CSCE 633: Machine Learning

Lecture 10: Support Vector Machines

Texas A&M University

9-16-19

Last Time

- Logistic Regression
- Regularization

Goals of this lecture

- Support Vector Machines - an overview

Decision Boundaries

- It is important to consider what the decision boundary looks like
- Logistic Regression
- k-NN

SVM

- Maximal Margin Classifier
- Support Vector Classifier
- Support Vector Machines

What is a Hyperplane?

- In p dimensional space, a hyperplane is a flat subspace of $p-1$ dimensions
- What is it in 2D?
- What is it in 3D?

In 2D - $\beta_0 + \beta_1 x_1 + \beta_2 x_2 = 0$ defines a hyperplane
In p-Dim - $\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p = 0$
Can define if a point x lies on this hyperplane

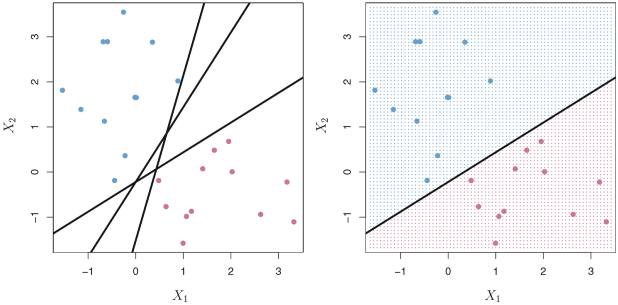
Hyperplane Boundaries

In 2D - $\beta_0 + \beta_1 x_1 + \beta_2 x_2 = 0$ defines a hyperplane
In p-Dim - $\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p = 0$
Can define if a point x lies on this hyperplane or on a side:

$\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p > 0$ means x lies above the hyperplane
$\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p < 0$ means x lies below the hyperplane

Now, if $y \in \{-1, +1\}$, then we want to train a classifier that finds this separating hyperplane

Hyperplanes



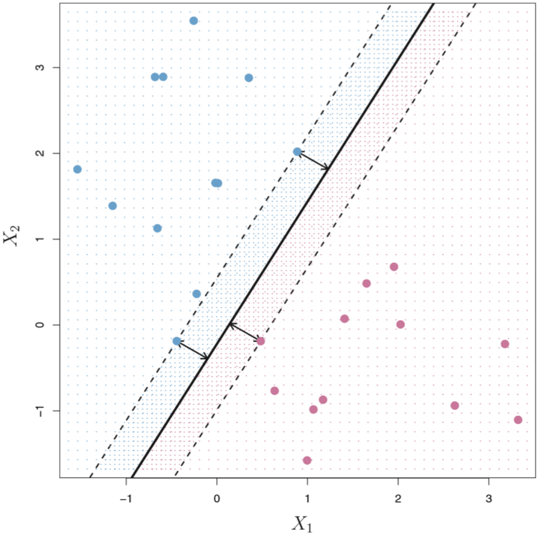
Which Hyperplane?

- If a separating hyperplane exists - then it is easy to classify
- $f(x) > 0$ implies $\hat{y} = 1$
- $f(x) < 0$ implies $\hat{y} = -1$
- Classifier $f = H = \{x \mapsto sgn(w \cdot x + b : w \in \mathbb{R}^N, b \in \mathbb{R}\}$
- Can use the magnitude of f to see how far away the object is from the hyperplane. The farther, the more confident we are in the prediction.
- However, as seen in last image, if one such hyperplane exists, infinite such hyperplanes exist, so which is the optimal hyperplane?

Maximal Marginal Hyperplane

- Pick the hyperplane that is the farthest from the training set points.
- Take the perpendicular distance of each point. Look to maximize this sum.
- Have to be careful - if p is large this can overfit
- want to find $f(x*) = sign(\beta_0 + \beta_1 x_1^* + \cdots + \beta_p x_p^*$
- Ideally we end up with a line that is the decision boundary - and an area between the closest points and the line

Hyperplanes



Maximal Marginal Hyperplane

- The maximal margin hyperplane depends directly on the points that lie on the margin
- These are called the support vectors
- So how do we build it?

$$x_1, \cdots, x_n \in \mathbb{R}^p$$

$$y_1, \cdots, y_n \in \{-1, +1\}$$

Then we want to:

$$maximize_{\beta_0, \beta_1, \cdots, \beta_p, M} M$$

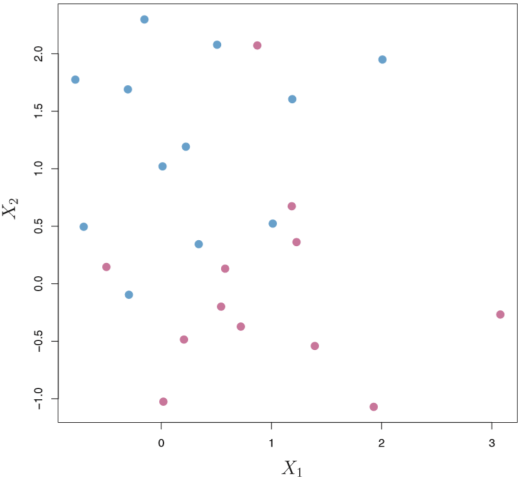
Subject to constraints:

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

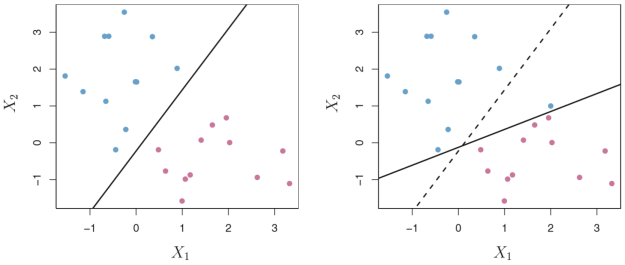
and

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

Maximal Marginal Hyperplanes



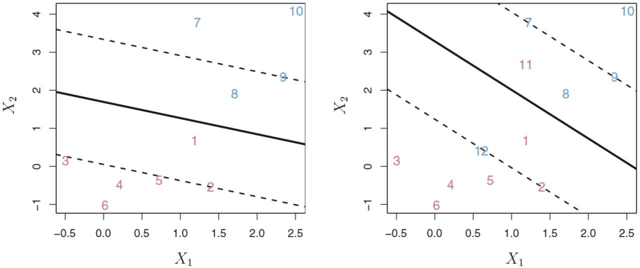
Maximal Marginal Hyperplanes



Maximal Marginal Hyperplane

- Learning details next time
- What if the training data is non-separable?
- Then no solution exists with $M > 0$
- What if we allow a soft margin (something that almost separates but has some mistakes?)

Soft Margin Hyperplanes



$$x_1, \cdots, x_n \in \mathbb{R}^p$$

$$y_1, \cdots, y_n \in \{-1, +1\}$$

Then we want to:

$$maximize_{\beta_0, \beta_1, \cdots, \beta_p, M} M$$

Subject to constraints:

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

and

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

$$\forall i = 1, \cdots, n$$

Support Vector Classifier: Slack Variables

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

where

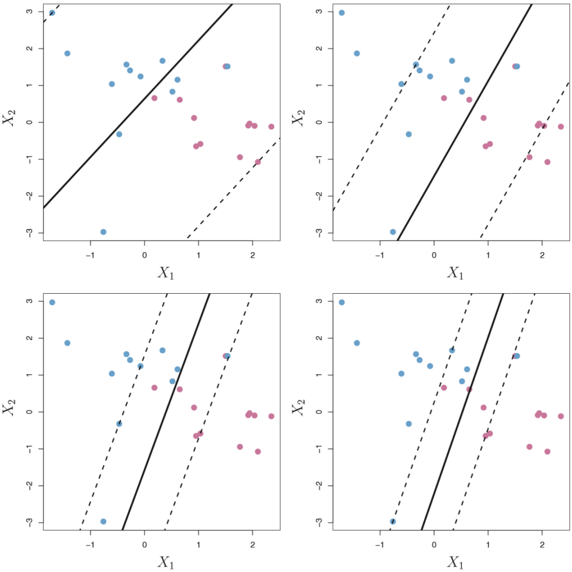
$$\epsilon_i \geq 0$$
$$\sum_{i=1}^{n} \epsilon_i \leq C$$

- C is a non-negative tuning parameter
- M is the width of the margin
- ϵ_i are the slack variables. When $\epsilon_i > 1$ the object is on the wrong side of the hyperplane, when $\epsilon_i > 0$ the object violates the margin
- Therefore, C determines the number and severity of margin violations

Support Vector Classifier: C

- C is often chosen through cross-validation
- Small C leads to low bias but high variance
- Large C leads to high bias but low variance
- Only items on the margin or those that violate the margin really matter for setting the hyperplane
- These, again, are called the support vectors, and C affects how many we have

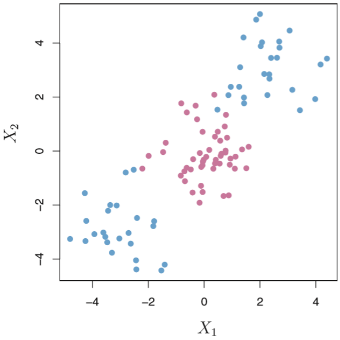
SVC

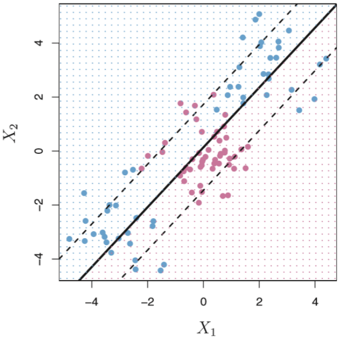


Support Vector Classifier: C

- Robust to behavior far from the hyperplane
- There is similarity to the decision boundary found by SVC and Logistic Regression
- Now - what if we want a non-linear boundary?

Multi-class?





- SVC is natural for 2 class decision
- Remember back to Logistic Regression with interaction terms
- $x_1, x_2, \cdots, x_p, x_1^2, x_2^2, \cdots, x_p^2$ now we have p terms
- We can re-write SVC to maximize M subject to

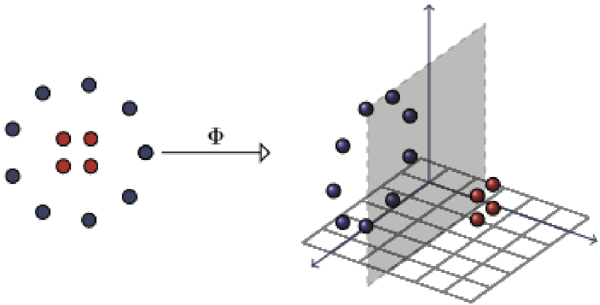
$$y_i(\beta_0 + \sum_{j=1}^{p} \beta_{j1} x_{ij} + \sum_{j=1}^{p} \beta_{j2} x_{ij}^2) \geq M(1 - \epsilon_i)$$

and

$$\sum_{j=1}^{p} \sum_{k=1}^{2} \beta_{jk}^2 = 1$$

Can we enlarge the feature space even more? Would this give us non-linear decision boundaries?

Takeaways and Next Time



- Support Vector Machines
- Next Time: Support Vector Machines