Chapter 4. Classification methods
Part 2

March 26, 2007

1 Examples of using linear regression in classification

Suppose we have two classes: A and B, with indicators (covariates) $X = (X_1, ..., X_p)$. We need to estimate a separating hyperplane

$$f(X) = \beta^\top X + c.$$

and classify x_{new} to A if

$$f(x_{new}) = \beta^\top x_{new} + c > 0$$

to B if

$$f(x_{new}) = \beta^\top x_{new} + c <= 0$$

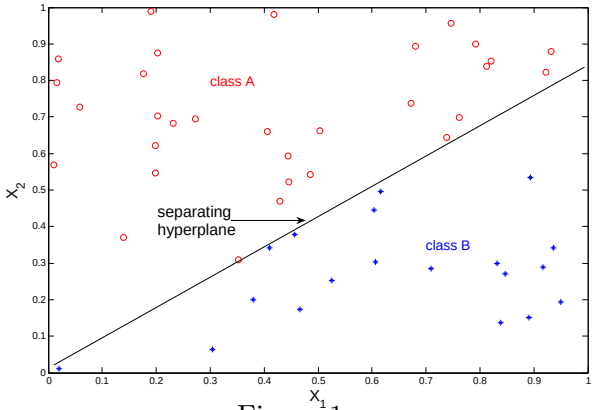

Figure 1:

Example 1.1 *For data (***(training data)***. There are two variables* X_1 *and* X_1*, the first 22 observations are from* A *and the others* B*. By assigning value* $Y = 0$ *to observations in* A *and* $Y = 1$ *to those in* B*. We generate a nominal linear regression model*

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \varepsilon_i$$

The estimated model is

$$\hat{Y} = 0.4017 - 0.7883 * X_1 + 1.0227 X_2$$

(how to implement it?), we need to select c *and classify*

$$\textit{observation } i \in A \textit{ if } \hat{Y}_i - c < 0$$

$$\textit{observation } i \in B \textit{ if } \hat{Y}_i - c > 0$$

i.e.

$$c = \min_c \sum_{i=1}^{n} \{Y_i - I(\hat{Y}_i - c > 0)\}^2$$

where $I(.)$ *is the indicator function. In other words the separating hyperplane is*

$$f(X) = 0.4017 - 0.7883 * X_1 + 1.0227 X_2 - c$$

If we assign value $Y = -1$ *to observations in* A *and* $Y = 1$ *to those in* B*. The estimated model is*

$$\hat{Y} = -0.1965 - 1.5767 * X_1 + 2.0454 X_2$$

we can simply use

$$f(X) = -0.1965 - 1.5767 * X_1 + 2.0454 X_2$$

as the separating hyperplane (why?). The separating hyperplane is shown in figure 2

Example 1.2 (cell classification based on gene) *For the leukemia gene expression data (***(training data)***. There are 38 cells with 250 genes (selected from about 7000 genes). they are from two types of cells.*

*To check the models, a new experiment was done and the data were collected (***(testing set)***.*

with ridge parameter $\lambda = 1/n$*, we can construct a separating hyperplane easily.*

R code for the calculation **(code)**

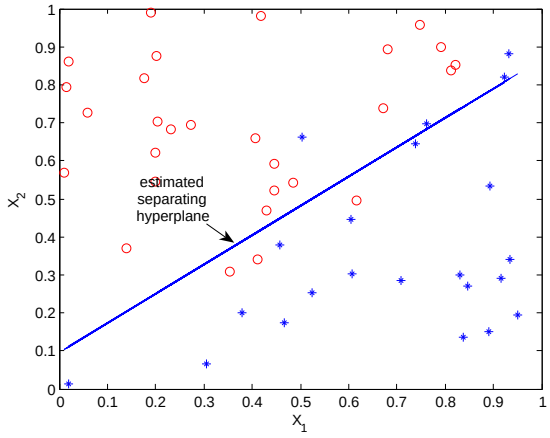

Figure 2: calculation results for Example 1.1

2 Support vector machines

Suppose each sample $X = (\mathbf{x}_1, ..., \mathbf{x}_p)$ belongs to one of two classes, denoted by -1 and 1 respectively. In other words, we have (X, Y) with response variable Y takes values -1 (class A) and 1 (class B). Suppose we have samples (training set) $(X_i, Y_i), i = 1, ..., n$. Define a separating hyperplane by

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

where β is a unit vector: $\|\beta\| = 1$. A classification rule induced by $f(x)$ is

$$sign[x^\top \beta + \beta_0]$$

Since the classes are *separable* by a hyperplane, if X_i is from class A, then $f(X_i) < 0$ and $Y_i f(X_i) > 0$; if X_i is from class B, then $f(X_i) > 0$ and $Y_i f(X_i) > 0$. Therefore the separating plane is estimated to create the biggest margin between the training points for class 1 and -1. In mathematics, we need to estimate β and β_0 by optimizing

$$\begin{aligned} &\max_{\beta_0, \beta: \|\beta\|=1} m \\ \text{subject to} \quad &Y_i(X_i^\top \beta + \beta_0) \geq m, \ i = 1, ..., n \end{aligned}$$

figure 3 shows the interpretation of $m > 0$.

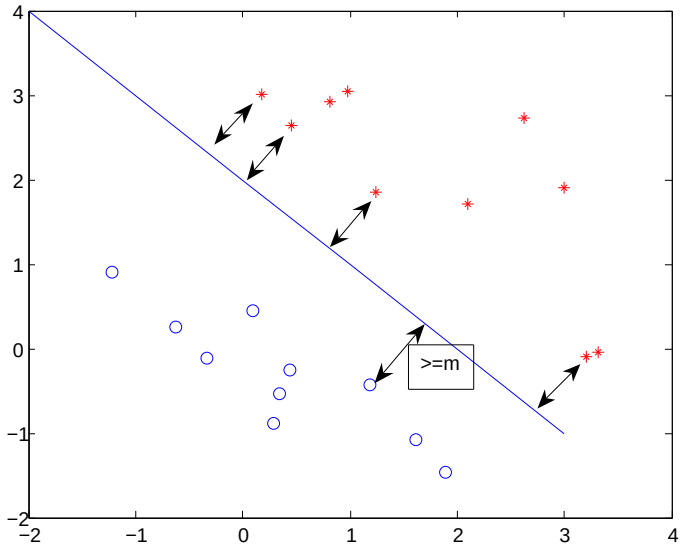

Figure 3: An example to show the separating hyperplane and the margin C

If the classes are *not separable*, or they have overlap (statisticians like this assumption), one way to deal with the overlap is to still maximize m, but allow some points to be on the wrong side of the margin. define the slack variables $\xi = (\xi_1, ..., \xi_n)$. One natural way to modify the optimization problem

$$\begin{aligned} & \max_{\beta_0, \beta: \|\beta\|=1} m \\ \text{subject to} \quad & Y_i(X_i^\top \beta + \beta_0) \geq m(1 - \xi_i), \ i = 1, ..., n \\ & \sum_{i=1}^{n} \xi_i < C, \ \xi_i \geq 0, i = 1, ..., n \end{aligned}$$

where C is a constant.

Note that if $\xi_i > 1$ then, the sample is misclassified. Thus, C means we can have at most C misclassifications.

If we drop the norm constraint on β and let $m = 1/\|\beta\|$ then the above problem becomes

$$\begin{aligned} & \min \|\beta\| \\ \text{subject to} \quad & Y_i(X_i^\top \beta + \beta_0) \geq (1 - \xi_i),\ i = 1, ..., n \\ & \sum_{i=1}^{n} \xi_i < C,\ \xi_i \geq 0, i = 1, ..., n \end{aligned}$$

where C is a constant.

The above optimization can further be rephrased as

$$\begin{aligned} & \min \|\beta\| + C \sum_{i=1}^{n} \xi_i \\ \text{subject to} \quad & \xi_i \geq 0, Y_i(X_i^\top \beta + \beta_0) \geq (1 - \xi_i),\ i = 1, ..., n. \end{aligned}$$

This is a penalized optimization problem.

The calculation of β (using Lagrange multipliers) is equivalent to minimizing

$$L_P = \frac{1}{2}||\beta||^2 + C \sum_{i=1}^{n} \xi_i - \sum_{i=1}^{n} \alpha_i \{Y_i(X_i^\top \beta + \beta_0) - (1 - \xi_i)\} - \sum_{i=1}^{n} \mu_i \xi_i$$

w.r.t. β, β_0, ξ_i with constraints $\alpha_i, \mu_i, \xi_i \geq 0$, for all i.

The calculation again is equivalent to minimizing

$$L_D = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j Y_i Y_j X_i^\top X_j$$

w.r.t. α_i with constraints $0 \leq \alpha_i \leq C$ and $\sum_{i=1}^{n} \alpha_i Y_i = 0$

The solution can be written as

$$\hat{\beta} = \sum_{i=1}^{n} \hat{\alpha}_i Y_i X_i.$$

An interesting phenomenon is that most $\hat{\alpha}_i$ are zero. the nonzero coefficients α_i corresponds to those observations for which the constraints are exactly met. These observations are called the *support vectors*, since $\hat{\beta}$ is determined by them.

The connection between least square estimator and SVM estimator: Standardizing X_i, the LS estimator of β is

$$\begin{pmatrix} \hat{\beta}_{0,LS} \\ \hat{\beta}_{LS} \end{pmatrix} = \begin{pmatrix} n & 0 \\ 0 & \mathbf{X}^\top \mathbf{X} \end{pmatrix}^{-1} \sum_{i=1}^{n} \begin{pmatrix} 1 \\ X_i \end{pmatrix} Y_i$$

we have

$$\hat{\beta}_{LS} = \sum_{i=1}^{n} (\mathbf{X}^\top \mathbf{X})^{-1} Y_i X_i$$

If we investigate the problem in a feature space $\phi(x) = (\phi_1(x), ..., \phi_m(x))^\top$, then all the procedure is the same as above *with* X *replaced* $\phi(X)$. The estiamator of β can be written as

$$\hat{\beta} = \sum_{i=1}^{n} \hat{\alpha}_i Y_i \phi(X_i).$$

3 Computing SVM for classification

We need to estimate

$$\beta = \sum_{i=1}^{n} \alpha_i Y_i \phi(X_i)$$

or we need to estimate $\alpha_i, i = 1, ...n$.

Based on this, we can write our separating hyperplane as

$$f(x) = \phi(x)^\top \beta + \beta_0$$

In other words, for a new sample x, we need to check whether $f(x) > 0$ or $f(x) < 0$. Combining the two equaitons above, we have

$$\begin{aligned} f(x) &= \phi(x)^\top \beta + \beta_0 \\ &= \sum_{i=1}^{n} \alpha_i Y_i \langle \phi(x), \phi(X_i) \rangle + \beta_0, \end{aligned}$$

where $\langle \phi(x), \phi(X_i) \rangle = \phi(x)^\top \phi(X_i)$ is the inner product. More generally, we define a kernel

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

Instead of choosing ϕ (the feature space), we can change to choosing the kernel $K(x, x')$. The popular choices of K are

- linear: $x \top x'$
- polynomial: $(\gamma * x \top x' + c_0)^{degree}$
- radial basis: $\exp(-\gamma * ||x - x'||^2)$

- sigmoid (neural network): $\tanh(\gamma * x^\top x' + c_0)$

The estimation is to minimize

$$L_D = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j Y_i Y_j K(X_i, X_j)$$

w.r.t. α_i with constraints $0 \leq \alpha_i \leq C$ and $\sum_{i=1}^{n} \alpha_i Y_i = 0$

Statisticians (Wahba et al (2000)) proved that the above minimization problem is equivalent to minimizing

$$\sum_{i=1}^{n} \Big(1 - Y_i\{\beta_0 + \sum_{i=1}^{n} \alpha_i K(X_i, X_j)\}\Big)_+ + \lambda \alpha^\top \mathcal{K} \alpha$$

with respect to α, β_0, where $\mathcal{K}$ is matrix with is (i, j) entry $K(X_i, X_j)$.

This is also called kernel methods. [any connection with our NW kernel smoothing?]

4 selection of parameters in SVM

there are two parameters γ and C (corresponding to cost in R) and kernel functions. All these can be done by CV methods. however, I suggest leaving them to be default in using R package.

5 Categorical response and covariates

For categorical covariates, we need to use dummy variables. For Categorical response, we need to use "categorical variable" using function factor in R.

6 SVM can also be used for regression if the response is numerical variable

7 Examples

Example 7.1 *One application of SVM is in image reconstruction. Here is an example for this. The original photo is as shown in the first panel. 500 points are spotted with random errors as shown in panel 2. Using SVM, the picture can be recovered as shown in panel 3.*

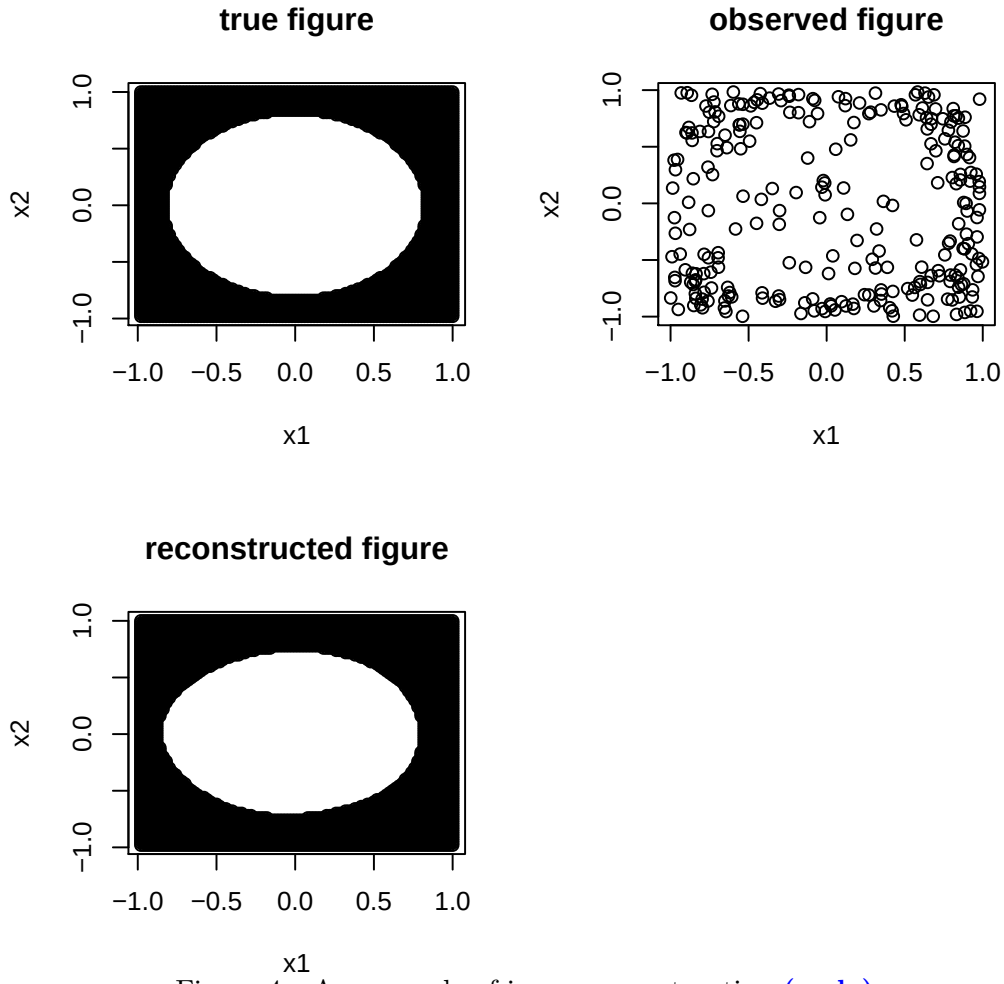

Figure 4: An example of image reconstruction **(code)**

Example 7.2 *For the heart diseases data ((training set) , (validation set)), we use CART and SVM to classify the data. The response variable is whether a man has coronary heart disease.*

We use the training data to estimate the parameters in separating plane and validation set to check the methods.

The true response in the validation sets are

0 0 1 0 0 0 1 1 1 0 0 1 0 0 1

The predicted value based on CART ((code)) is

0 0 1 0 0 0 0 0 0 0 0 0 0 0 0

The predicted value based on SVM ((code)) is

0 0 0 0 0 0 0 0 0 1 1 0 0 0 0

Example 7.3 *Speaker independent recognition of the eleven steady state vowels of British English using a specified training set of lpc derived log area ratios. ((training set) , (validation set)), we use SVM to classify the data. The response variable has 11 categories.*

There are 9 covariates $\mathbf{x}_1, ..., \mathbf{x}_9$. *we use the training data to estimate the separating plane and validation set to check the methods. The error rate for the testing set is 0.4264069 ((code))*

Suppose we use a feature space $\mathbf{x}_1, ..., \mathbf{x}_9, \mathbf{x}_1^2, ..., \mathbf{x}_9^2$. *The error rate for the testing set is 0.3961039*

If we use CART with covariate $\mathbf{x}_1, ..., \mathbf{x}_9$. *The error rate for the testing set is 0.6082251 ((code))*

References

N. Cristianini and J. Shawe-Taylor (2000) *AN INTRODUCTION TO SUPPORT VECTOR MACHINES (and other kernel-based learning methods)* Cambridge University Press 2000