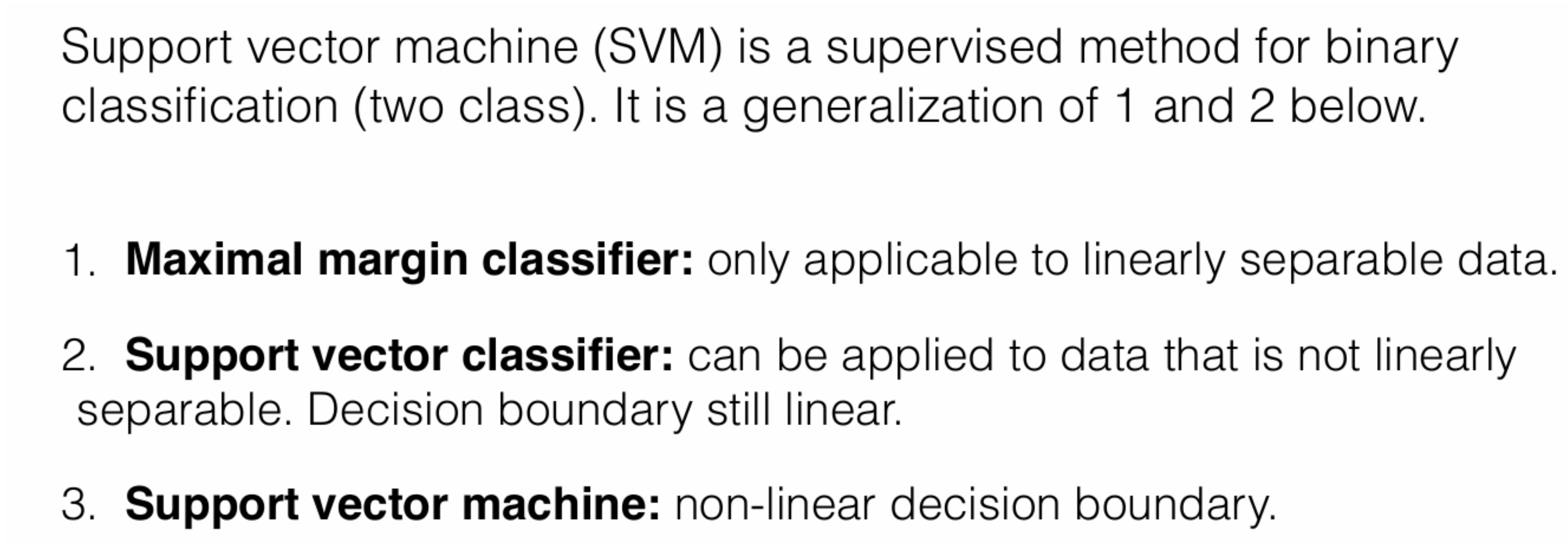
Support Vector Machine

Support Vector Machine



Support vector machine (SVM) is a supervised method for binary classification (two class). It is a generalization of 1 and 2 below.

1. **Maximal margin classifier:** only applicable to linearly separable data.

2. **Support vector classifier:** can be applied to data that is not linearly separable. Decision boundary still linear.

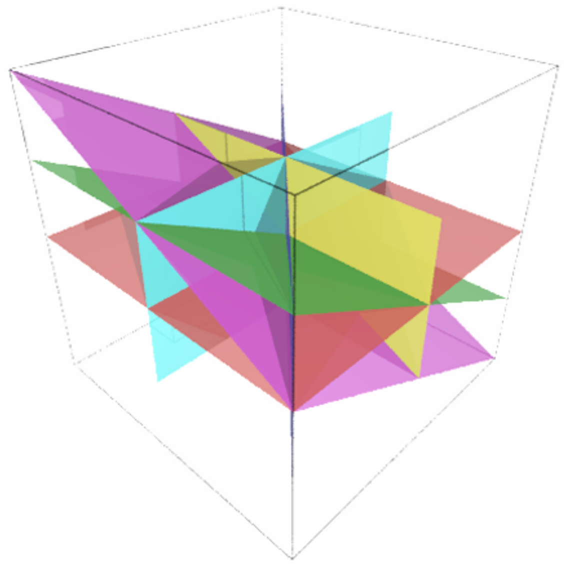
3. **Support vector machine:** non-linear decision boundary.

What is a hyperplane?

In p-dimensional space, a hyperplane is a $(p-1)$-dimensional affine subspace.

In 2D, a hyperplane is a flat 1D subspace, aka a line.

In 3D, a hyperplane is a flat 2D subspace, aka a plane.



Mathematical Definition

A 2D hyperplane is defined by the equation

$$\beta_0 + \beta_1 X_1 + \beta_2 X_2 = 0$$

By "define", we mean that any $X = (X_1, X_2)$ for which the above equation holds is a point on the hyperplane.

Mathematical Definition

In p dimensions, a hyperplane is defined by the equation

$$\beta_0 + \beta_1 X_1 + \ldots + \beta_p X_p = 0$$

Similarly, any $X = (X_1, X_2, \ldots, X_p)$ for which the above equation holds is a point on the hyperplane.

Separating Hyperplane

Instead of a point on the hyperplane, consider $\boldsymbol{X}$ for which

$$\beta_0 + \beta_1 X_1 + \ldots + \beta_p X_p > 0$$

This point lies on one side of the hyperplane. An $\boldsymbol{X}$ for which

$$\beta_0 + \beta_1 X_1 + \ldots + \beta_p X_p < 0$$

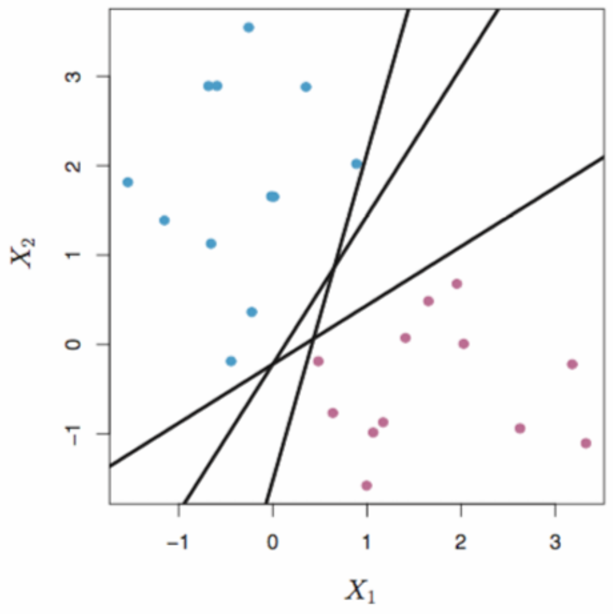
lies on the other side of the hyperplane.

We can think of the hyperplane as dividing the p-dimensional space into two halves.

Separating Hyperplane Classifier

Idea: Use a separating hyperplane for binary classification.

Key assumption: Classes can be separated by a linear decision boundary.



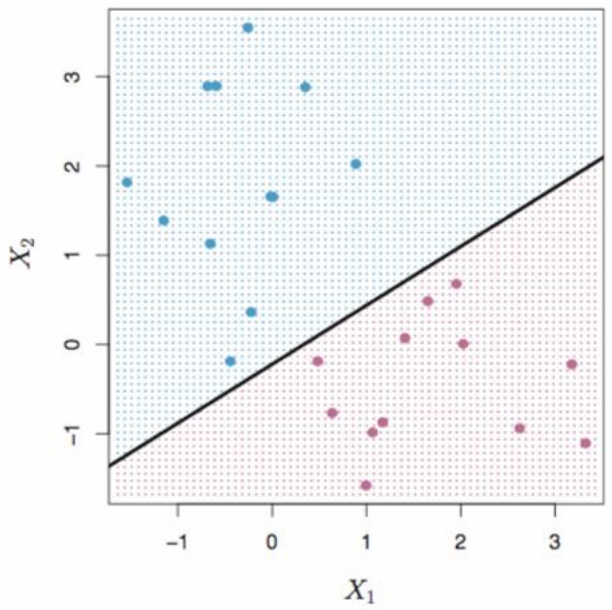
Separating Hyperplane Classifier

To classify new data points:

Assign class by location of new data point with respect to the hyperplane.

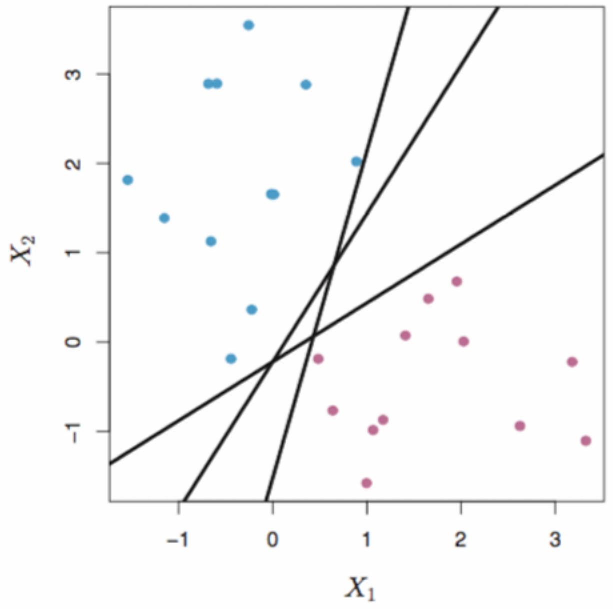
$$\hat{y} = \mathrm{sign}(\beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p)$$

The farther away a point is from the separating hyperplane, the more confident we are about its class assignment.

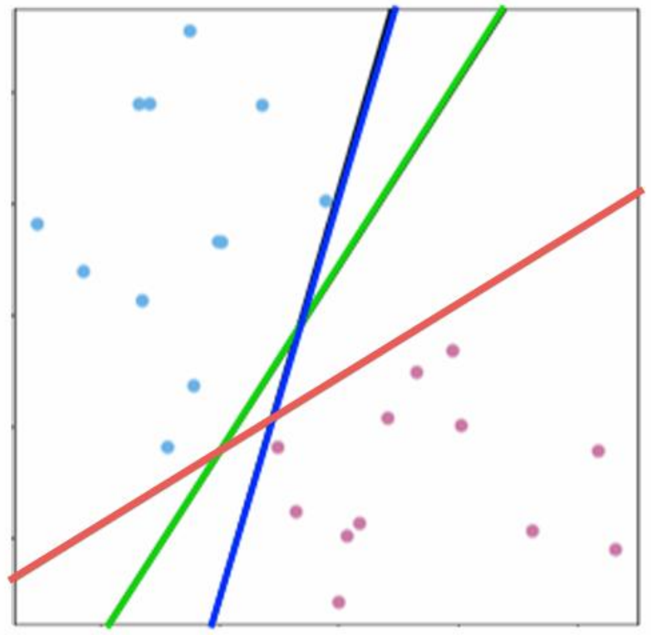


Separating Hyperplane Classifier

Notice that for a linearly separable dataset, there are many possible separating hyperplanes that divide the dataset into two classes (in fact, an infinite number).



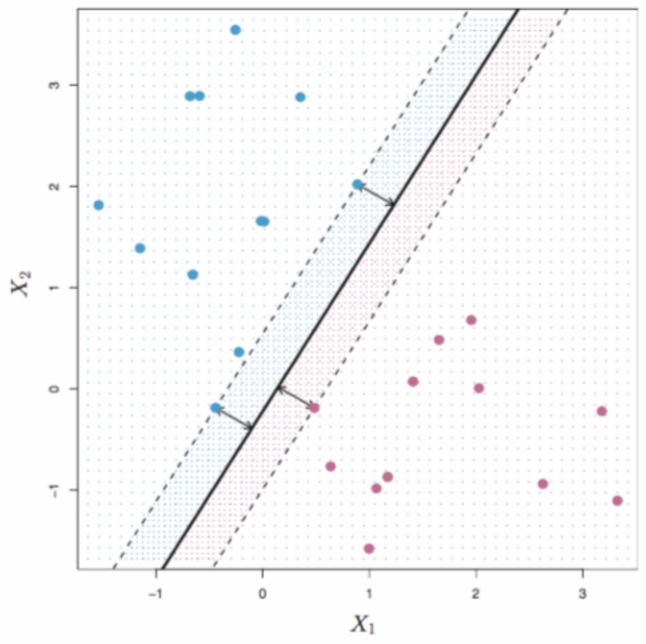
Which decision boundary?



Maximal Margin Hyperplane

Which of the infinite separating hyperplanes should we choose?

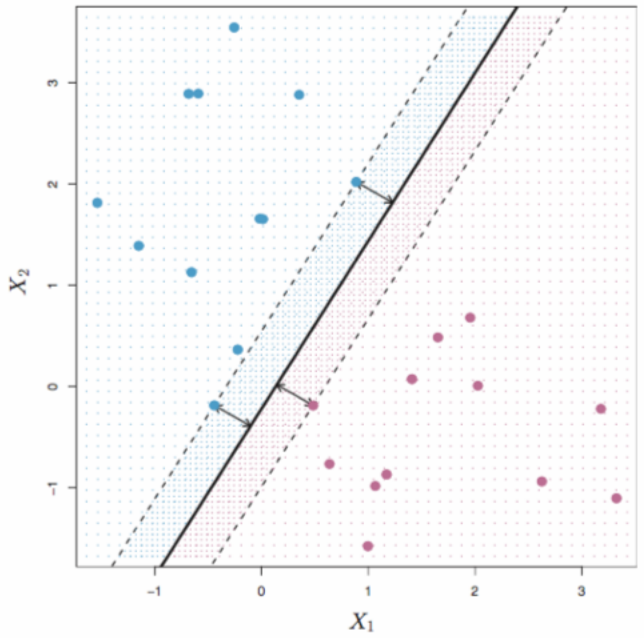
A natural choice is the **maximal margin hyperplane**, the separating hyperplane that is farthest from the training samples.



Maximal Margin Hyperplane

Margin: smallest distance between any training observation and the hyperplane.

Support vectors: training observations whose distance to the hyperplane is equal to the margin



Why is it called a support vector?

“Support”: maximal margin hyperplane only depends on these observations.

“Vector”: points are vectors in p-dimensional space.

If support vectors are perturbed, then MM hyperplane will change.

If other training observations perturbed (provided not perturbed within margin distance of hyperplane), then MM hyperplane not affected.

Finding Maximal Margin Classifier

To find the maximal margin hyperplane on data $(\vec{x}^{(i)}, y^{(i)}), \quad y^{(i)} \in \{-1, 1\}$ solve:

$$\max_{\beta_0, \ldots, \beta_p} M$$

maximize the margin, M

$$\text{subject to} \sum_{j=0}^{p} \beta_j^2 = 1$$

constraint necessary for well-defined optimization problem

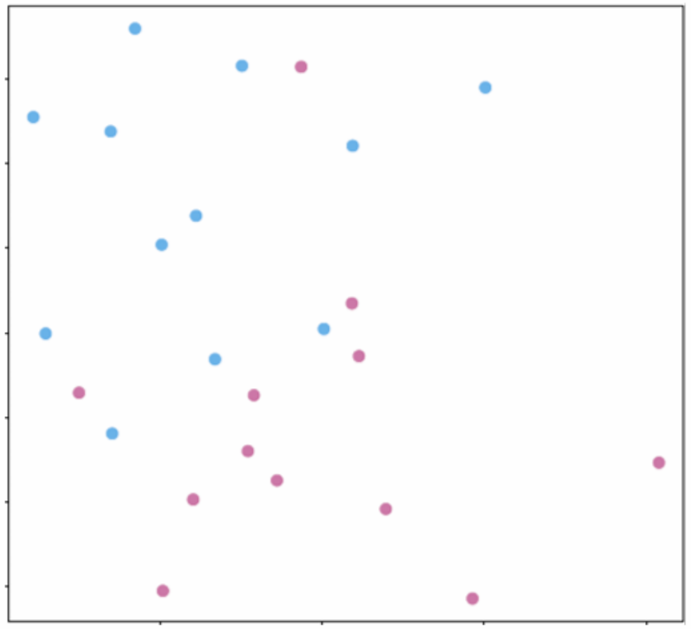
$$\boxed{y^{(i)}(\beta_0 + \beta_1 x_1^{(i)} + \ldots \beta_p x_p^{(i)}) \geq M, \quad \forall i}$$

all training points must be at least distance M from hyperplane

Maximal Margin Classifier

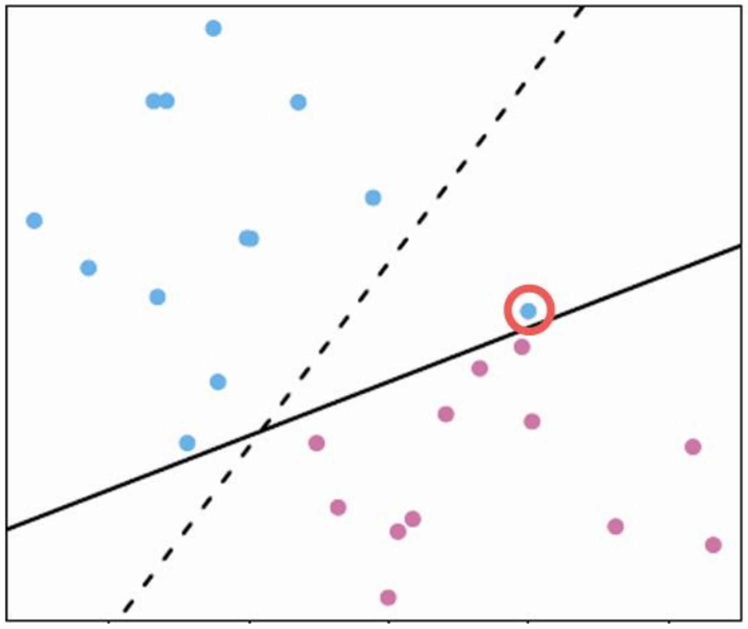
Recall the assumption: Classes can be separated by a linear decision boundary.

What if there is no separating hyperplane?



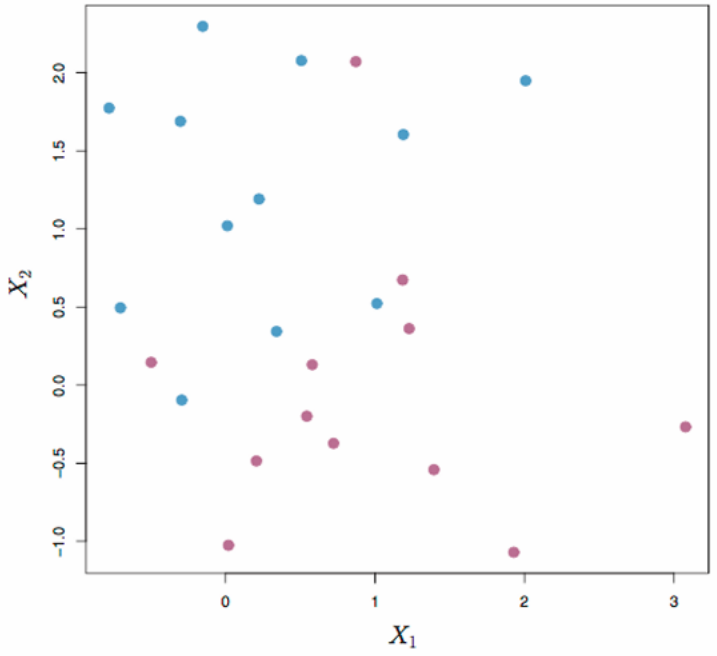
Maximal Margin Classifier

- Can be sensitive to individual observations
- May overfit training data



Support Vector Classifier

Like the maximal margin classifier, it looks for a hyperplane to perform classification.

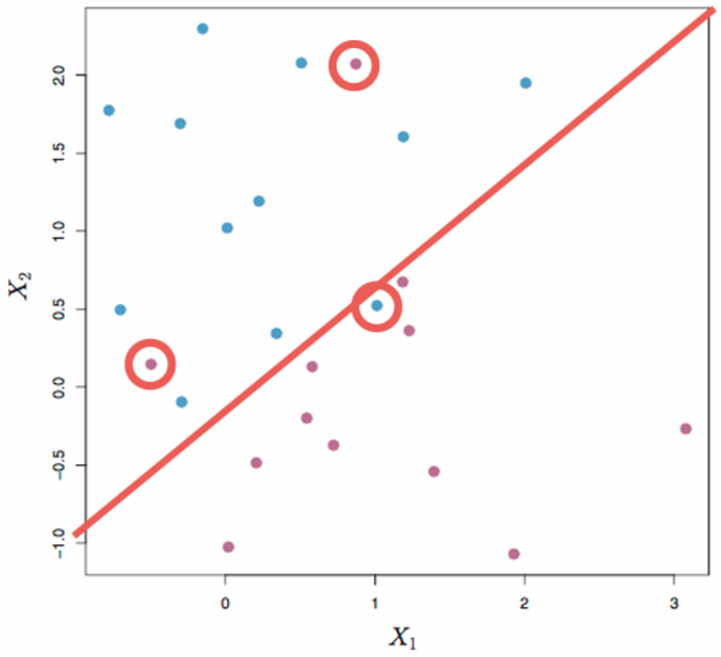


Support Vector Classifier

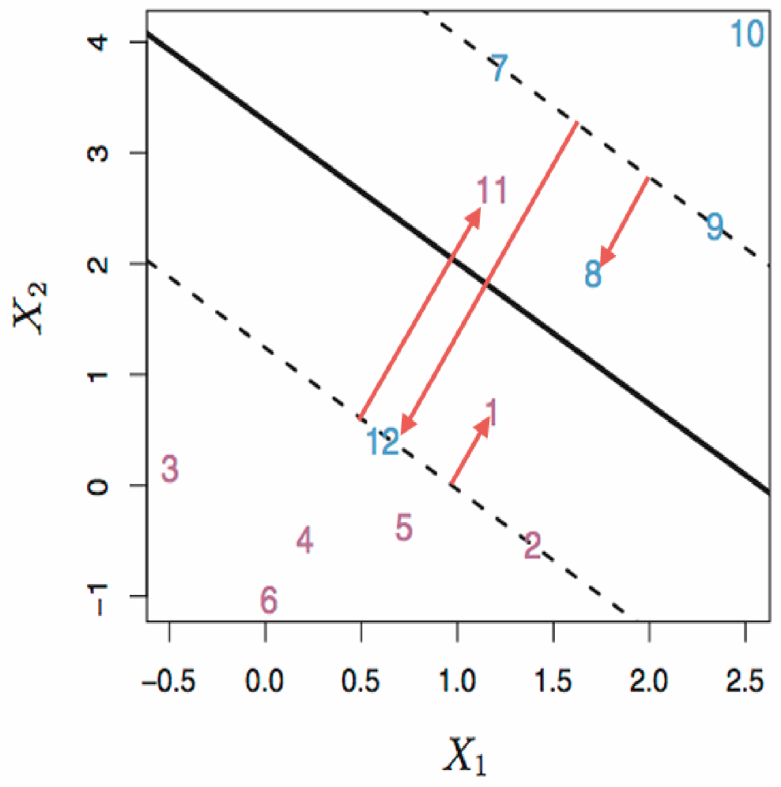
Like the maximal margin classifier, it looks for a hyperplane to perform classification.

However, training samples are allowed to be on the "wrong side" of the margin or hyperplane.

This hyperplane *almost* separates the classes using a "soft margin".

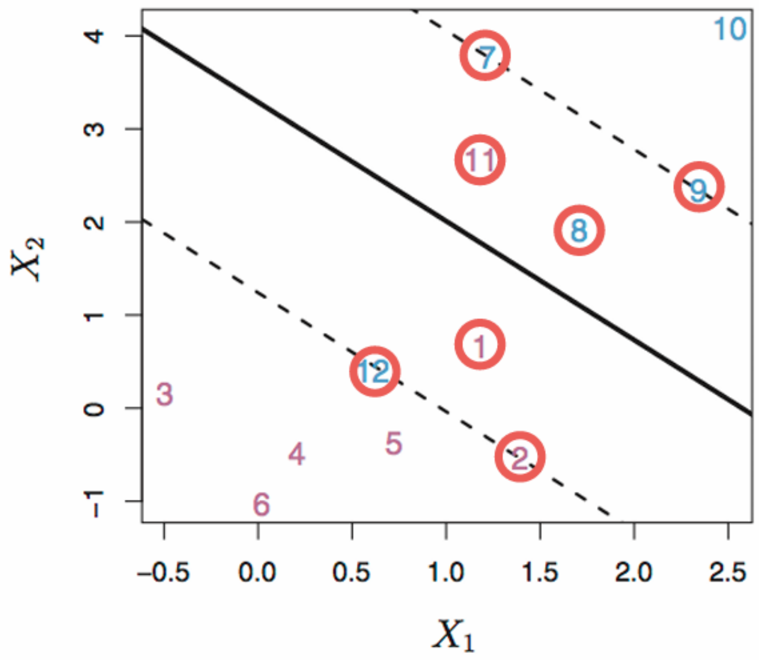


Support Vector Classifier



Some points are allowed to violate the margin.

Support Vector Classifier



Support vector classifiers also have support vectors.

They are points lying directly on the margin, or on the wrong side of the margin for their class.

These observations affect the hyperplane.

Finding Support Vector Classifier

To find the support vector classifier hyperplane, solve:

$$\max_{\beta_0,\ldots,\beta_p,\epsilon_1,\ldots,\epsilon_n} M$$

$$\text{subject to} \sum_{j=0}^{p} \beta_j^2 = 1$$

training points must be at least distance M from hyperplane, or pay a penalty ε_i

$$y^{(i)}(\beta_0 + \beta_1 x_1^{(i)} + \ldots \beta_p x_p^{(i)}) \geq M(1 - \epsilon_i), \quad \forall i$$

limit on total penalties. C is a constant.

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

“slack” variable ε_i

Finding Support Vector Classifier

To find the support vector classifier hyperplane, solve:

$$\max_{\beta_0,\ldots,\beta_p,\epsilon_1,\ldots,\epsilon_n} M$$

$$\text{subject to} \sum_{j=0}^{p} \beta_j^2 = 1$$

$$y^{(i)}(\beta_0 + \beta_1 x_1^{(i)} + \ldots \beta_p x_p^{(i)}) \geq M(1 - \epsilon_i), \quad \forall i$$

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

In other words, you can violate the margin, but only by a total amount C on your entire dataset.

Support Vector Classifier

- Slack variables ε_i allow for violations of the margin.
  - $\varepsilon_i = 0$: training point is on correct side of margin
  - $\varepsilon_i > 0$: training point violates the margin
  - $\varepsilon_i > 1$: training point is misclassified (wrong side of hyperplane)
- Penalty parameter C is the total “budget” for violations.
  - Allows at most C misclassifications on training set.

How do we choose C?

As with many things we don't know *a priori* in machine learning, C is a hyperparameter that we tune using cross-validation.

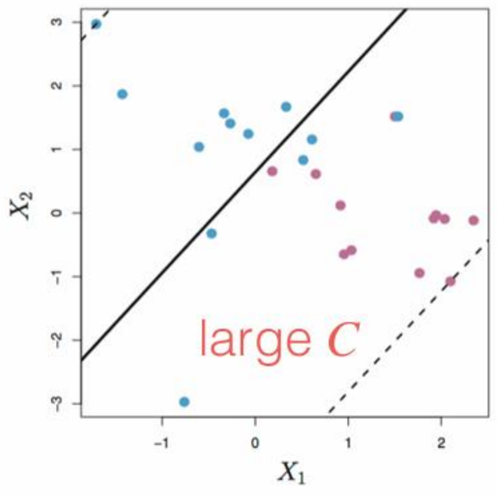
Note that it must be non-negative.

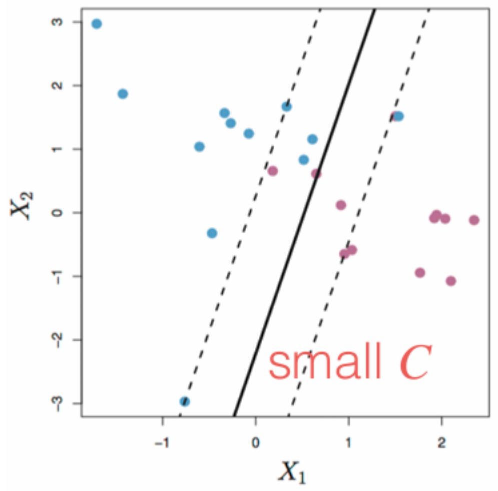
If $C = 0$, we recover the maximal margin classifier (if one exists).

As C goes from small to large, there is a bias-variance tradeoff.

Bias, Variance and C

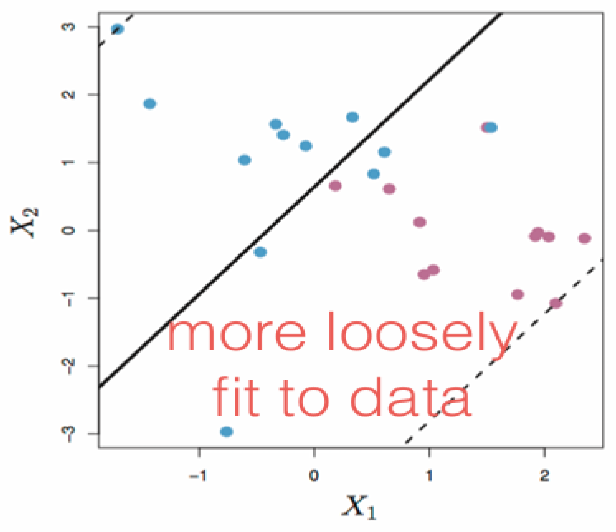
- Large C
  - Large violation budget
  - Large margin
  - Many support vectors
- Small C
  - Small violation budget
  - Small margin
  - Few support vectors

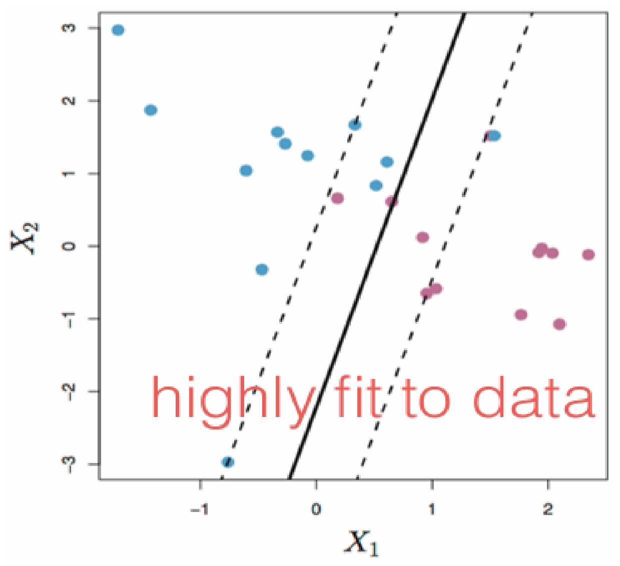




Bias, Variance and C

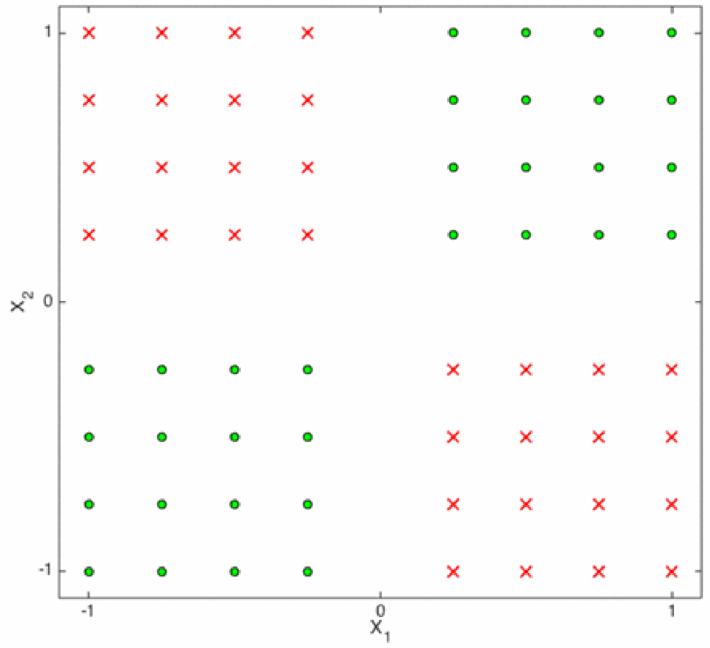
- Large C
  - High bias
  - Low variance
- Small C
  - Low bias
  - High variance





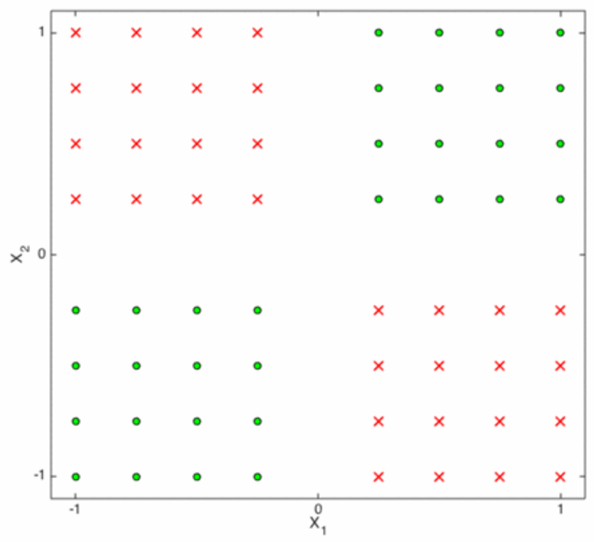
Expanding Feature Space

Some datasets are not linearly separable, but they *become* linearly separable when transformed into a *higher* dimensional space.



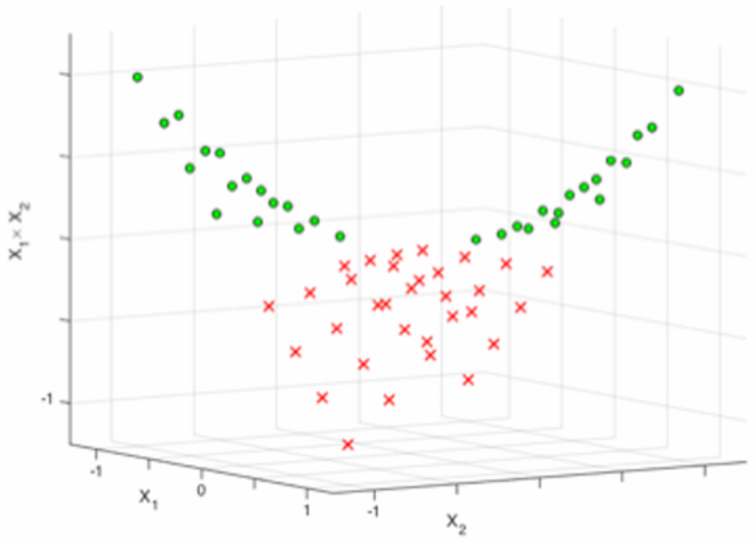
Expanding Feature Space

Original feature space



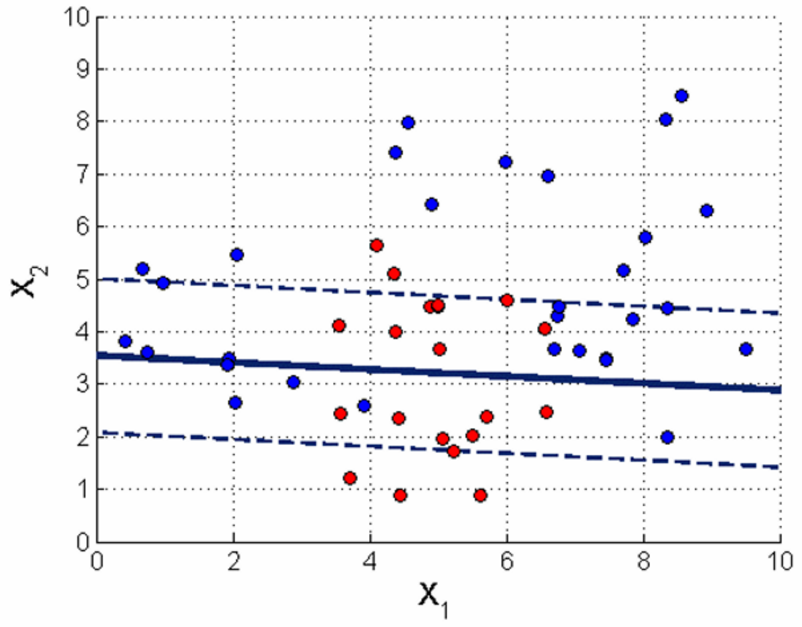
variables x_1, x_2

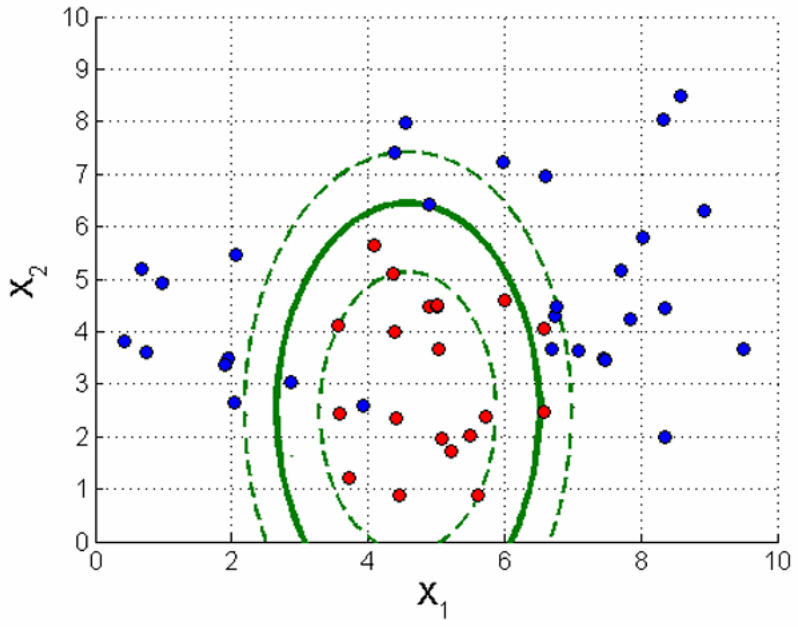
New feature space



variables x_1, x_2, x_1x_2

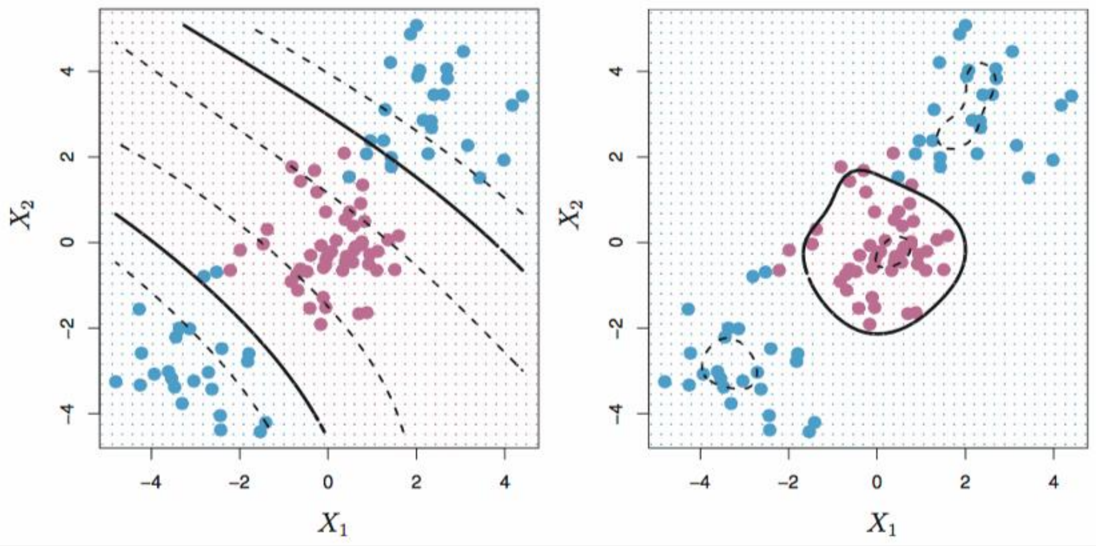
Non-linear Decision Boundary





Support Vector Machine (SVM)

Extends the support vector classifier by using **kernel functions** to achieve non-linear decision boundaries.



SVM with 3+ Classes

SVMs are designed for binary classification, given the nature of a separating hyperplane.

We can adapt SVMs to perform classification when we have more than 2 classes.

Popular approaches:

- One-versus-one
- One-versus-all

One-versus-one Classification

Construct an SVM for each pair of classes.

For k classes, this requires training $k(k\text{-}1)/2$ SVMs.

To classify a new observation, apply all $k(k\text{-}1)/2$ SVMs to the observation. Take the most frequent class among pairwise results as the predicted class.

Con: computationally expensive for large k.

One-versus-all Classification

Construct an SVM for each class against the k-1 other classes pooled together.

For k classes, this requires training k SVMs.

Distance to separating hyperplane is a proxy for confidence of classification. For new observation, choose the “highest confidence” class as the prediction.

Note: These slides are adapted from the lecture notes of Dr. Sherrie Wang