Support Vector Machine

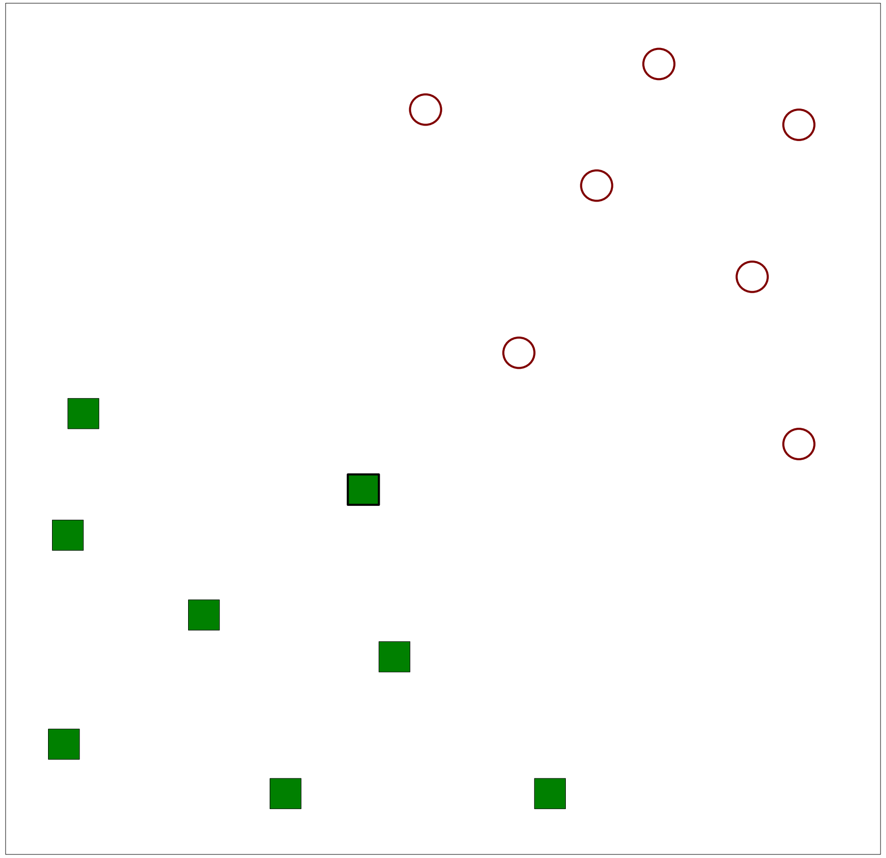
UNIVERSITÀ DI PISA

SVM

- This technique has its roots in statistical learning
- Promising results in different applications
  - Text classification, handwritten digit recognition
- Works very well with high-dimensional data
- Represents the **decision bourndary** by a subset of training examples
  - **Support vectors**

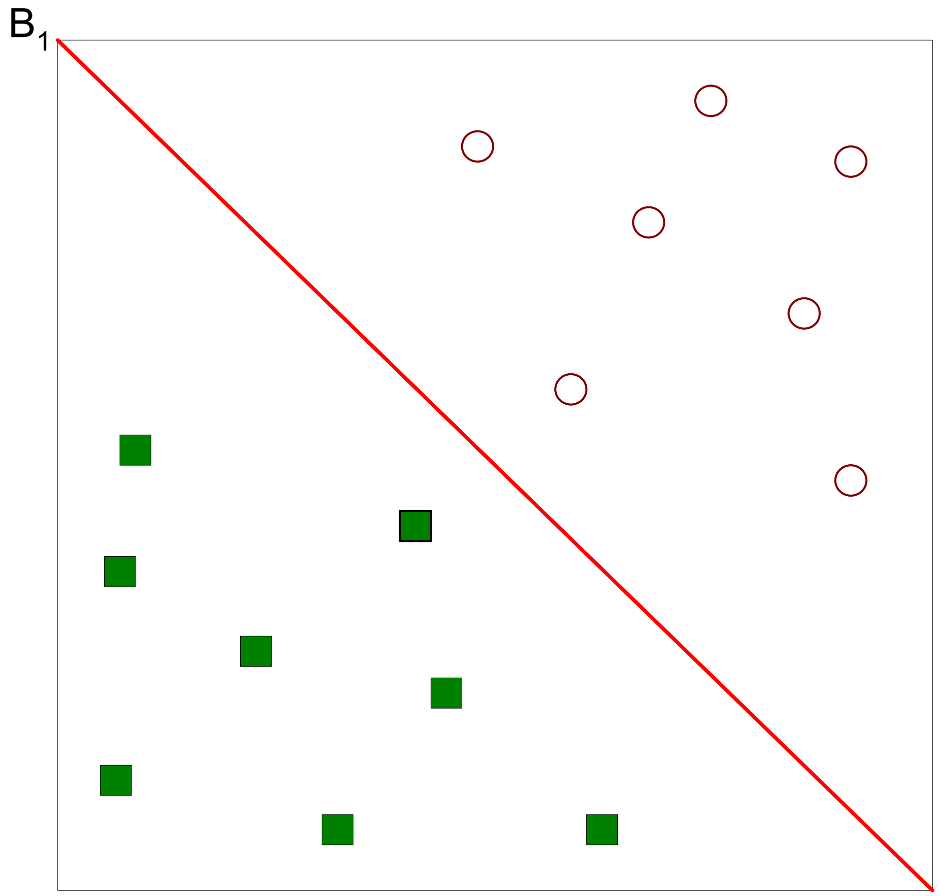
Linear Separators

- Binary classification can be viewed as the task of separating classes in feature space
- Find a linear hyperplane (decision boundary) that separates the data.



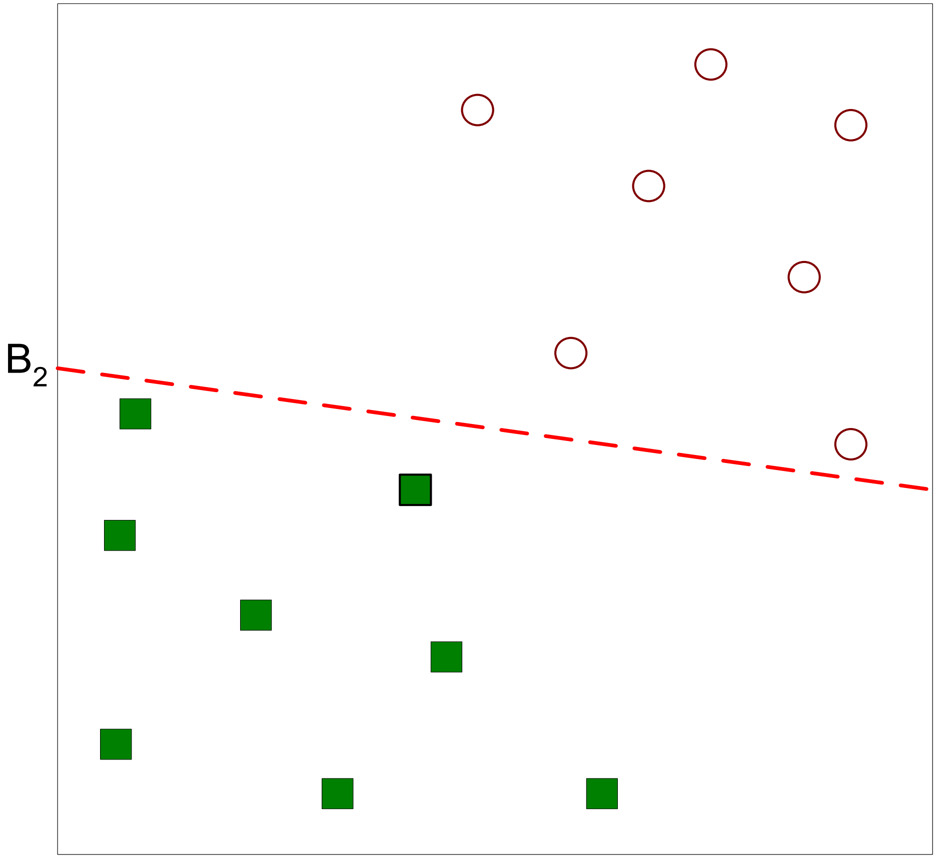
Maximum Margin Hyperplanes

- One possible solution.



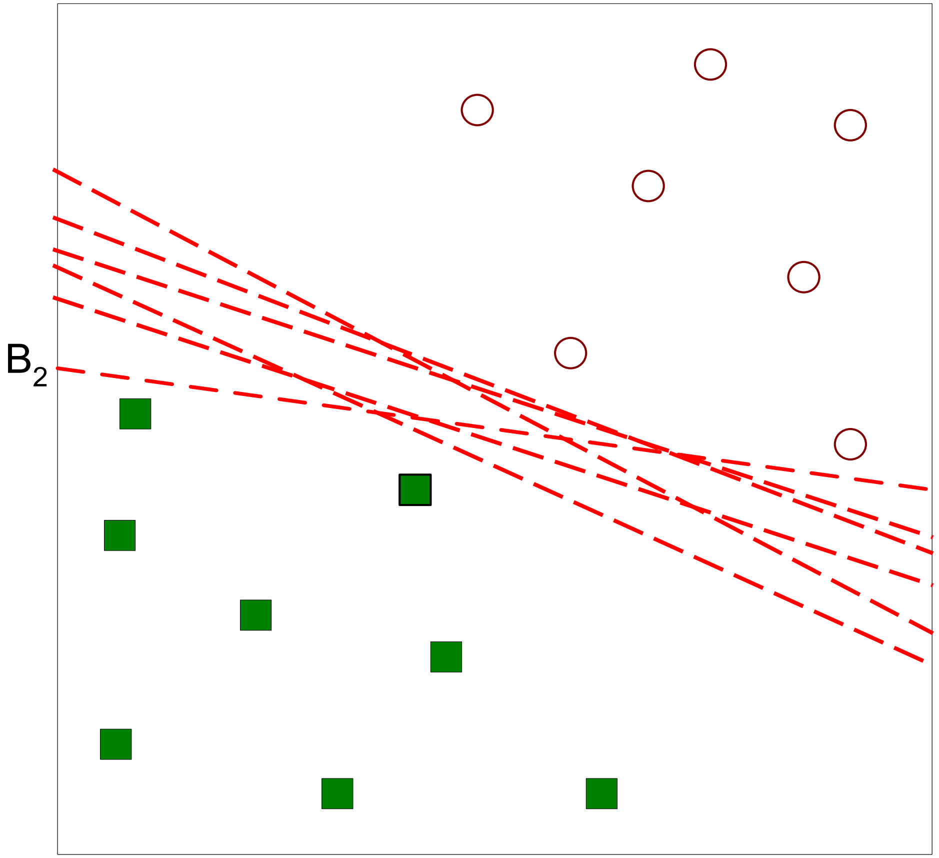
Linear Separators

- Another possible solution.



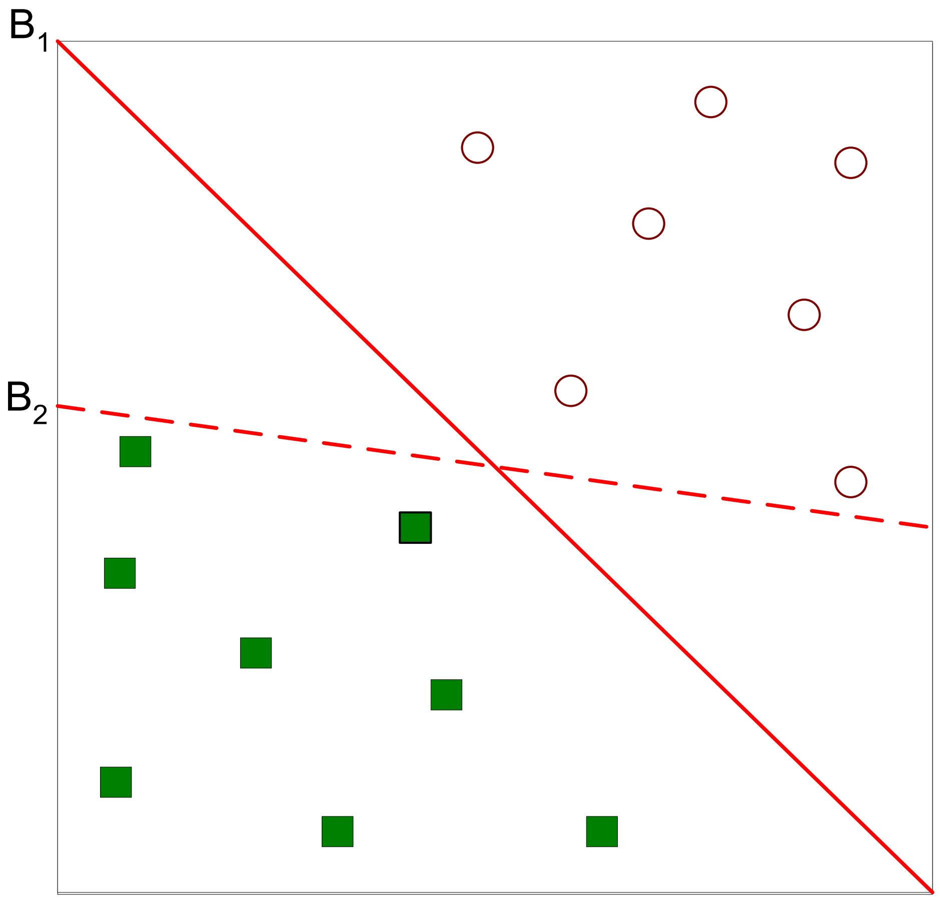
Linear Separators

- Other possible solutions.



Linear Separators

- Let's focus on B_1 and B_2.
- Which one is better?
- How do you define better?

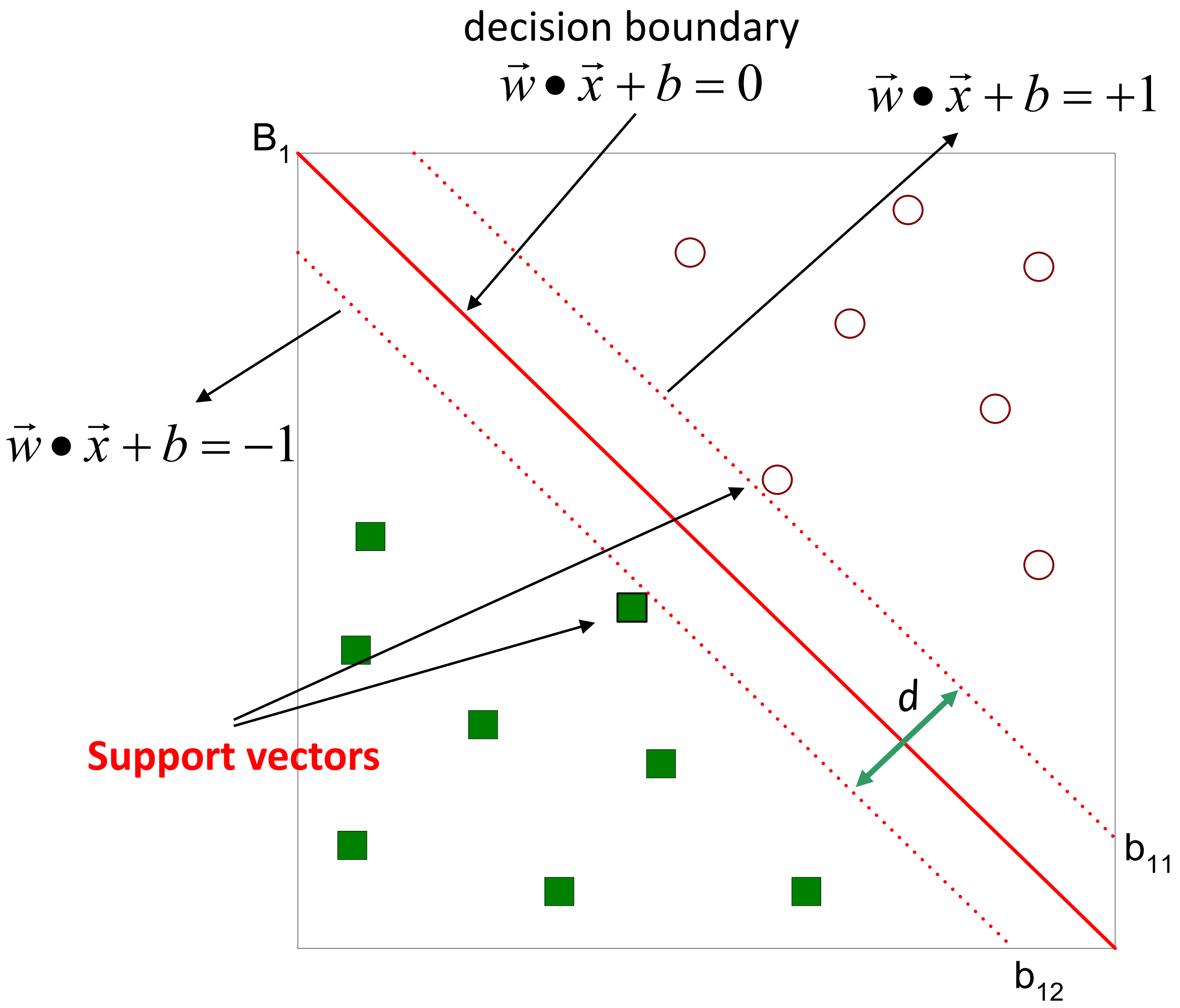


Support Vector Machine (SVM)

- SVM represents the decision boundary using a subset of the training examples, known as the **support vectors**.
- SVM is based on the concept of **maximal margin hyperplane**

Classification Margin

- Decision Boundary is associated to 2 hyperplanes obtained by support vectors
- Examples closest to the hyperplane are ***support vectors***.
- ***Margin*** *d* of the separator is the distance between support vectors.
- A simple binary classification problem:
  - buys computer = yes → +1
  - buys computer = no → -1



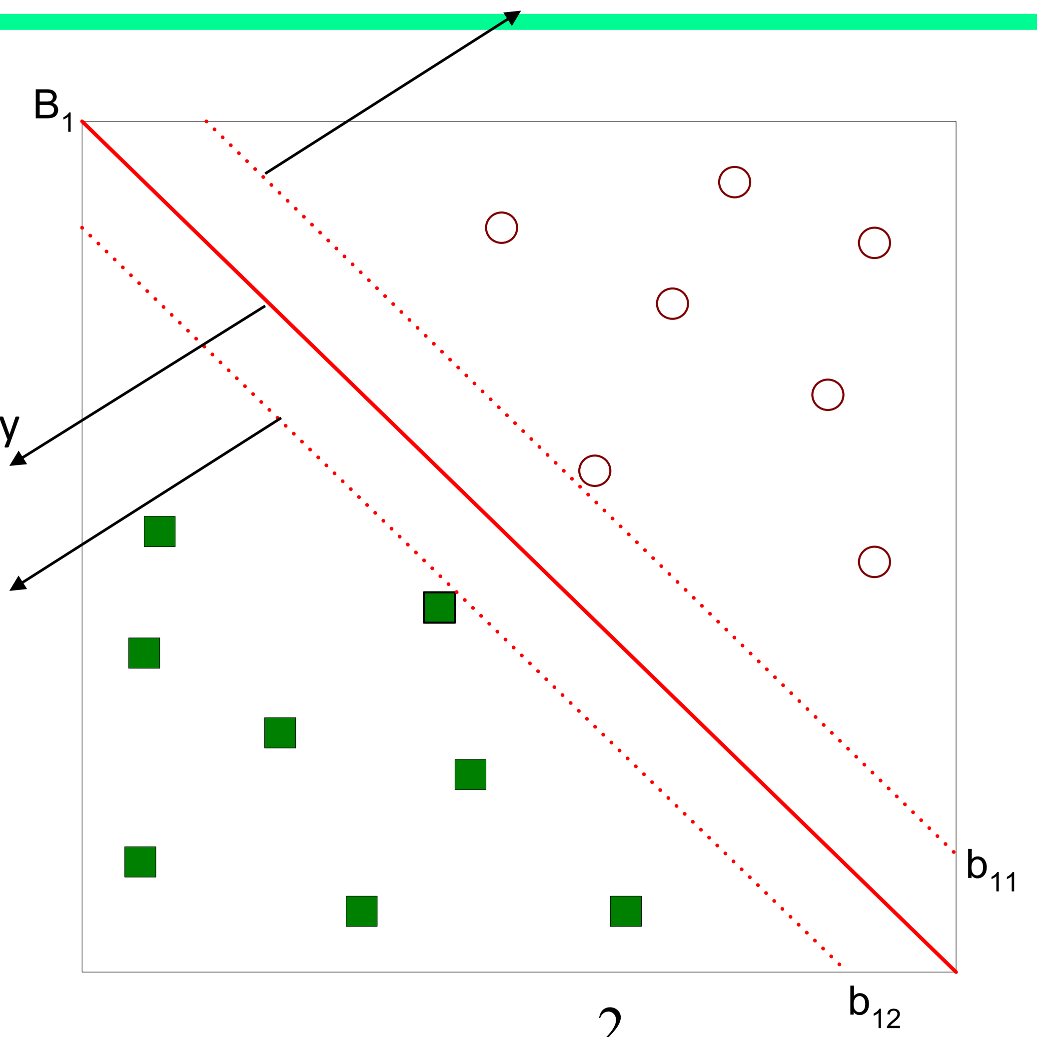
Linear SVM: Separable Case

- A linear SVM is a classifier that searches for a hyperplane with the largest margin (a.k.a. maximal margin classifier).

- *w* and *b* are parameters.

- Given *w* and *b* the classifier works as

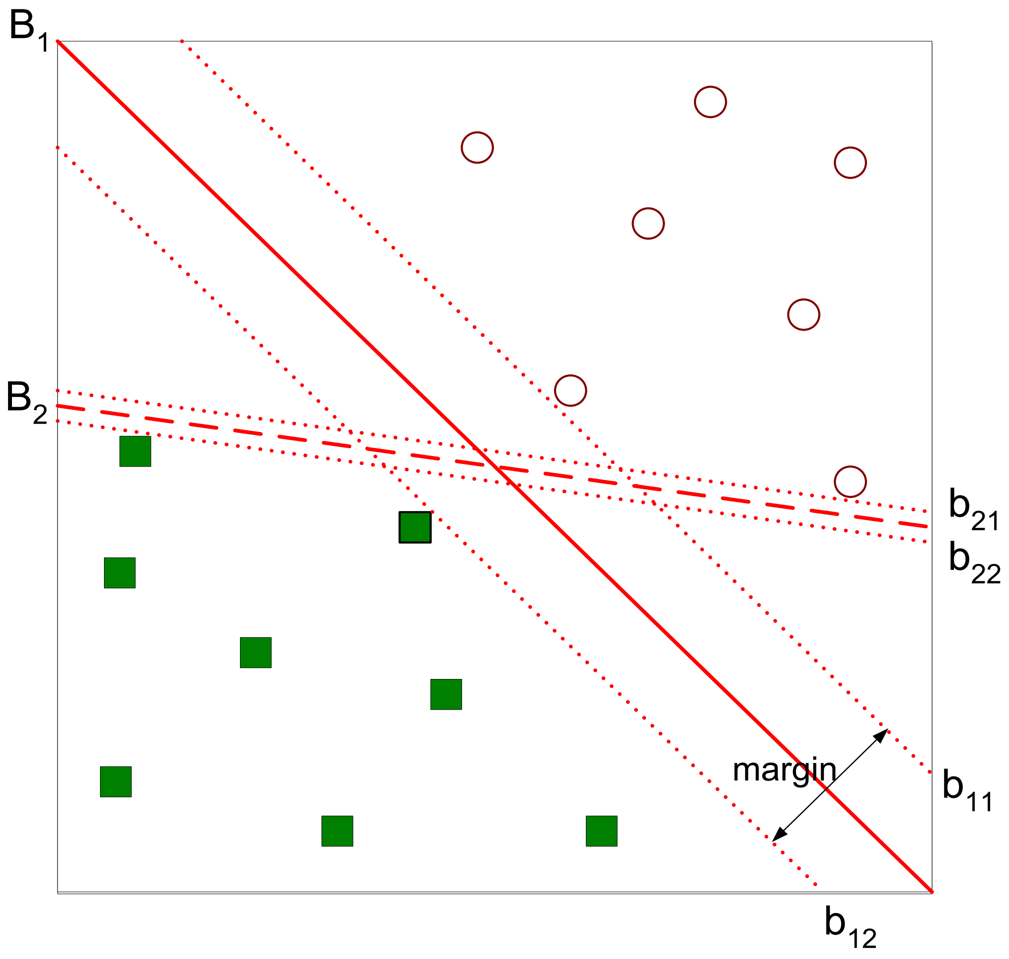
$$f(\vec{x}) = \begin{cases} 1 & \text{if } \vec{w} \bullet \vec{x} + b \geq 1 \\ -1 & \text{if } \vec{w} \bullet \vec{x} + b \leq -1 \end{cases}$$



$$\text{Margin} = \frac{2}{\|\vec{w}\|}$$

Maximum Margin Hyperplanes

- The best solution is the hyperplane that **maximizes** the **margin**.
- Thus, B_1 is better than B_2.



Learning a Linear SVM

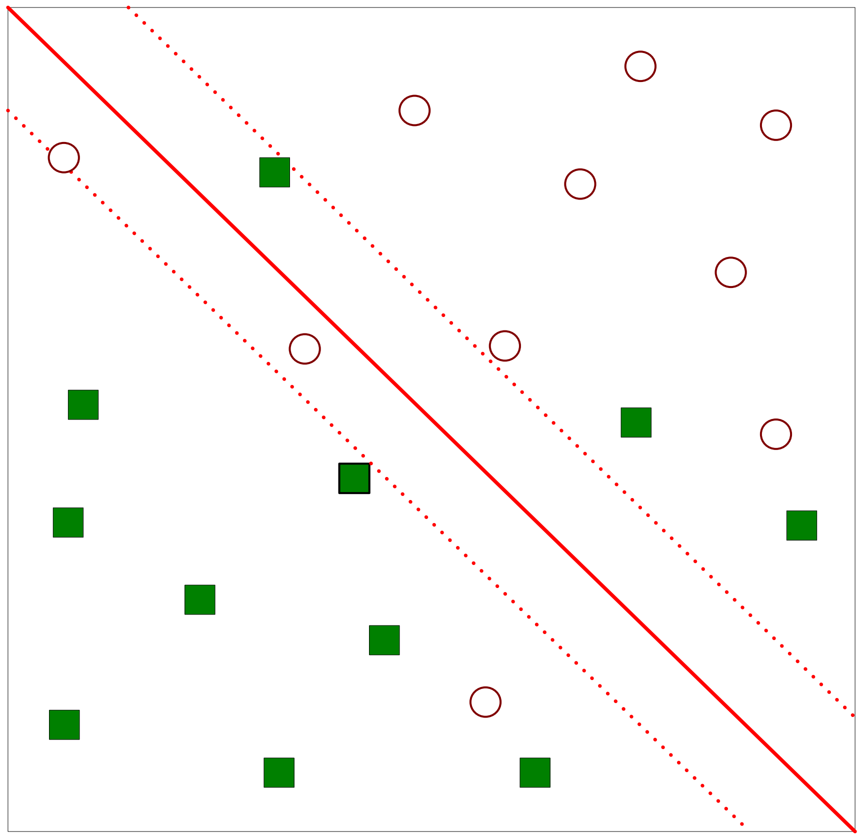
- Learning the model is equivalent to determining *w* and *b.*
- How to find *w* and *b?*
- Objective is to **maximize the margin by minimizing** $L(\vec{w}) = \frac{\|\vec{w}\|^2}{2}$
- Subject to the following constraints

$$y_i = \begin{cases} 1 & \text{if } \vec{\mathrm{w}} \bullet \vec{\mathrm{x}}_{\mathrm{i}} + \mathrm{b} \geq 1 \\ -1 & \text{if } \vec{\mathrm{w}} \bullet \vec{\mathrm{x}}_{\mathrm{i}} + \mathrm{b} \leq -1 \end{cases}$$

- This is a constrained optimization problem: a Quadratic optimization problem, a well-known class of mathematical programming problem, and many algorithms exist for solving them (with many special ones built for SVMs)

Linear SVM: Nonseparable Case

- What if the problem is not linearly separable?

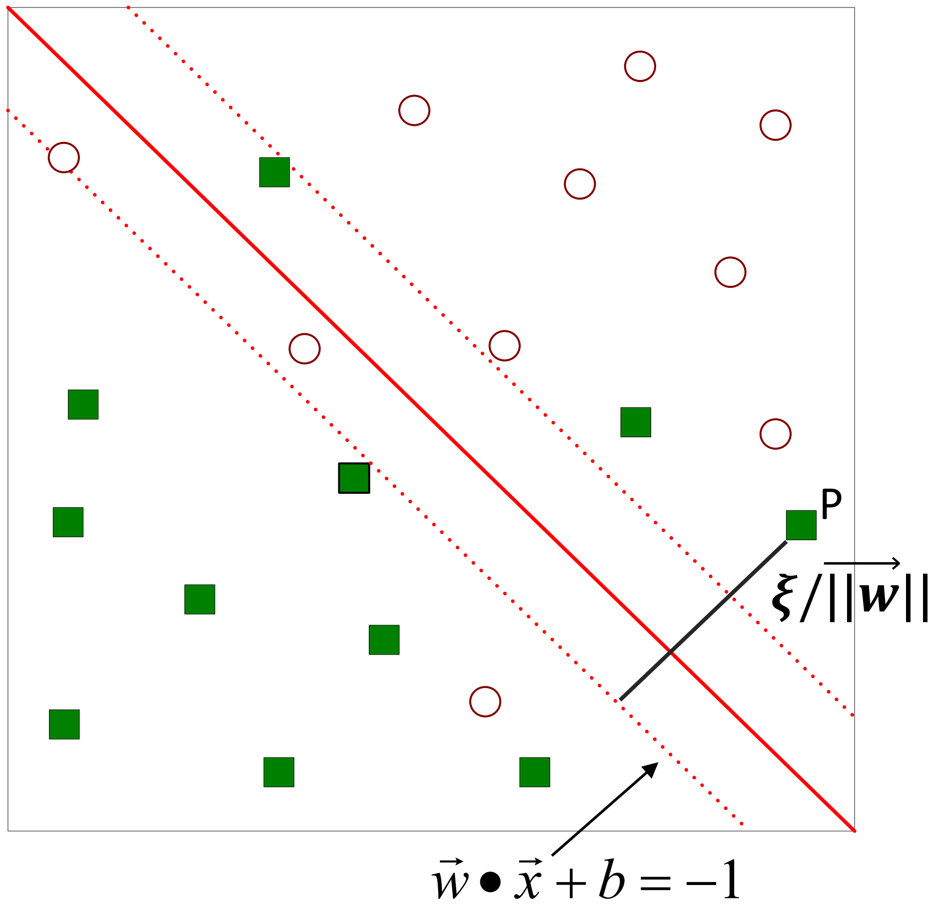


Slack Variables

- The inequality constraints must be relaxed to accommodate the nonlinearly separable data.
- This is done introducing slack variables ξ into the constrains of the optimization problem

$$y_i = \begin{cases} 1 & \text{if } \vec{w} \bullet \vec{x}_i + b \geq 1 - \xi_i \\ -1 & \text{if } \vec{w} \bullet \vec{x}_i + b \leq -1 + \xi_i \end{cases}$$

- ξ provides an estimate of the error of the decision boundary on the misclassified training examples.



Learning a Nonseparable Linear SVM

- Objective to minimize

$$L(w) = \frac{\|\vec{w}\|^2}{2} + C\left(\sum_{i=1}^{N} \xi_i^k\right)$$

- Subject to to the constraints

$$y_i = \begin{cases} 1 & \text{if } \vec{\mathrm{w}} \bullet \vec{\mathrm{x}}_{\mathrm{i}} + \mathrm{b} \geq 1 - \xi_{\mathrm{i}} \\ -1 & \text{if } \vec{\mathrm{w}} \bullet \vec{\mathrm{x}}_{\mathrm{i}} + \mathrm{b} \leq -1 + \xi_{\mathrm{i}} \end{cases}$$

- where *C* and *k* are user-specified parameters representing the penalty of misclassifying the training instances
- Parameter *C* can be viewed as a way to control overfitting: it “trades off” the relative importance of maximizing the margin and fitting the training data.

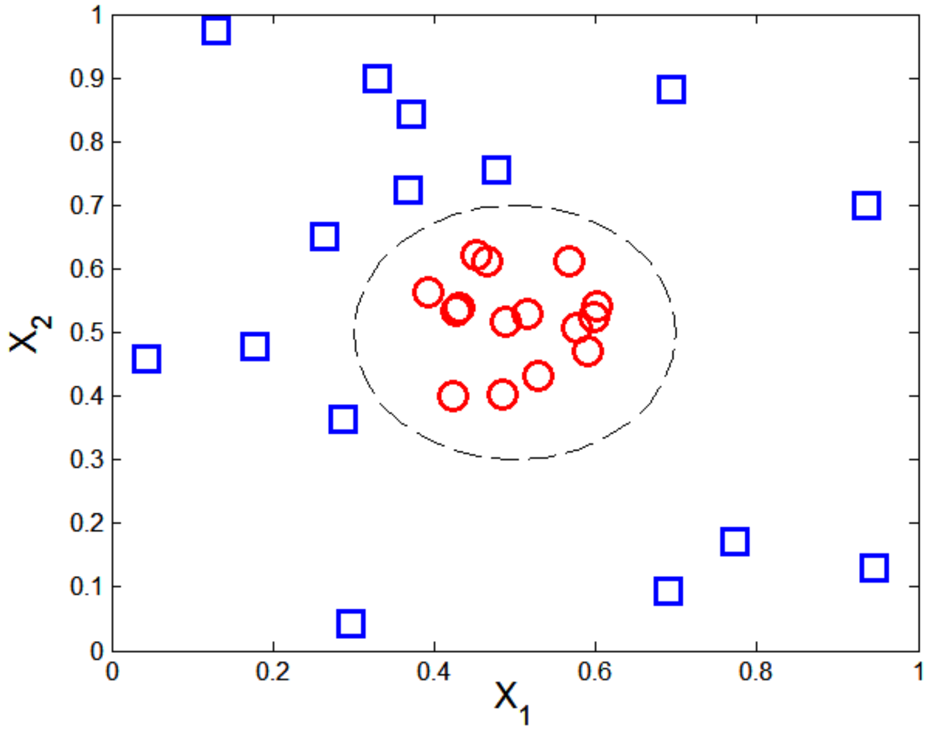
C is a regularization parameter and allows to control overfitting:

- small C allows constraints to be easily ignored → large margin -> misclassification
- large C makes constraints hard to ignore → narrow margin (overfitting)
- C = ∞ enforces all constraints: hard margin

Nonlinear SVM

- What if the decision boundary is not linear?

$$y(x_1, x_2) = \begin{cases} 1 & \text{if } \sqrt{(x_1 - 0.5)^2 + (x_2 - 0.5)^2} > 0.2 \\ -1 & \text{otherwise} \end{cases}$$



Nonlinear SVM

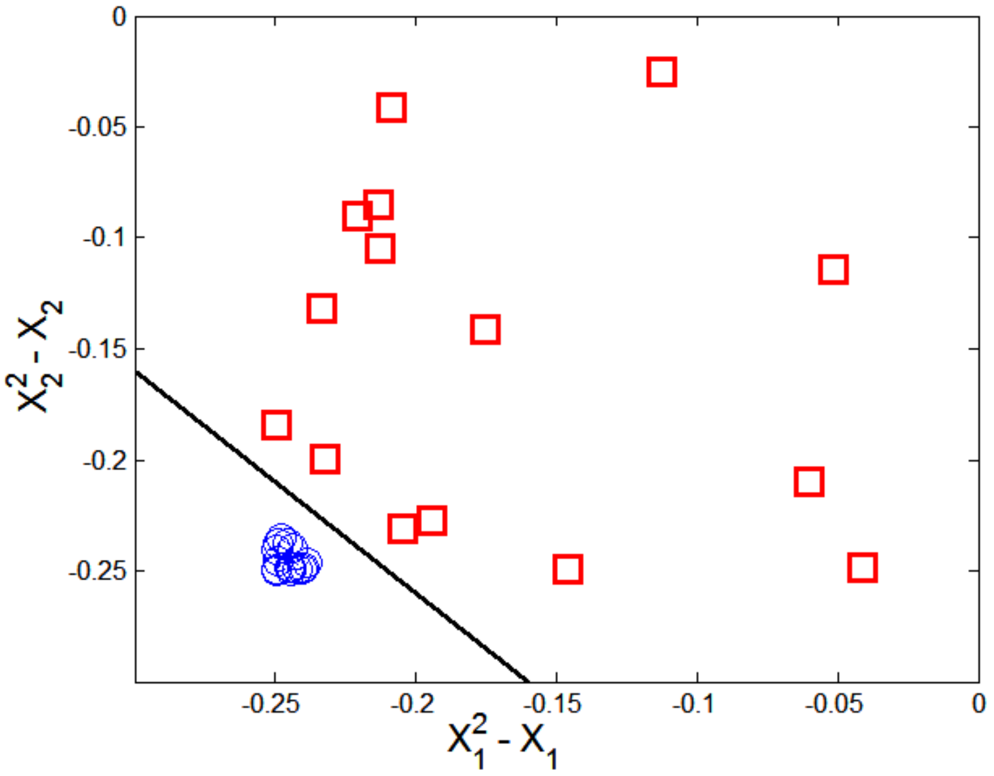
- The trick is to transform the data from its original space x into a new space $\Phi(x)$ so that a linear decision boundary can be used.

$$x_1^2 - x_1 + x_2^2 - x_2 = -0.46.$$

$$\Phi : (x_1, x_2) \longrightarrow (x_1^2, x_2^2, \sqrt{2}x_1, \sqrt{2}x_2, 1).$$

$$w_4 x_1^2 + w_3 x_2^2 + w_2\sqrt{2}x_1 + w_1\sqrt{2}x_2 + w_0 = 0.$$

- Decision boundary $\vec{w} \bullet \Phi(\vec{x}) + b = 0$



References

- Support Vector Machine (SVM). Chapter 5.5. Introduction to Data Mining.