HKU BUSINESS SCHOOL 港大經管學院

Business Statistics

Support Vector Machines

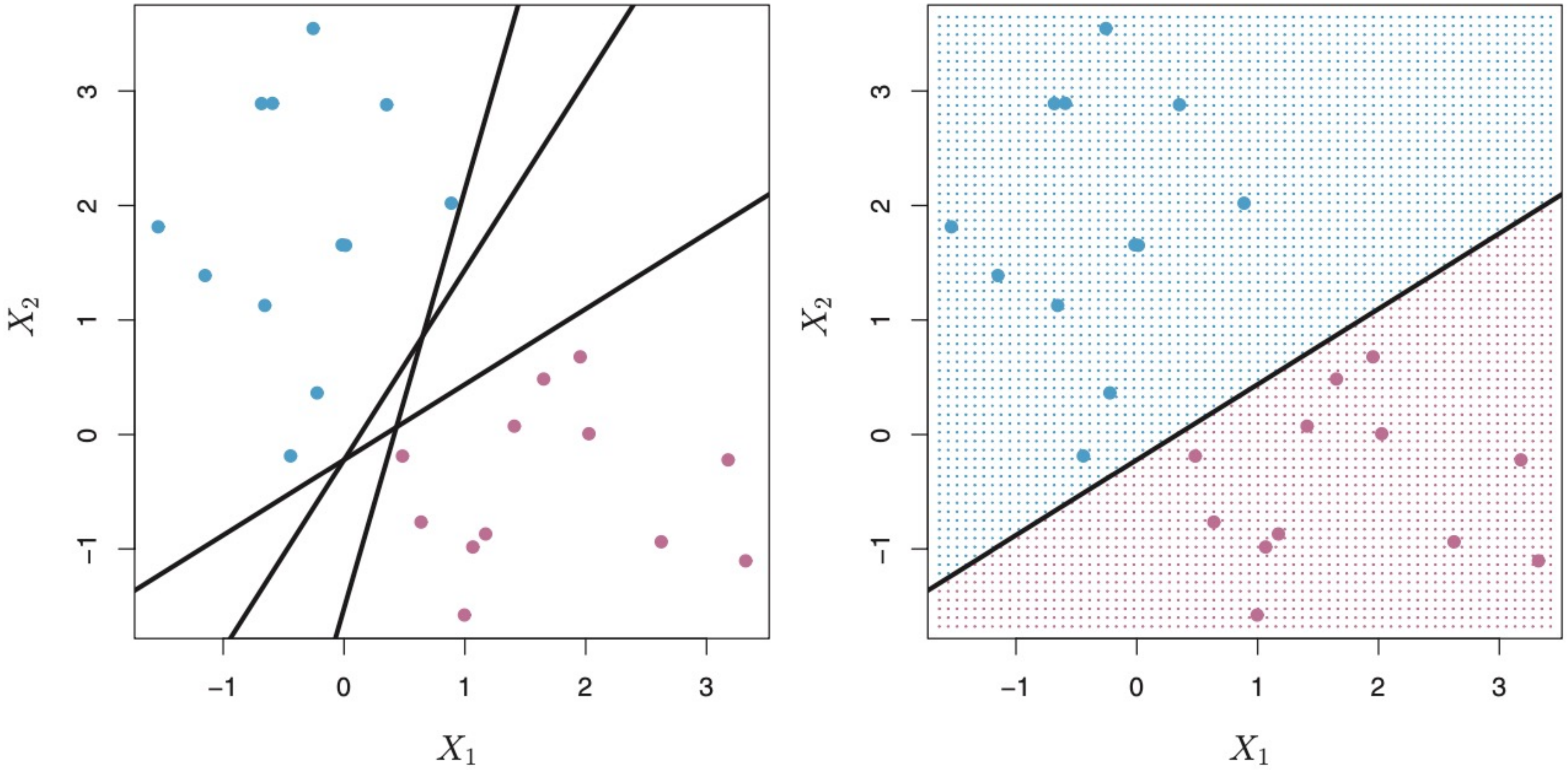
Weichen Wang

Assistant Professor
Innovation and Information Management

ISLR Chapter 9

Support Vector Machine (SVM)

- An approach for classification.
- Developed in the computer science community in the 1990s and that has grown in popularity since then.



The Maximal Margin Classifier

- If data can be perfectly separated by a linear hyperplane, choose the one with the maximal margin, i.e. the one farthest from the training observations.

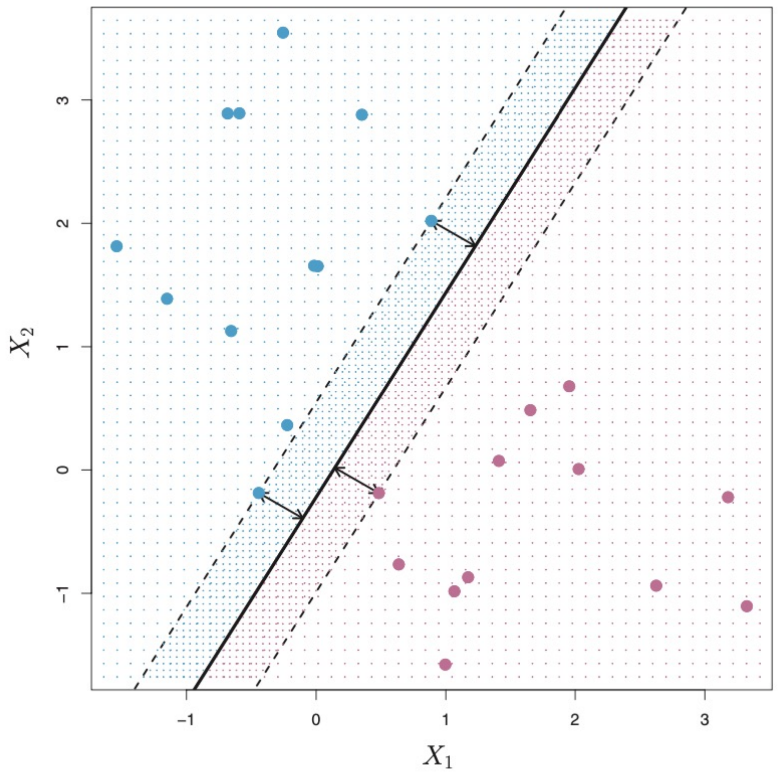
- Questions:

  What is hyperplane?

  What is margin?

  How to maximize it?

- Three “support vectors”
  They “support” the maximal margin hyperplane. If they were moved slightly, the hyperplane would move too.



Hyperplane

$\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p = 0$ defines a p-dimensional hyperplane.

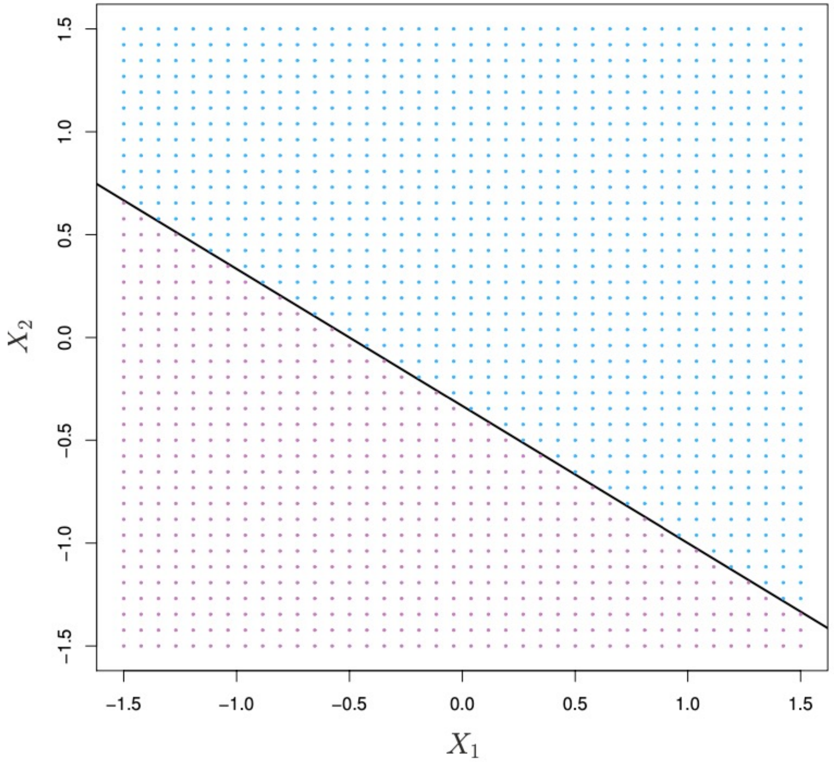


FIGURE 9.1. *The hyperplane $1 + 2X_1 + 3X_2 = 0$ is shown. The blue region is the set of points for which $1 + 2X_1 + 3X_2 > 0$, and the purple region is the set of points for which $1 + 2X_1 + 3X_2 < 0$.*

Construction of Maximal Margin Classifier

- Collect n training observations $X_1, X_2, \dots, X_n$ of p dimensions, and associated class labels $y_1, y_2, \dots, y_n \in \{-1,1\}$.

$$\underset{\beta_0,\beta_1,\dots,\beta_p}{\text{maximize}}\, M$$

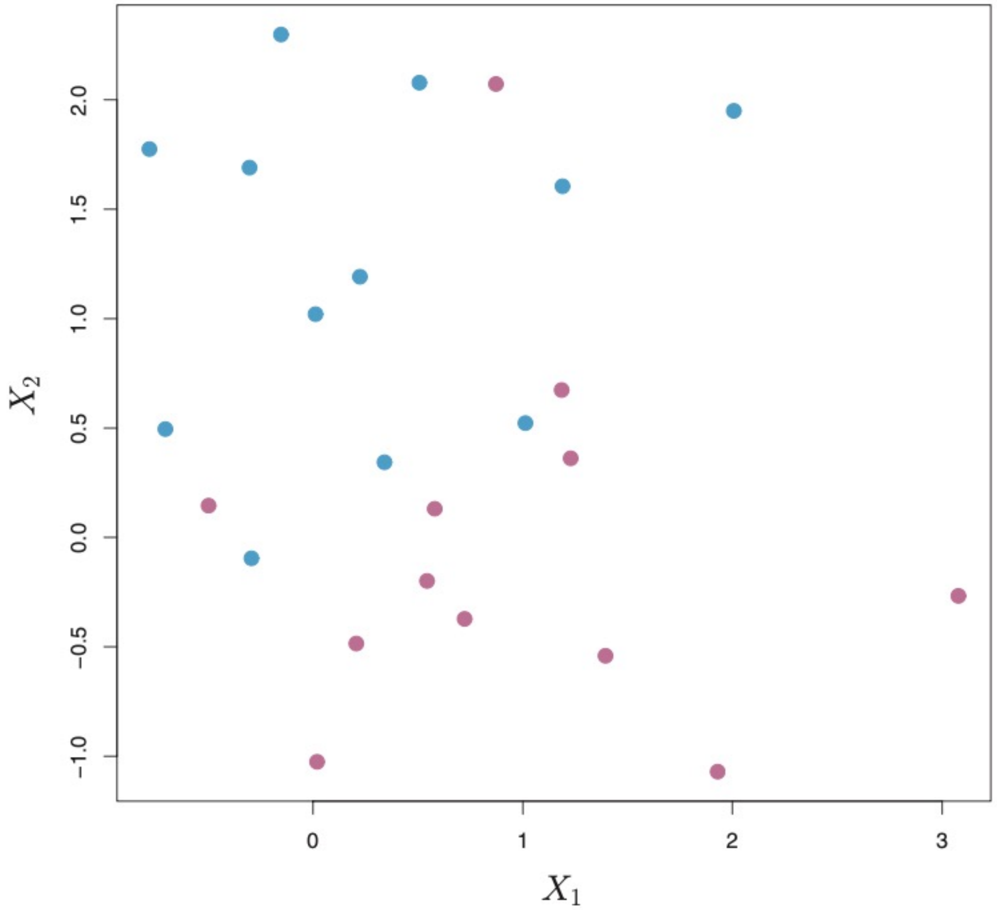
$$\text{subject to } \sum_{j=1}^{p} \beta_j^2 = 1,$$

$$y_i(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \dots + \beta_p x_{ip}) \geq M \;\; \forall\, i = 1, \dots, n.$$

- The inequality constraint: each observation is on the correct side of the hyperplane, provided $M > 0$.
- The equality constraint: just a normalization, since rescaling β does not change the hyperplane.
- Maximal M: each observation is on the correct side and at least a distance M from the hyperplane.

The Non-separable Case

- If no separating hyperplane exists, that means no maximal margin classifier, or no solution with $M > 0$.



The Non-separable Case: Soft Margin

- The non-separable case pursues:
  - Greater robustness to individual observations,
  - Better classification for most of the training observations.

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

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

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

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

- Slack variables ϵ_i allow observations to be on the wrong side.
- Cost C is a budget for the amount that the margin can be violated.

The Non-separable Case: Soft Margin

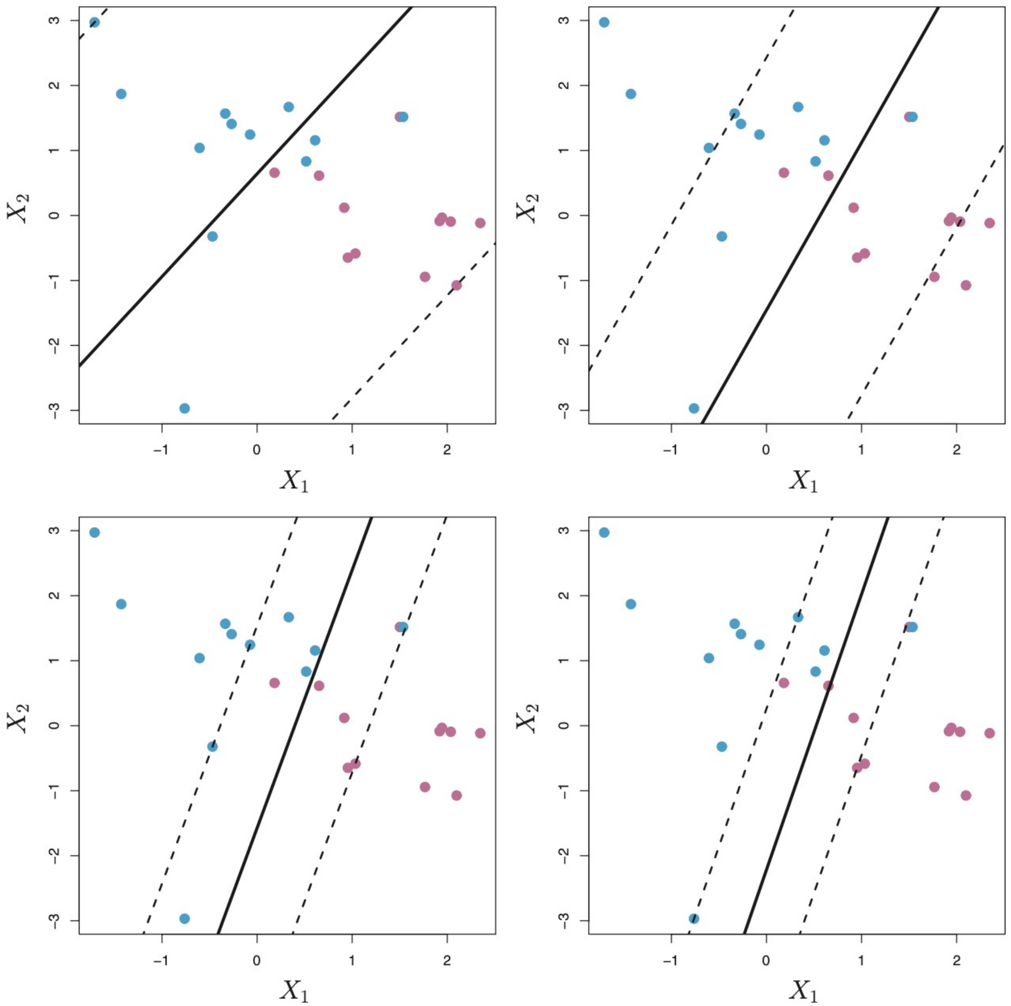
- Interpretation of C
  - $C = 0$, no budget for violations, $\epsilon_1 = \cdots \epsilon_n = 0$, the non-separable optimization works as the hard margin case.
  - $C > 0$, no more than C observations can be on the wrong side of the hyperplane.
  - C, as a nonnegative tuning parameter, can be chosen via cross-validation.

- Interpretation of ϵ_i
  - $\epsilon_i > 0$, the i^{th} observation is on the wrong side of the margin.
  - $\epsilon_i > 1$, it is on the wrong side of the hyperplane.

- “Support Vectors”: observations that lie on the margin, or on the wrong side of the margin.

The Non-separable Case: Soft Margin

- Larger C leads to wider margin with more violations, thus a classifier that is more biased but has lower variance.

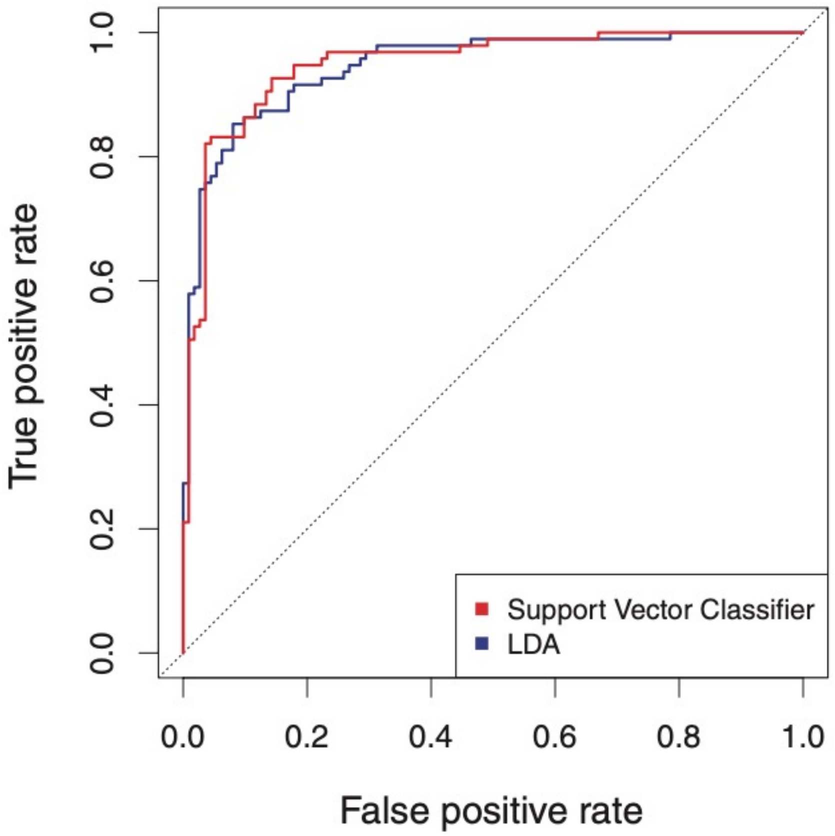


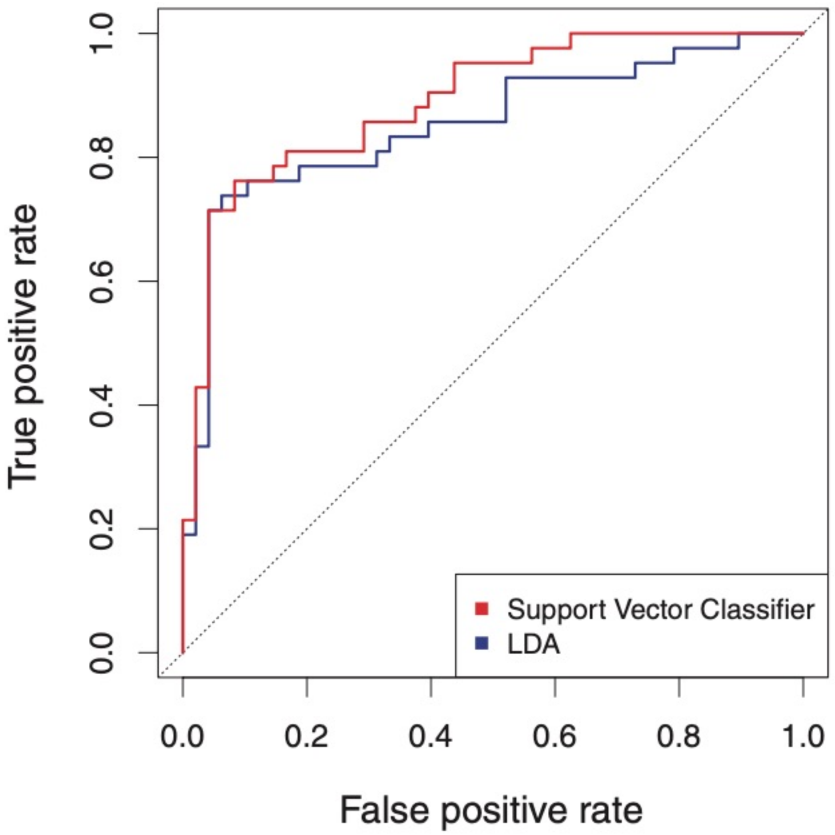
An Application to the Heart Disease Data

- Heart disease data set:
  - 303 patients who presented with chest pain
  - a binary outcome Yes/No indicates whether HD presents
  - 13 predictors including Age, Sex, Chol (a cholesterol measurement), and other heart and lung function measurements
  - randomly split into 207 training and 90 test observations

- Fitted value = $\hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \cdots + \hat{\beta}_p x_{ip}$.
- If fitted value is positive (negative), assign to the group +1 (-1).
- Can construct ROC curve from fitted values.

ROC Curves of LDA vs SVM



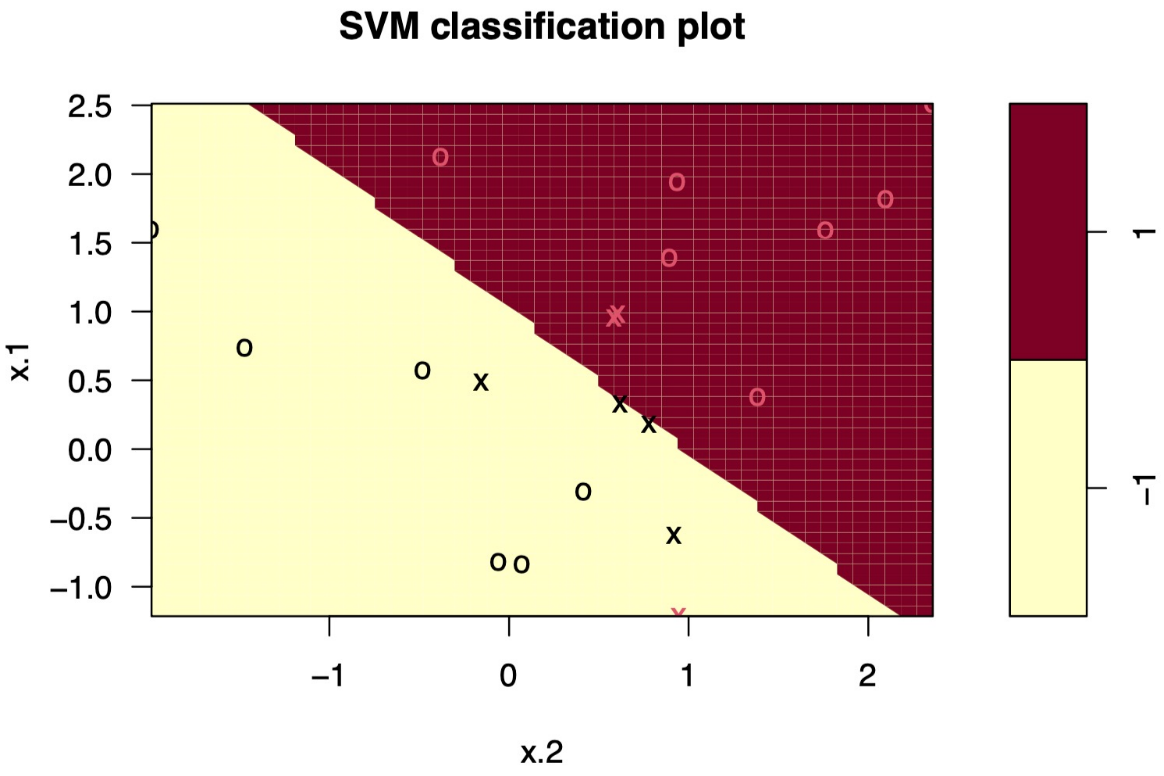


Left: Training

Right: Testing

Implementation

```r
plot(svmfit, dat)
```




```r
dat <- data.frame(x = x, y = as.factor(y))
# response must be a factor to perform classification

library(e1071)
svmfit <- svm(y ~ ., data = dat, kernel = "linear",
              cost = 10, scale = FALSE)
```

```r
tune.out <- tune(svm, y ~ ., data = dat, kernel = "linear",
                 ranges = list(cost = c(0.001, 0.01, 0.1, 1, 5, 10, 100)))
summary(tune.out)
```

```
##
## Parameter tuning of 'svm':
##
## - sampling method: 10-fold cross validation
##
## - best parameters:
##  cost
##   0.1
##
## - best performance: 0.05
##
## - Detailed performance results:
##    cost error dispersion
## 1 1e-03  0.55  0.4377975
## 2 1e-02  0.55  0.4377975
## 3 1e-01  0.05  0.1581139
## 4 1e+00  0.15  0.2415229
## 5 5e+00  0.15  0.2415229
## 6 1e+01  0.15  0.2415229
## 7 1e+02  0.15  0.2415229
```

SVMs with More than Two Classes

Two popular ideas:

- One-Versus-One Approach:
  - Construct $\binom{K}{2}$ SVMs for each pair of classes.
  - Assign a test observation to the class to which it was most frequently assigned in these $\binom{K}{2}$ pairwise classifications.
- One-Versus-All Approach:
  - Construct K SVMs, each time comparing one of the K classes to the remaining $K-1$ classes.
  - Assign a test observation x to the class for which $\beta_{0k} + \beta_{1k}x_1 + \cdots + \beta_{pk}x_p$ is largest.

Compare with LDA and Logistic Regression

- Support vector classifier is based on a small subset of the training samples (the support vectors). It is robust to samples far away from the hyperplane.
- LDA classifier depends on the mean and covariance matrix of all of the observations within each class. It is not robust to any observations.
- Logistic regression, similar to SVM, also has low sensitivity to observations far from the decision boundary. Why?

Compare with Logistic Regression

- An equivalent form of SVM:

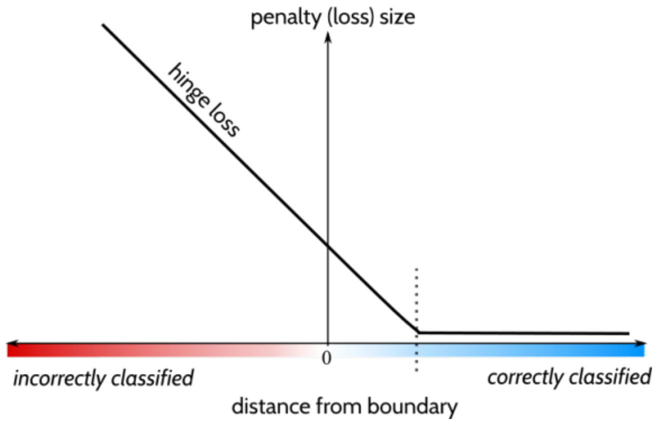
$$\underset{\beta_0,\beta_1,\ldots,\beta_p,\epsilon_1,\ldots,\epsilon_n}{\text{maximize}} \quad M$$

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

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

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

$$\Longleftrightarrow \quad \underset{\beta_0,\beta_1,\ldots,\beta_p}{\text{minimize}} \left\{ \sum_{i=1}^{n} \max\left[0, 1 - y_i f(x_i)\right] + \lambda \sum_{j=1}^{p} \beta_j^2 \right\},$$



$$f(X) = \beta_0 + \beta_1 X_1 + \ldots + \beta_p X_p$$

- Hinge loss + Ridge (ℓ_2) penalty

Compare with Logistic Regression

- Logistic regression minimizes (without penalty)

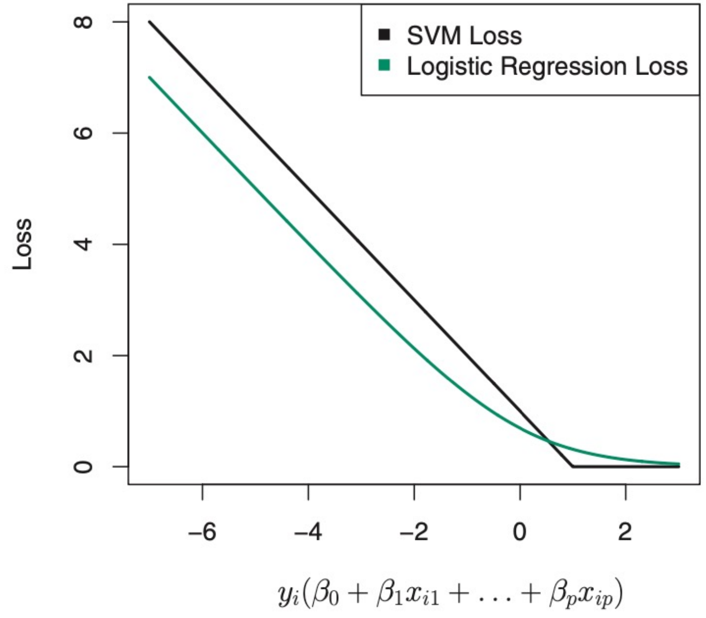
$$\textit{Neg-log-likelihood} = -\log\left(\Pi_i\, p(x_i)^{\frac{1+y_i}{2}}\left(1-p(x_i)\right)^{\frac{1-y_i}{2}}\right)$$

- Recall

$$\log\left(\frac{p(x_i)}{1-p(x_i)}\right) = f(x_i) \Longrightarrow p(x_i) = \frac{1}{e^{-f(x_i)}+1} \text{ and } 1-p(x_i) = \frac{1}{e^{f(x_i)}+1}$$

- Plug in $p(x_i)$

Neg-log-likelihood = $\sum_i \log(e^{-y_i f(x_i)} + 1)$



Support Vector Regression

- Classification:

$$\underset{\beta_0,\beta_1,\ldots,\beta_p}{\text{minimize}} \left\{ \sum_{i=1}^{n} \max\left[0, 1 - y_i f(x_i)\right] + \lambda \sum_{j=1}^{p} \beta_j^2 \right\},$$

- Regression:

$$\underset{\beta_0,\beta_1,\ldots,\beta_p}{\text{minimize}} \left\{ \sum_{i=1}^{n} \max[0, |y_i - f(x_i)| - \epsilon] + \lambda \sum_{j=1}^{p} \beta_j^2 \right\}$$

- ϵ-insensitive loss + ridge penalty

Summary

- SVM finds the classification hyperplane with maximal margin.
- Use CV to tune cost C, which controls bias-variance tradeoff.
- Equivalent to hinge loss + ridge penalty.
- Can be used for multi-class classification and regression.
- SVM with nonlinear kernels is beyond this scope of the course.