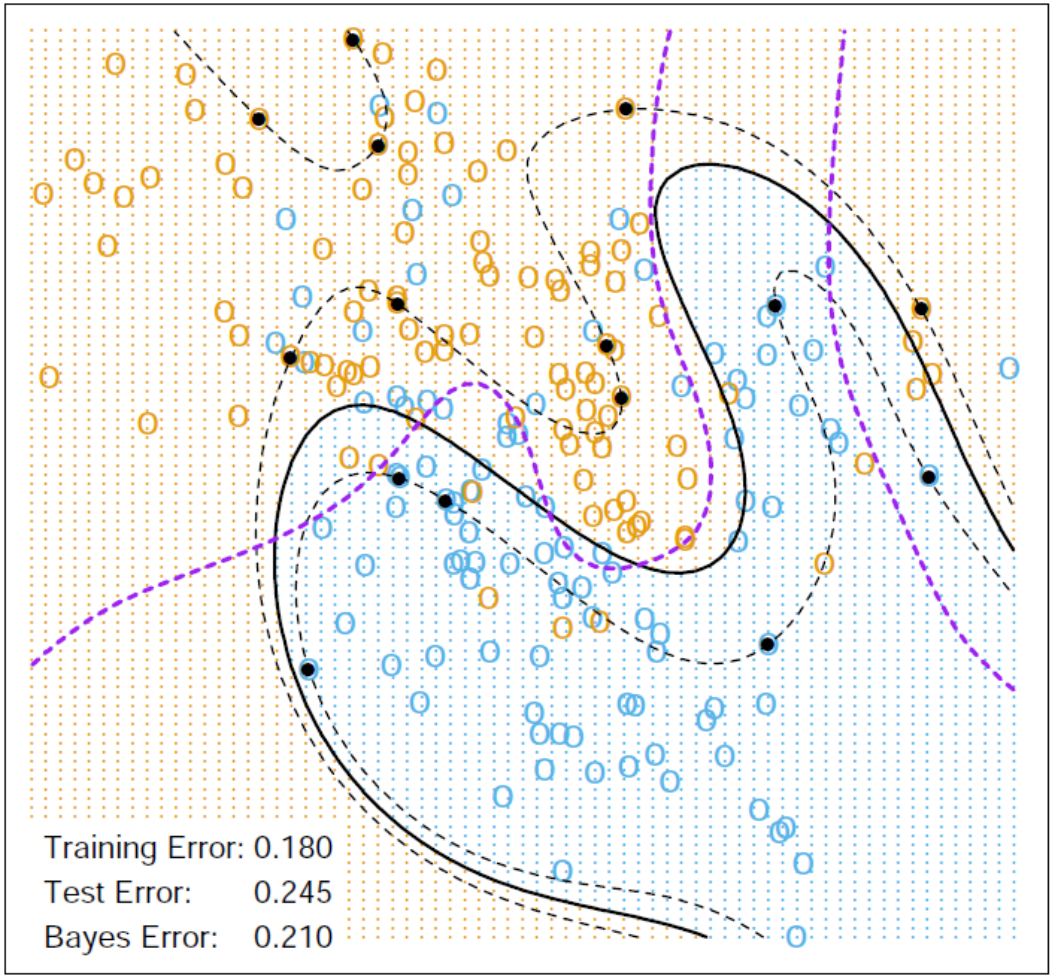


Figure: The plot uses a 4th degree polynomial kernel. C was tuned to approximately achieve the best test error performance. The radial basis kernel performs the best (close to Bayes optimal). The broken purple curve in the background is the Bayes decision boundary.

SVM for regression

- Consider the linear regression model

$$f(x) = x'\beta + \beta_0$$

and we estimate β by minimizing

$$H(\beta, b_0) = \sum_{i=1}^{N} V(y_i - f(x_i)) + \frac{\lambda}{2}||\beta||^2$$

where

$$V(y_i - f(x_i)) = \begin{cases} 0 & \text{if } |(y_i - f(x_i))| < \epsilon \\ |(y_i - f(x_i))| - \epsilon, & \text{otherwise} \end{cases}$$

is the **support vector error**

- There is a rough analogy with the support vector classification setup, where points on the correct side of the decision boundary and far away from it, are ignored in the optimization. In regression, these "low error" points are the ones with small residuals.

SVM for regression

- Use similar type of error

$$V(y_i - f(x_i)) = \begin{cases} (y_i - f(x_i))^2 & \text{if } |(y_i - f(x_i))| < \epsilon \\ c|(y_i - f(x_i))| - c^2/2, & \text{otherwise} \end{cases}$$

- Some properties
  - This construction makes the fitting less sensitive to outliers
  - The support vector error measure (12.37) also has linear tails, but in addition it flattens the contributions of those cases with small residuals.
- The solution

$$\hat{\beta} = \sum_{i=1}^{N} (\hat{\alpha}_i^* - \hat{\alpha}_i) x_i$$

where $\hat{\alpha}_i^*$ and $\hat{\alpha}_i$ from the solution from the minimizing the Lagrange function.

SVM for regression

- We can approximate the regression in terms of basis functions $h(x)$

$$f(x) = \sum_{m=1}^{M} \beta_m h_m(x) + \beta_0$$

- We minimize

$$H(\beta, \beta_0) = \sum_{i=1}^{N} V(y_i - f(x_i)) + \frac{\lambda}{2} \sum \beta_m^2$$

- We can use the support vector errors $V()$ here too.
- Special case when $V(y_i - f(x_i)) = (y_i - f(x_i))^2$
  - we minimize

$$H(\beta) = (y - H\beta)'(y - H\beta) + \lambda ||\beta||^2$$

  - The solution is

$$\hat{y} = H\hat{\beta}$$

SVM in R

The e1071 package