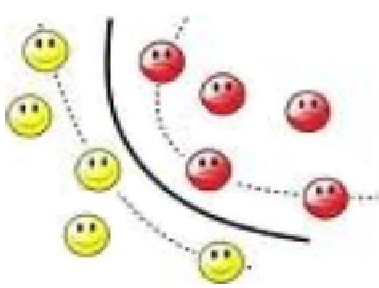
CIS 606 Machine Learning
Spring 2013
Lecture 9
Classification II

Wei Lee Woon and Zeyar Aung



Masdar INSTITUTE

Today's topics

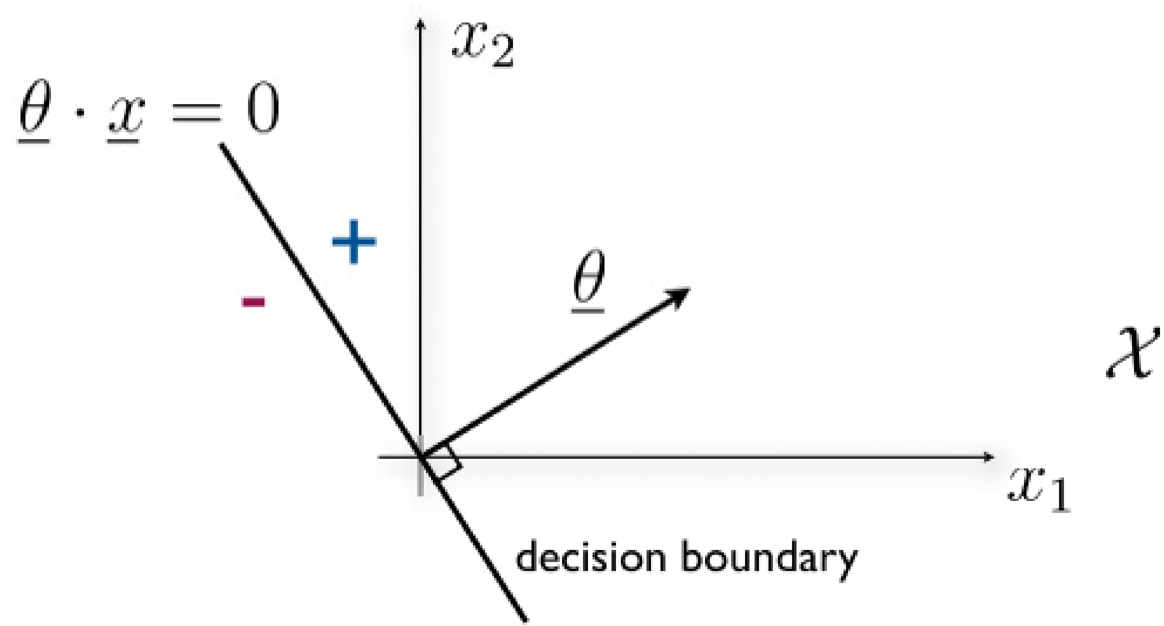
- Perceptron, convergence
  - the prediction game
  - mistakes, margin, and generalization
- Maximum margin classifier -- support vector machine
  - estimation, properties
  - allowing misclassified points

Recall: linear classifiers

- A linear classifier (through origin) with parameters $\underline{\theta}$ divides the space into positive and negative halves

$$f(\underline{x};\underline{\theta}) = \text{sign}\big(\underline{\theta}\cdot\underline{x}\big) = \text{sign}\big(\underbrace{\theta_1 x_1 + \ldots + \theta_d x_d}_{\text{discriminant function}}\big)$$

$$= \begin{cases} +1, & \text{if } \underline{\theta}\cdot\underline{x} > 0 \\ -1, & \text{if } \underline{\theta}\cdot\underline{x} \leq 0 \end{cases}$$



The perceptron algorithm

- A sequence of examples and labels

$$(\underline{x}_t, y_t), \quad t = 1, 2, \ldots$$

- The perceptron algorithm applied to the sequence

$$\text{Initialize: } \underline{\theta} = 0$$

$$\text{For } t = 1, 2, \ldots$$

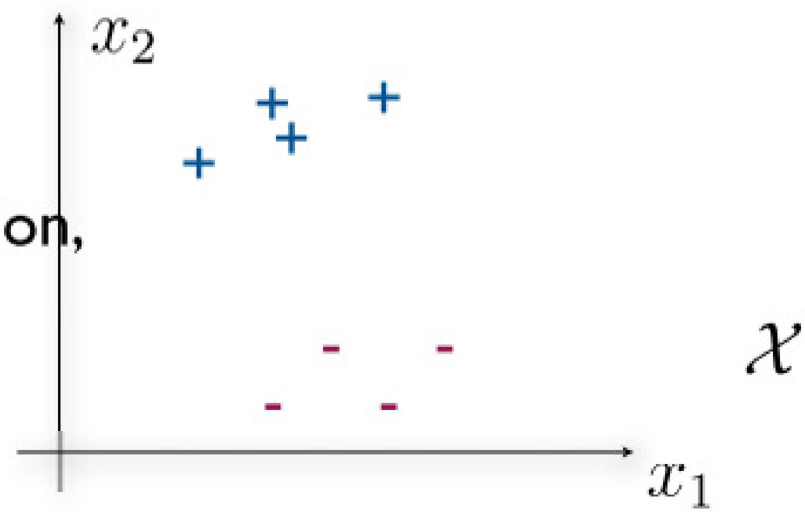
$$\text{if } y_t(\underline{\theta} \cdot \underline{x}_t) \leq 0 \text{ (mistake)}$$

$$\underline{\theta} \leftarrow \underline{\theta} + y_t \underline{x}_t$$

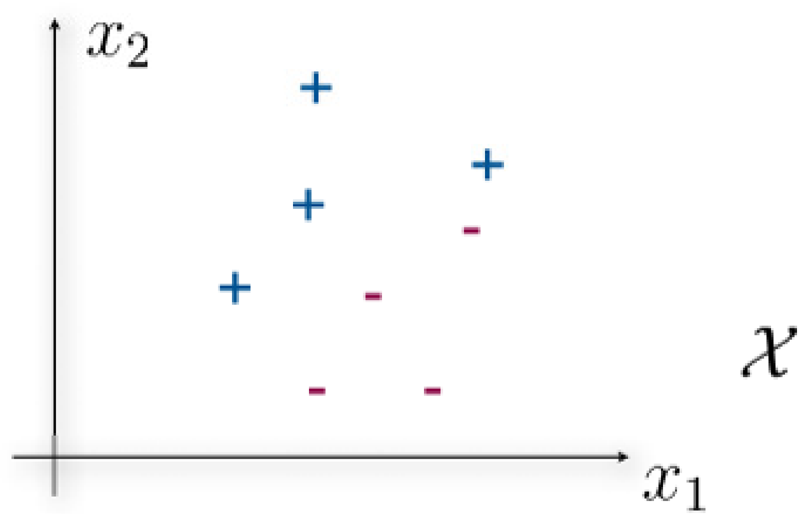
- We would like to bound the number of mistakes that the algorithm makes along the (infinite) sequence

Mistakes and margin

Good separation,
few mistakes

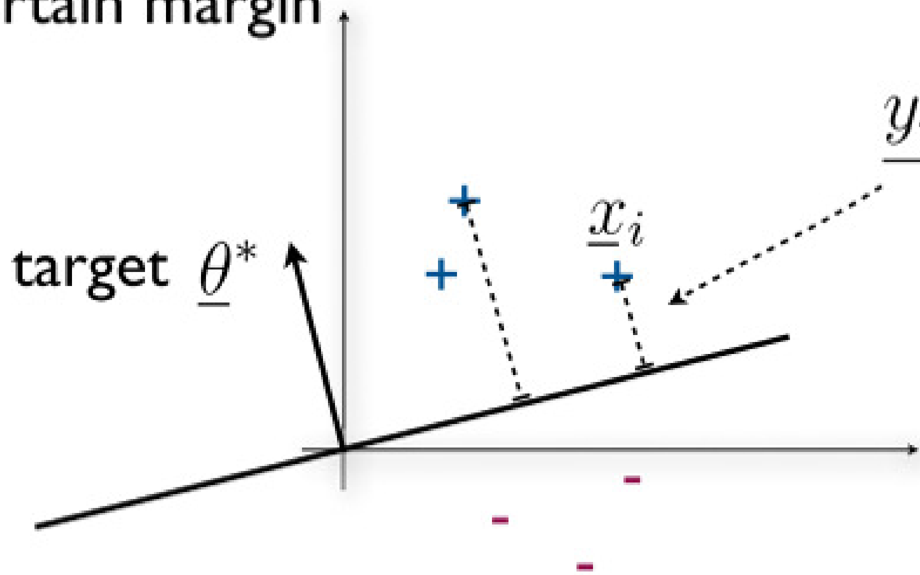


Poor separation,
many mistakes



The target classifier

- We can quantify how hard the problem is by assuming that there exists a target classifier that achieves a certain margin



$$\frac{y_i(\underline{\theta}^* \cdot \underline{x}_i)}{\|\underline{\theta}^*\|} = \gamma_g(i)$$

distance from the point to the separating boundary

- The geometric margin γ_g is the closest distance to the separating boundary $\gamma_g = \min_i \gamma_g(i)$

- Our "target" classifier is one that achieves the largest geometric margin (max-margin classifier)

Perceptron mistake guarantee

- If the sequence of examples and labels is such that there exists $\underline{\theta}^*$ with geometric margin γ_g and $\|\underline{x}_i\| \leq R$ then the perceptron algorithm makes at most
$$\frac{R^2}{\gamma_g^2}$$
mistakes along the (infinite) sequence!

- Key points
  - large geometric margin relative to the norm of the examples implies few mistakes
  - the result does not depend on the dimension of the examples (the number of parameters)

Mistake guarantee: proof

- We show that after k updates (mistakes),

$$\begin{aligned} \underline{\theta}^{(k)} \cdot \underline{\theta}^* &\geq k\gamma_g \|\underline{\theta}^*\| \\ \|\underline{\theta}^{(k)}\|^2 &\leq kR^2 \end{aligned}$$

Mistake guarantee: proof

- We show that after k updates (mistakes),

$$\begin{aligned} \underline{\theta}^{(k)} \cdot \underline{\theta}^* &\geq k\gamma_g \|\underline{\theta}^*\| \\ \|\underline{\theta}^{(k)}\|^2 &\leq kR^2 \end{aligned}$$

- Let the kth mistake be on the ith example

$$\begin{aligned} \underline{\theta}^{(k)} \cdot \underline{\theta}^* &= [\underline{\theta}^{(k-1)} + y_i \underline{x}_i] \cdot \underline{\theta}^* \\ &= \underline{\theta}^{(k-1)} \cdot \underline{\theta}^* + y_i \underline{x}_i \cdot \underline{\theta}^* \\ &\geq \underline{\theta}^{(k-1)} \cdot \underline{\theta}^* + \gamma_g \|\underline{\theta}^*\| \end{aligned}$$

Mistake guarantee: proof

- We show that after k updates (mistakes),

$$
\begin{aligned}
\underline{\theta}^{(k)} \cdot \underline{\theta}^{*} &\geq k\gamma_g \|\underline{\theta}^{*}\| \\
\|\underline{\theta}^{(k)}\|^2 &\leq kR^2
\end{aligned}
$$

- Let the kth mistake be on the ith example

$$
\begin{aligned}
\|\underline{\theta}^{(k)}\|^2 &= \|\underline{\theta}^{(k-1)} + y_i \underline{x}_i\|^2 \\
&= \|\underline{\theta}^{(k-1)}\|^2 + 2y_i \underline{\theta} \cdot \underline{x}_i + \|\underline{x}_i\|^2 \\
&\leq \|\underline{\theta}^{(k-1)}\|^2 + \|\underline{x}_i\|^2 \\
&\leq \|\underline{\theta}^{(k-1)}\|^2 + R^2
\end{aligned}
$$

Mistake guarantee: proof

- We have shown that after k updates (mistakes),

$$\begin{aligned} \underline{\theta}^{(k)} \cdot \underline{\theta}^* &\geq k\gamma_g \|\underline{\theta}^*\| \\ \|\underline{\theta}^{(k)}\|^2 &\leq kR^2 \end{aligned}$$

- As a result,

$$1 \geq \frac{\overbrace{\underline{\theta}^{(k)} \cdot \underline{\theta}^*}^{\text{cosine}}}{\|\underline{\theta}^{(k)}\|\|\underline{\theta}^*\|} \geq \frac{k\gamma_g}{\sqrt{k}R}$$

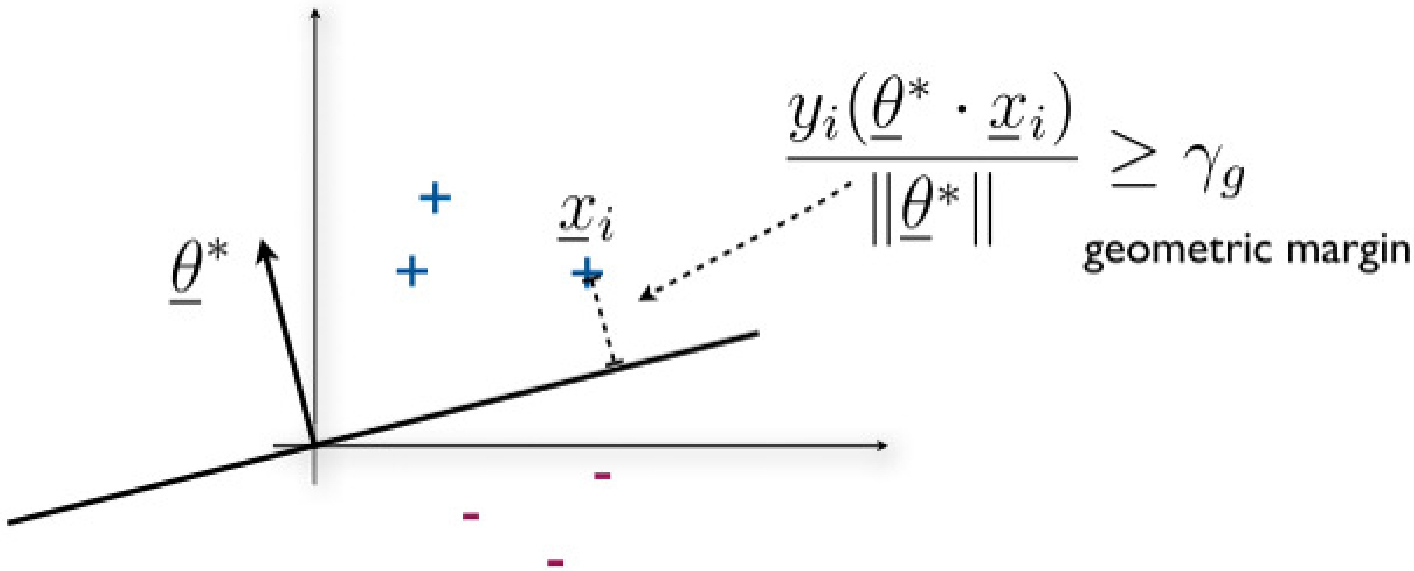
$$\Rightarrow k \leq \frac{R^2}{\gamma_g^2}$$

Summary (perceptron)

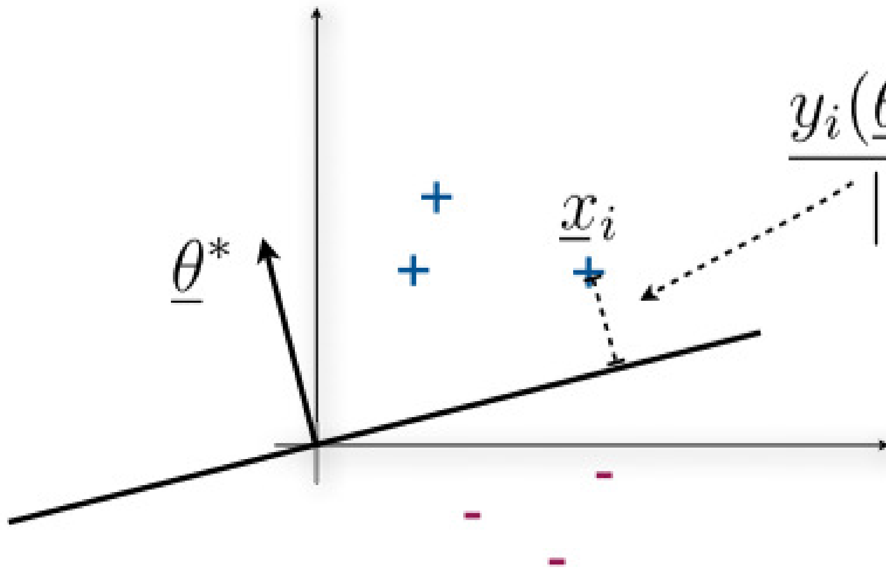
- By analyzing the simple perceptron algorithm, we were able to relate the number of mistakes, geometric margin, and generalization
- The perceptron algorithm converges to a classifier close to the max-margin target classifier

  In cases where we are given a fixed set of training examples, and they are linearly separable, we can find and use the maximum margin classifier directly

Maximum margin classifier



Maximum margin classifier



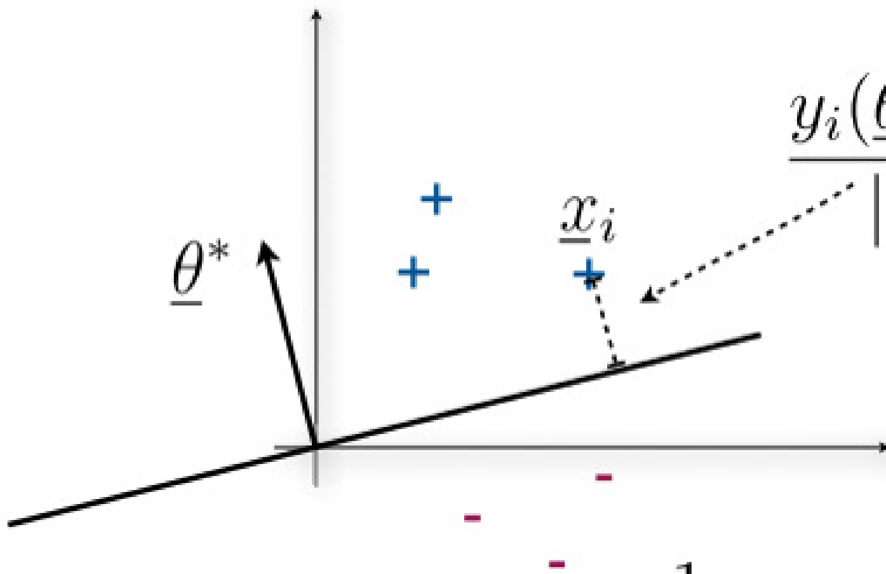
$$\frac{y_i(\underline{\theta}^* \cdot \underline{x}_i)}{\|\underline{\theta}^*\|} \geq \gamma_g$$

geometric margin

$$\text{maximize } \gamma_g \text{ subject to}$$

$$\text{To find } \underline{\theta}^* : \quad \frac{y_i(\underline{\theta} \cdot \underline{x}_i)}{\|\underline{\theta}\|} \geq \gamma_g, \quad i = 1, \ldots, n$$

Maximum margin classifier



$$\frac{y_i(\underline{\theta}^* \cdot \underline{x}_i)}{\|\underline{\theta}^*\|} \geq \gamma_g$$

geometric margin

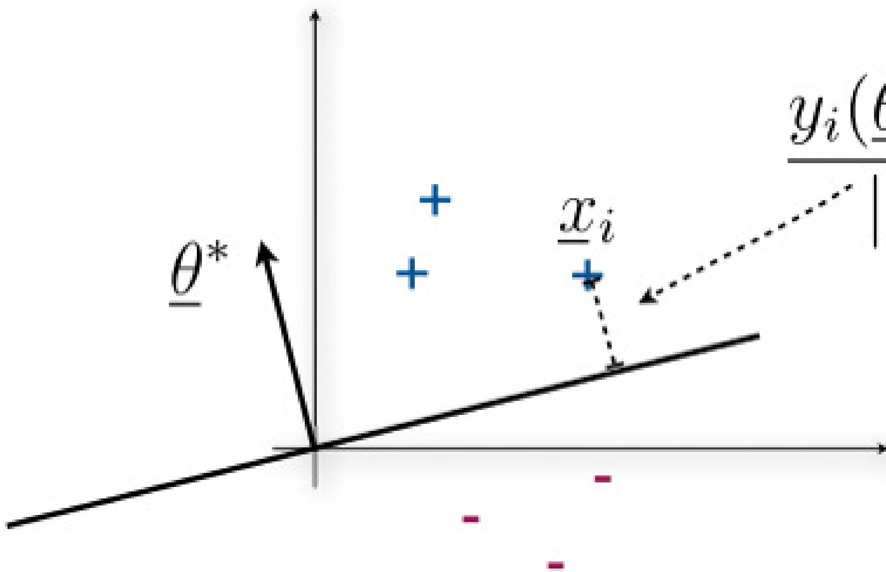
$$\gamma_g = \frac{1}{\|\underline{\theta}^*\|}$$

To find $\underline{\theta}^*$:

$$\text{maximize } \frac{1}{\|\underline{\theta}\|} \text{ subject to}$$

$$\frac{y_i(\underline{\theta} \cdot \underline{x}_i)}{\|\underline{\theta}\|} \geq \frac{1}{\|\underline{\theta}\|}, \quad i = 1, \ldots, n$$

Maximum margin classifier



$$\frac{y_i(\underline{\theta}^* \cdot \underline{x}_i)}{\|\underline{\theta}^*\|} \geq \gamma_g$$

geometric margin

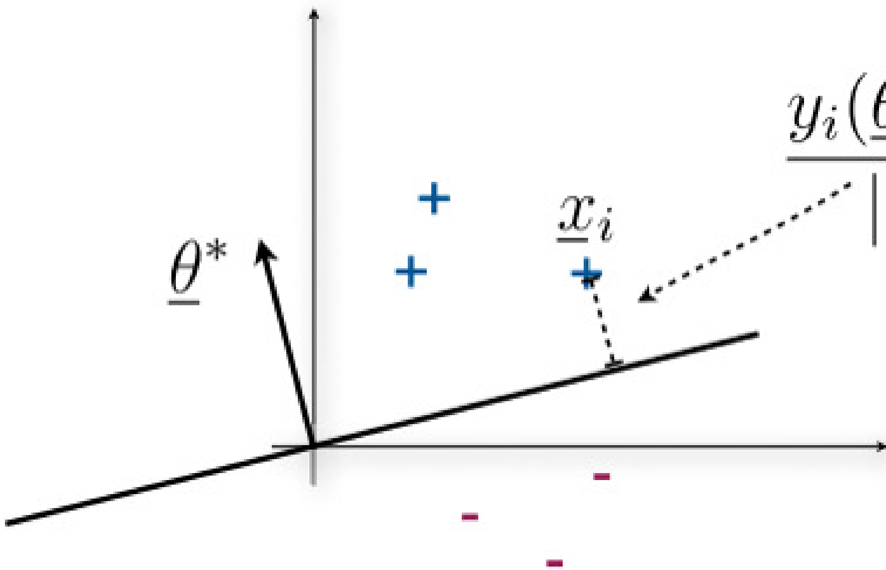
$$\gamma_g = \frac{1}{\|\underline{\theta}^*\|}$$

To find $\underline{\theta}^*$:

$$\text{maximize } \frac{1}{\|\underline{\theta}\|} \text{ subject to}$$

$$y_i(\underline{\theta} \cdot \underline{x}_i) \geq 1, \quad i = 1, \ldots, n$$

Maximum margin classifier



$$\frac{y_i(\underline{\theta}^* \cdot \underline{x}_i)}{\|\underline{\theta}^*\|} \geq \gamma_g$$

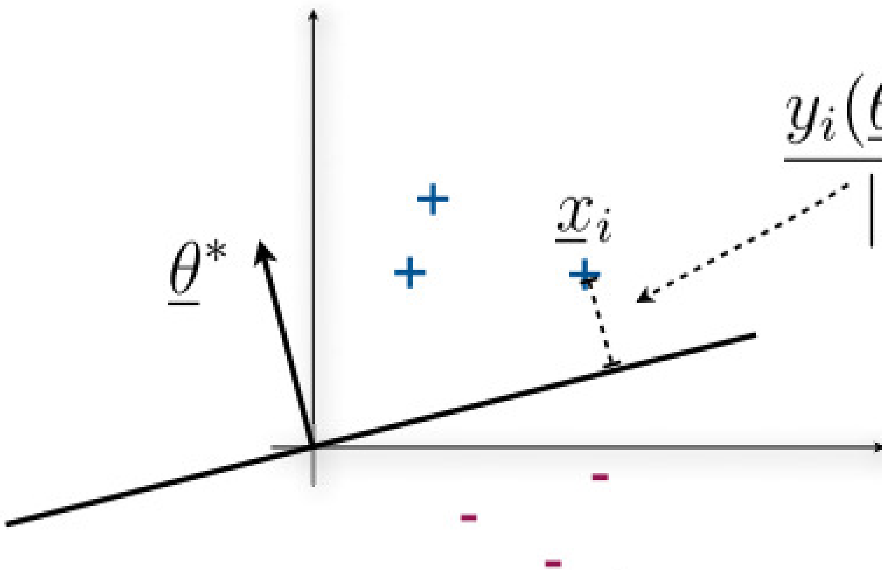
geometric margin

$$\gamma_g = \frac{1}{\|\underline{\theta}^*\|}$$

To find $\underline{\theta}^*$: minimize $\|\underline{\theta}\|$ subject to

$$y_i(\underline{\theta} \cdot \underline{x}_i) \geq 1, \quad i = 1, \ldots, n$$

Support vector machine



$$\frac{y_i(\underline{\theta}^* \cdot \underline{x}_i)}{\|\underline{\theta}^*\|} \geq \gamma_g$$

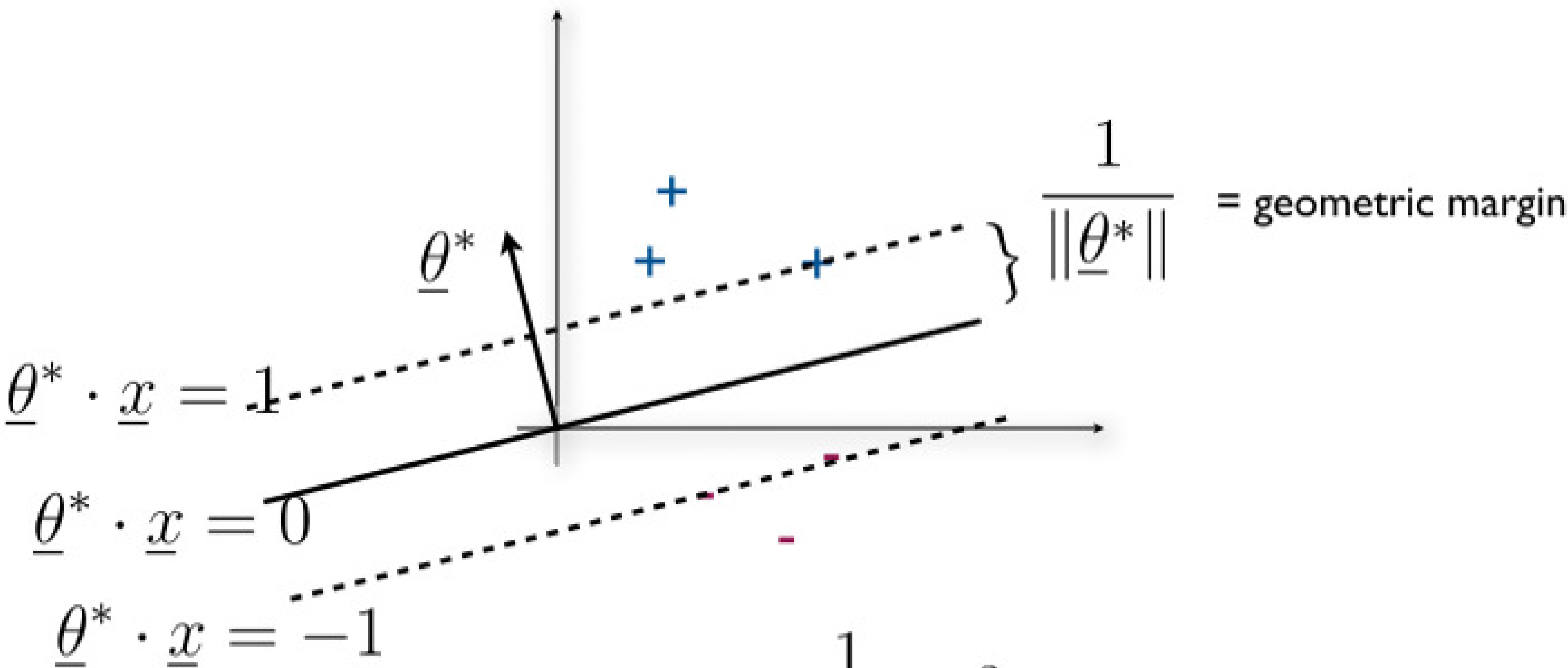
geometric margin

$$\gamma_g = \frac{1}{\|\underline{\theta}^*\|}$$

$$\text{To find } \underline{\theta}^* : \quad \begin{array}{l} \text{minimize } \dfrac{1}{2}\|\underline{\theta}\|^2 \text{ subject to} \\ y_i(\underline{\theta} \cdot \underline{x}_i) \geq 1, \quad i = 1, \ldots, n \end{array}$$

- This is a quadratic programming problem (quadratic objective, linear constraints)
- The solution is unique, typically obtained in the dual

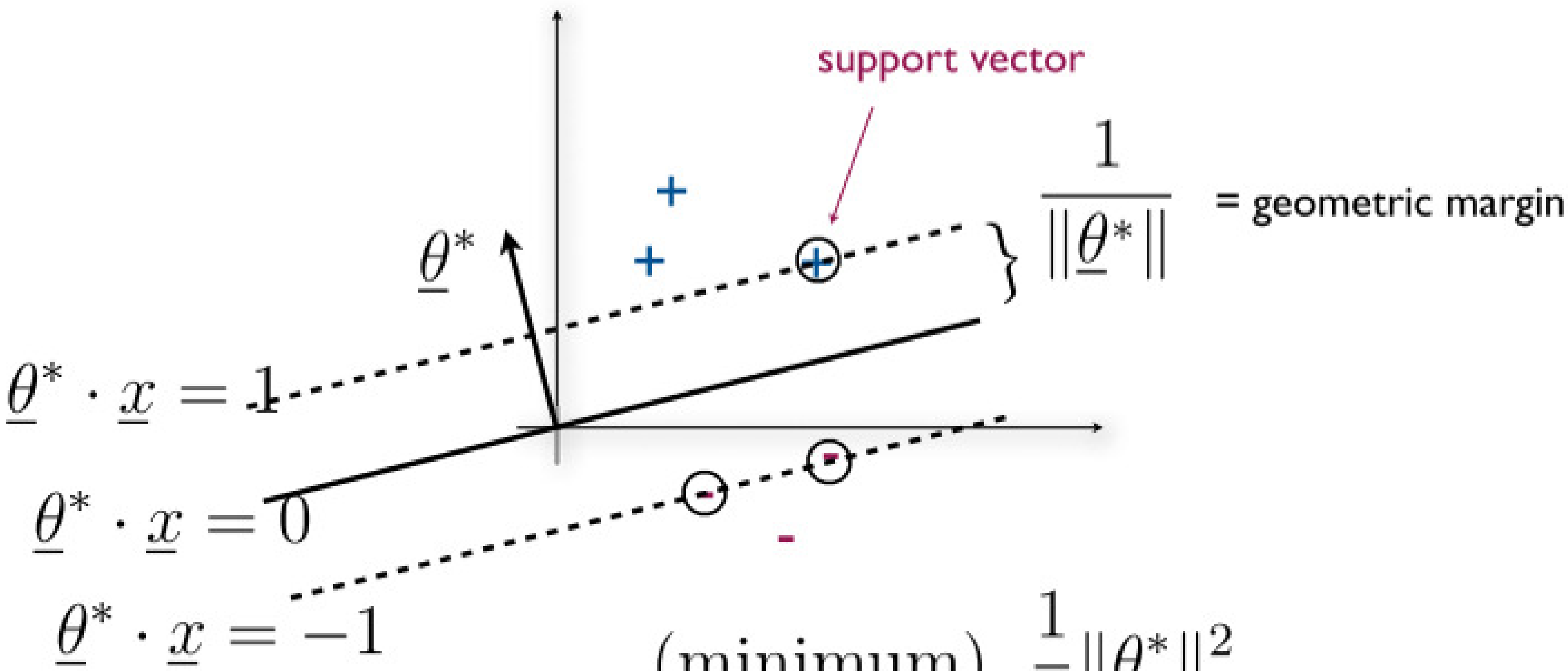
Support vector machine



To find $\underline{\theta}^*$:

$$\text{minimize } \frac{1}{2}\|\underline{\theta}\|^2 \text{ subject to}$$
$$y_i(\underline{\theta} \cdot \underline{x}_i) \geq 1, \quad i = 1, \ldots, n$$

Support vector machine



$$\text{(minimum)} \quad \frac{1}{2}\|\underline{\theta}^*\|^2$$

$$y_1(\underline{\theta}^* \cdot \underline{x}_1) = 1$$

$$y_2(\underline{\theta}^* \cdot \underline{x}_2) > 1$$

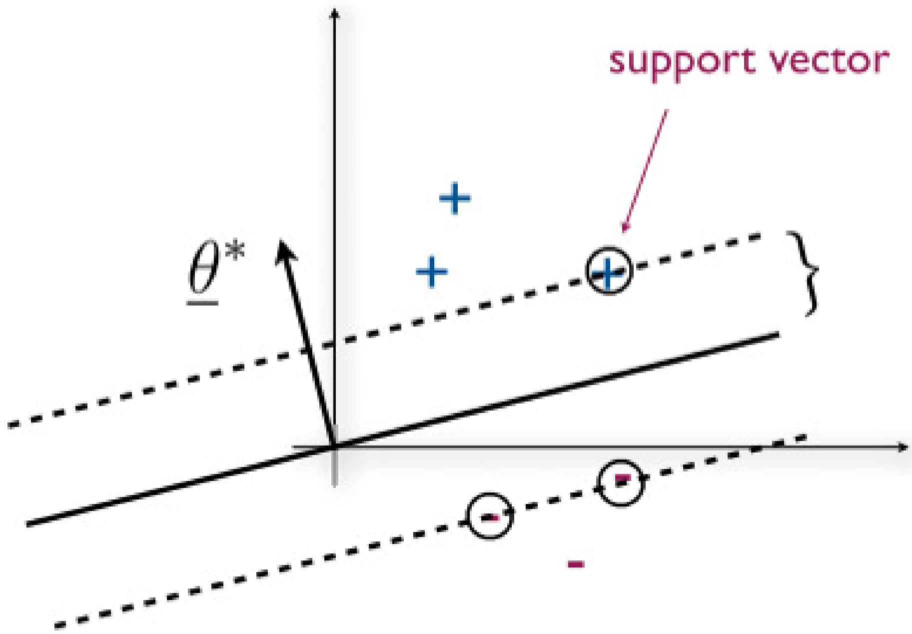
$$y_3(\underline{\theta}^* \cdot \underline{x}_3) = 1$$

$$\cdots$$

active constraints
= support vectors

The solution is
sparse

Is sparse solution good?



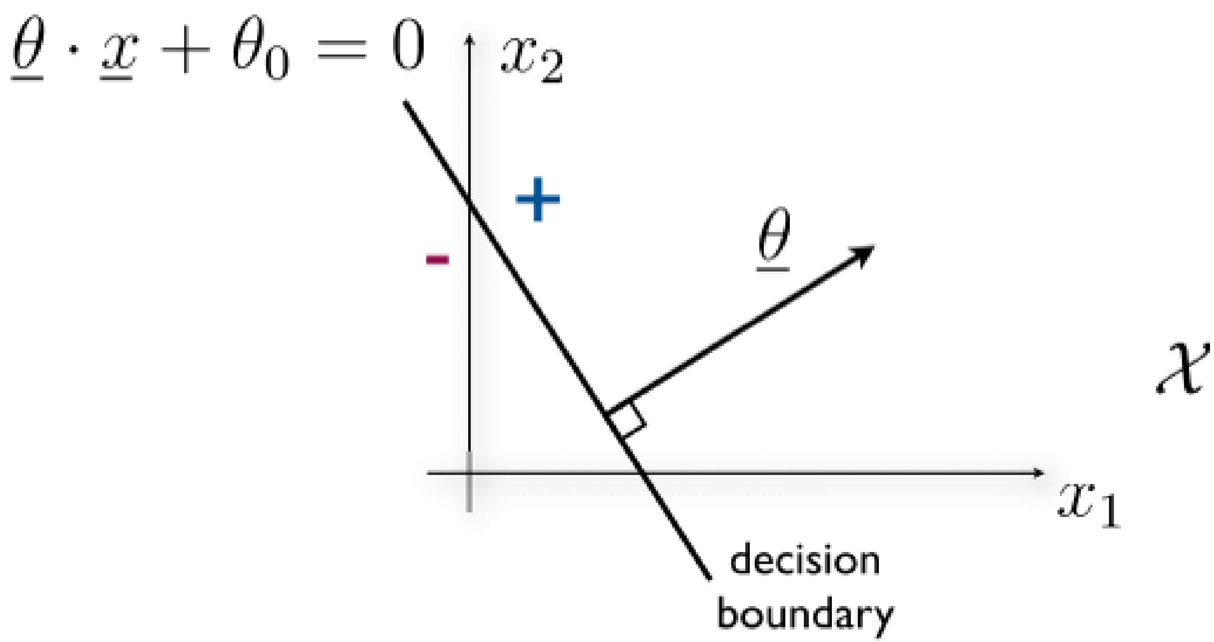
- We can simulate test performance by evaluating Leave-One-Out Cross-Validation error

$$\mathrm{LOOCV}(\underline{\theta}^*) \leq \frac{\#\text{ of support vectors}}{n}$$

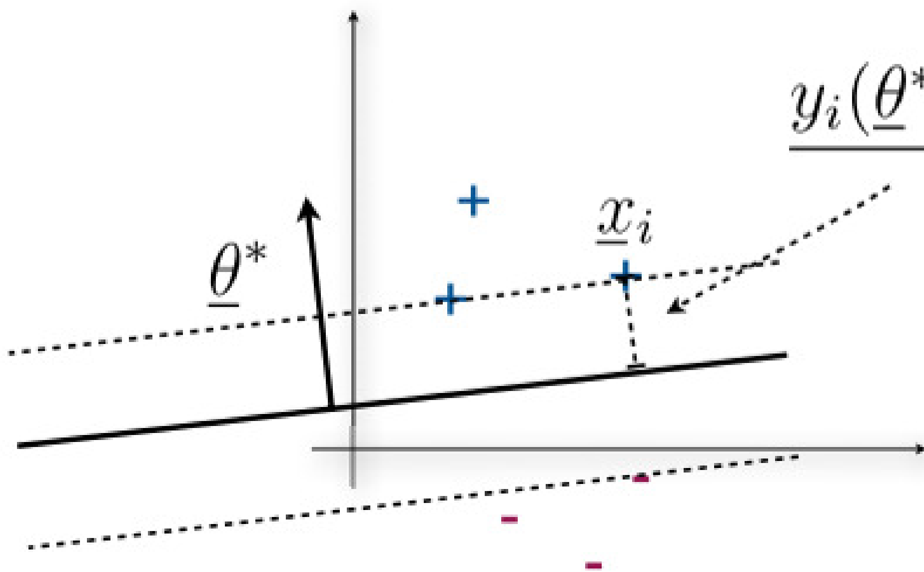
Linear classifiers (with offset)

- A linear classifier with parameters $(\underline{\theta}, \theta_0)$

$$
\begin{aligned}
f(\underline{x}; \underline{\theta}, \theta_0) &= \operatorname{sign}\big(\, \underline{\theta} \cdot \underline{x} + \theta_0 \,\big) \\
&= \begin{cases} +1, & \text{if } \underline{\theta} \cdot \underline{x} + \theta_0 > 0 \\ -1, & \text{if } \underline{\theta} \cdot \underline{x} + \theta_0 \leq 0 \end{cases}
\end{aligned}
$$



Support vector machine



$$\frac{y_i(\underline{\theta}^* \cdot \underline{x}_i + \theta_0^*)}{\|\underline{\theta}^*\|} \geq \gamma_g$$

geometric margin

$$\gamma_g = \frac{1}{\|\underline{\theta}^*\|}$$

To find $\underline{\theta}^*, \theta_0^*$:

$$\text{minimize } \frac{1}{2}\|\underline{\theta}\|^2 \text{ subject to}$$

$$y_i(\underline{\theta} \cdot \underline{x}_i + \theta_0) \geq 1, \;\; i = 1, \ldots, n$$

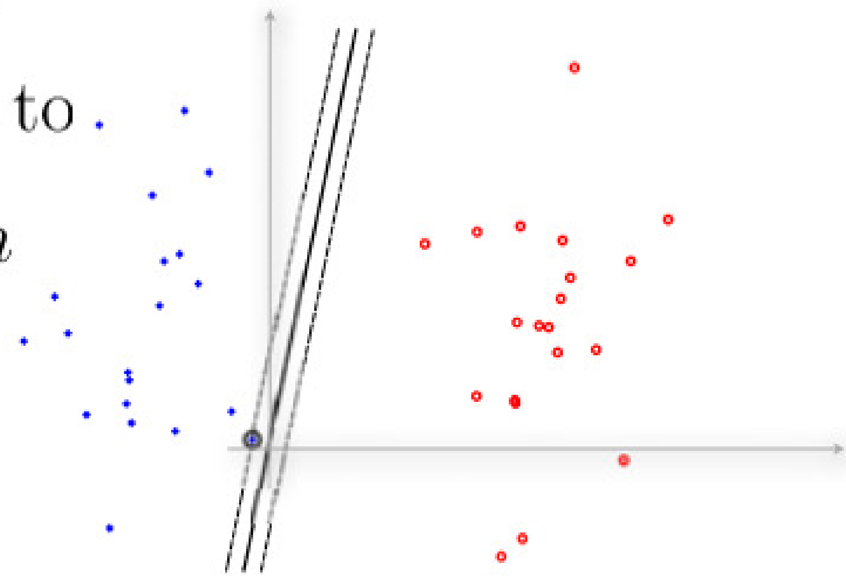
- Still a quadratic programming problem (quadratic objective, linear constraints)

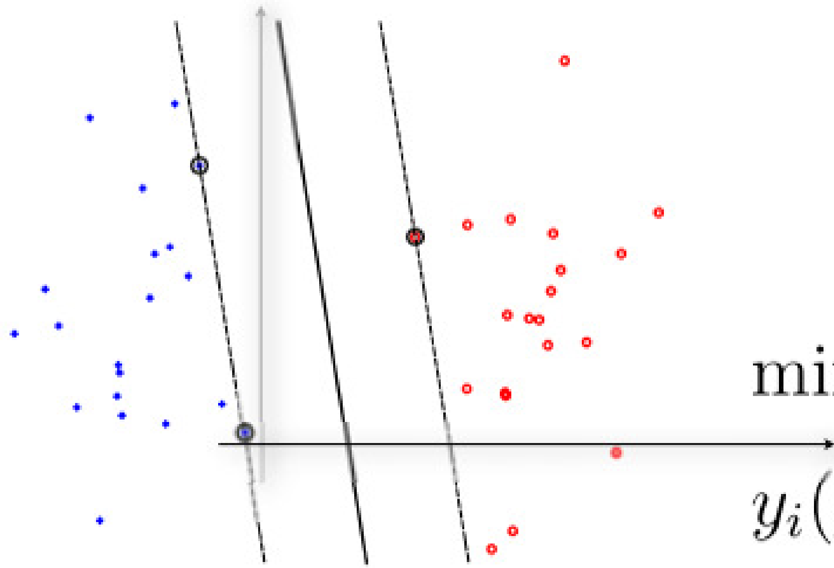
The impact of offset

- Adding the offset parameter to the linear classifier can substantially increase the margin

$$\text{minimize} \quad \frac{1}{2}\|\underline{\theta}\|^2 \quad \text{subject to}$$

$$y_i(\underline{\theta} \cdot \underline{x}_i) \geq 1, \quad i = 1, \ldots, n$$





$$\text{minimize} \quad \frac{1}{2}\|\underline{\theta}\|^2 \quad \text{subject to}$$

$$y_i(\underline{\theta} \cdot \underline{x}_i + \theta_0) \geq 1, \quad i = 1, \ldots, n$$

Support vector machine

- Several desirable properties
  - maximizes the margin on the training set ($\approx$ good generalization)
  - the solution is unique and sparse ($\approx$ good generalization)
- But...
  - the solution is sensitive to outliers, labeling errors, as they may drastically change the resulting max-margin boundary
  - if the training set is not linearly separable, there's no solution!

Support vector machine

- Relaxed quadratic optimization problem

penalty for constraint violation

$$\begin{aligned} \text{minimize} \quad \frac{1}{2}\|\underline{\theta}\|^2 \quad + \quad & \overline{C\sum_{i=1}^{n}\xi_i} \quad \text{subject to} \\ y_i(\underline{\theta}\cdot\underline{x}_i+\theta_0) \quad \geq \quad & 1-\xi_i, \quad i=1,\ldots,n \\ \xi_i \quad \geq \quad & 0, \quad i=1,\ldots,n \end{aligned}$$

slack variables permit us to violate some of the margin constraints

Support vector machine

- Relaxed quadratic optimization problem

penalty for constraint violation

$$\text{minimize} \;\; \frac{1}{2}\|\underline{\theta}\|^2 \;\; + \;\; \overline{C\sum_{i=1}^{n}\xi_i} \;\; \text{subject to}$$

$$\begin{aligned} y_i(\underline{\theta}\cdot\underline{x}_i+\theta_0) &\geq 1-\xi_i, \;\; i=1,\ldots,n \\ \xi_i &\geq 0, \;\; i=1,\ldots,n \end{aligned}$$

slack variables permit us to violate some of the margin constraints

large $C \Rightarrow$ few (if any) violations

small $C \Rightarrow$ many violations

Support vector machine

- Relaxed quadratic optimization problem

$$\text{minimize } \frac{1}{2}\|\underline{\theta}\|^2 + \overline{C\sum_{i=1}^{n}\xi_i}^{\text{penalty for constraint violation}} \text{ subject to}$$

$$\begin{aligned} y_i(\underline{\theta}\cdot\underline{x}_i + \theta_0) &\geq 1 - \xi_i,\;\; i = 1,\ldots,n \\ \xi_i &\geq 0,\;\; i = 1,\ldots,n \end{aligned}$$

slack variables permit us to violate some of the margin constraints

large $C \Rightarrow$ few (if any) violations

small $C \Rightarrow$ many violations

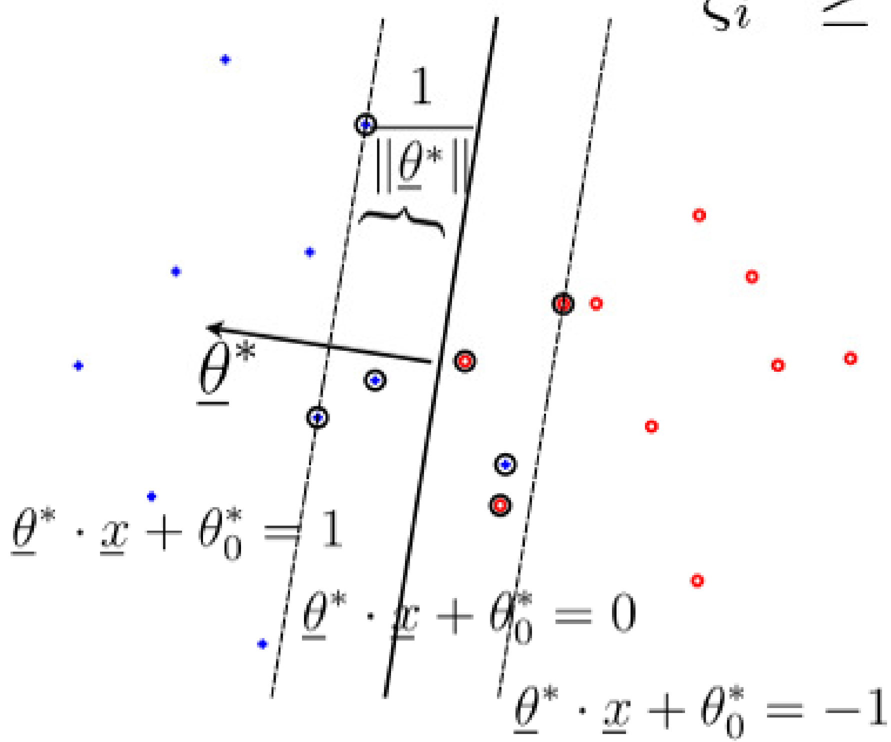
we can still interpret the margin as $1/\|\underline{\theta}^*\|$

Support vector machine

- Relaxed quadratic optimization problem

$$\text{minimize} \quad \frac{1}{2}\|\underline{\theta}\|^2 \quad + \quad C\sum_{i=1}^{n}\xi_i \quad \text{subject to}$$

$$\begin{aligned} y_i(\underline{\theta}\cdot\underline{x}_i+\theta_0) &\geq 1-\xi_i, \;\; i=1,\ldots,n \\ \xi_i &\geq 0, \;\; i=1,\ldots,n \end{aligned}$$

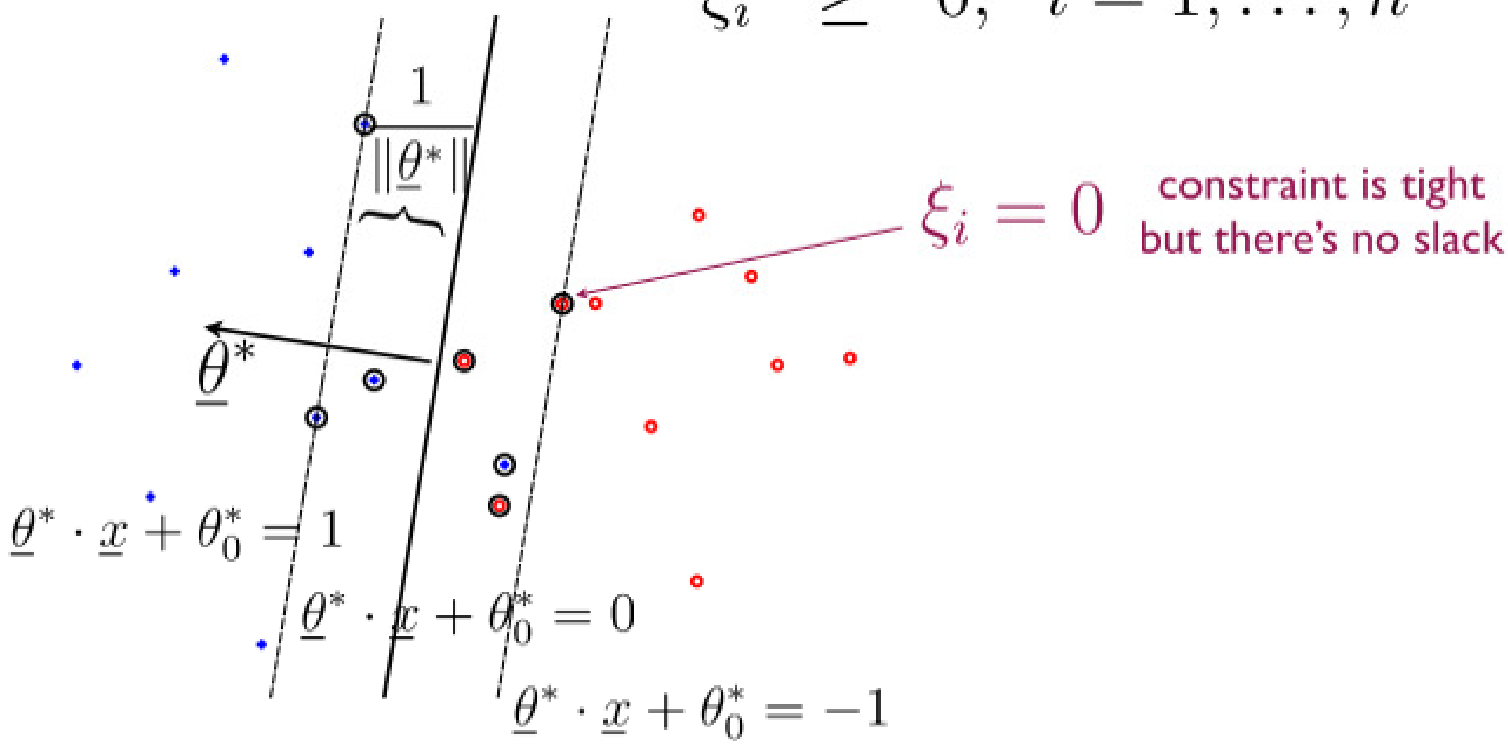


Support vectors and slack

- The solution now has three types of support vectors

$$\text{minimize} \quad \frac{1}{2}\|\underline{\theta}\|^2 \quad + \quad C\sum_{i=1}^{n}\xi_i \quad \text{subject to}$$

$$\begin{aligned} y_i(\underline{\theta}\cdot\underline{x}_i+\theta_0) \quad &\geq \quad 1-\xi_i, \;\; i=1,\ldots,n \\ \xi_i \quad &\geq \quad 0, \;\; i=1,\ldots,n \end{aligned}$$

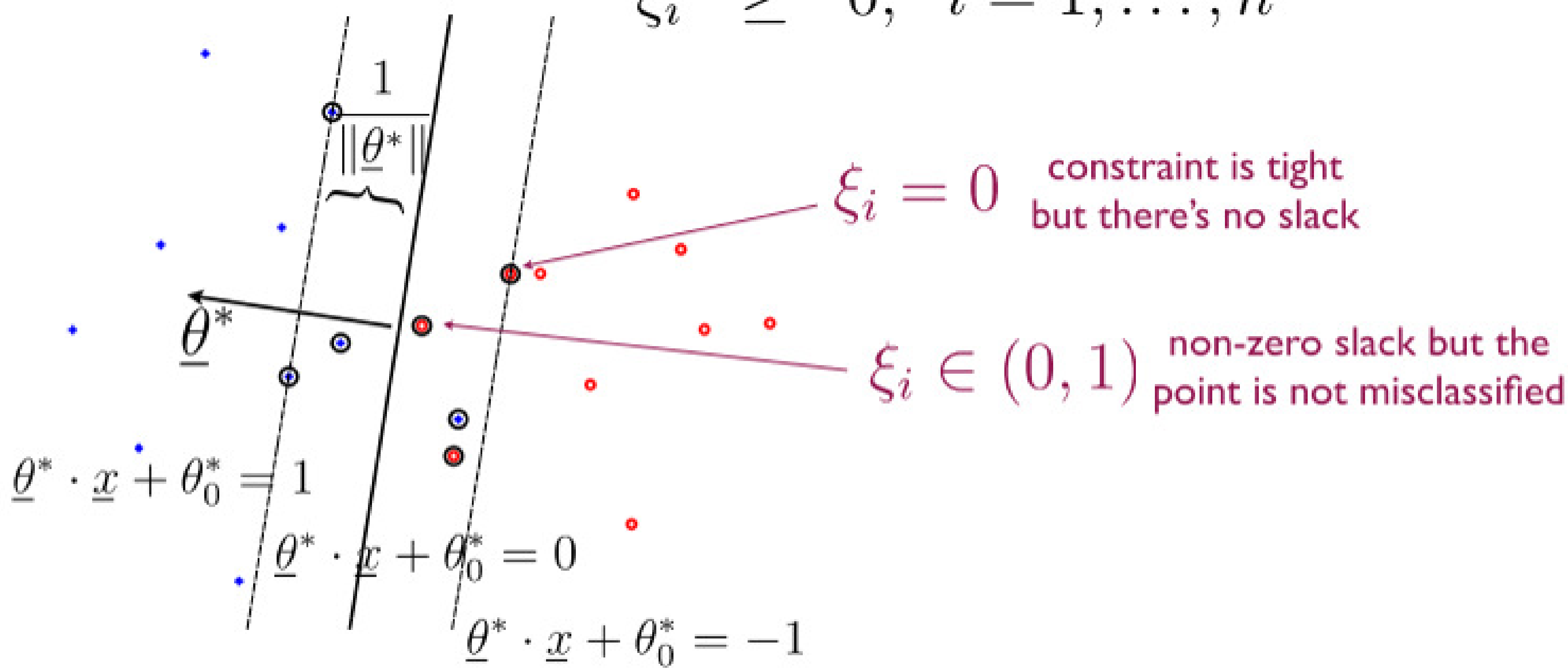


Support vectors and slack

- The solution now has three types of support vectors

$$\text{minimize} \quad \frac{1}{2}\|\underline{\theta}\|^2 \quad + \quad C\sum_{i=1}^{n}\xi_i \quad \text{subject to}$$

$$\begin{aligned} y_i(\underline{\theta}\cdot\underline{x}_i+\theta_0) &\geq 1-\xi_i, \;\; i=1,\ldots,n \\ \xi_i &\geq 0, \;\; i=1,\ldots,n \end{aligned}$$

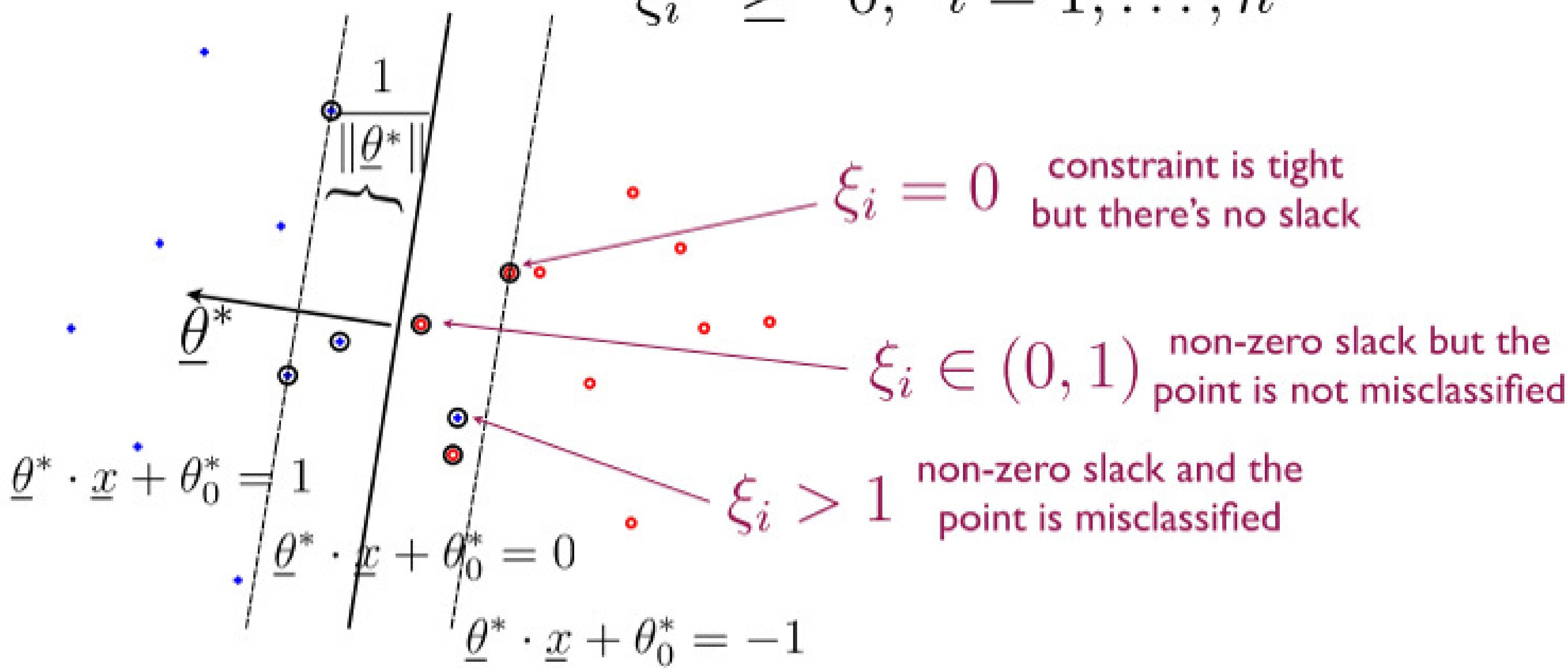


Support vectors and slack

- The solution now has three types of support vectors

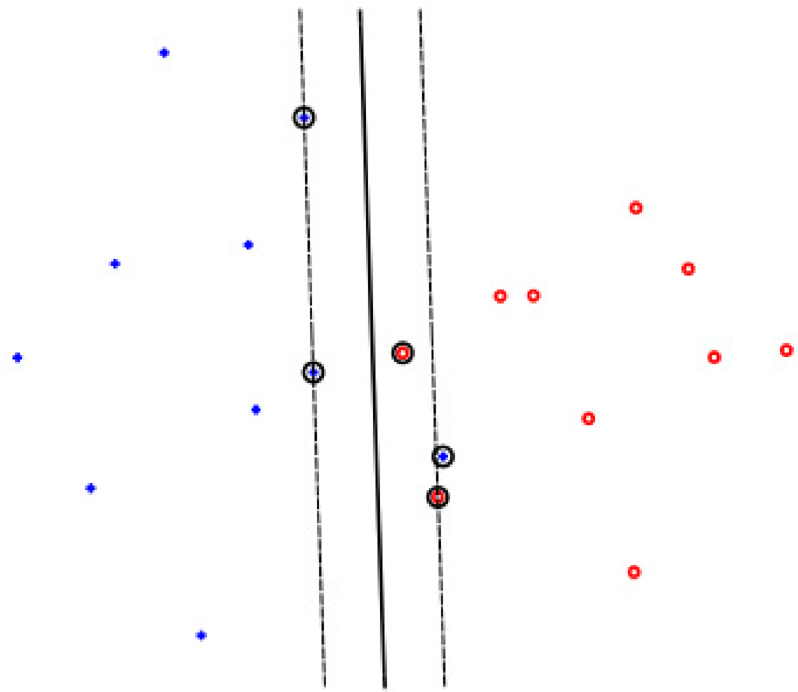
$$\text{minimize} \quad \frac{1}{2}\|\underline{\theta}\|^2 \; + \; C\sum_{i=1}^{n}\xi_i \quad \text{subject to}$$

$$\begin{aligned} y_i(\underline{\theta}\cdot\underline{x}_i+\theta_0) &\geq 1-\xi_i,\;\; i=1,\ldots,n \\ \xi_i &\geq 0,\;\; i=1,\ldots,n \end{aligned}$$



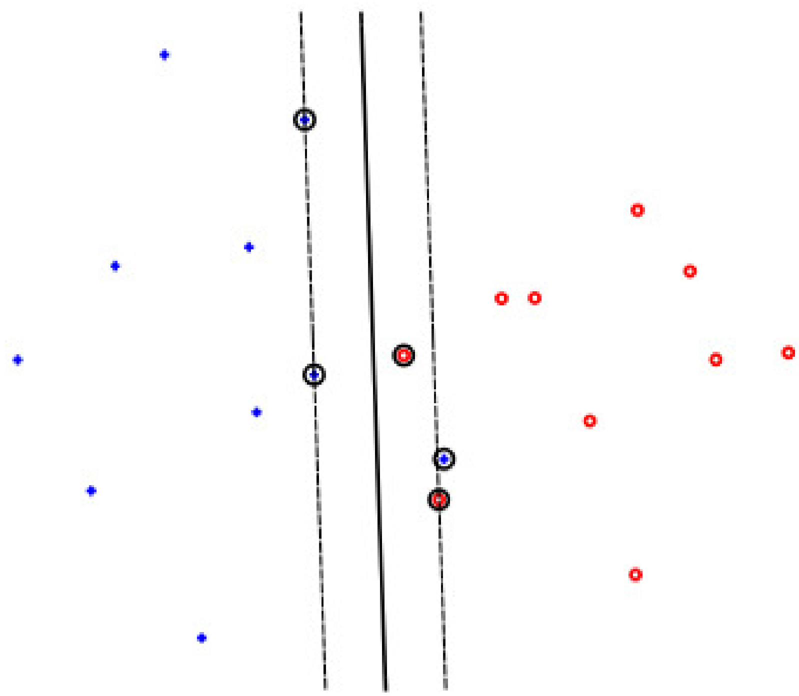
Examples

- C=100



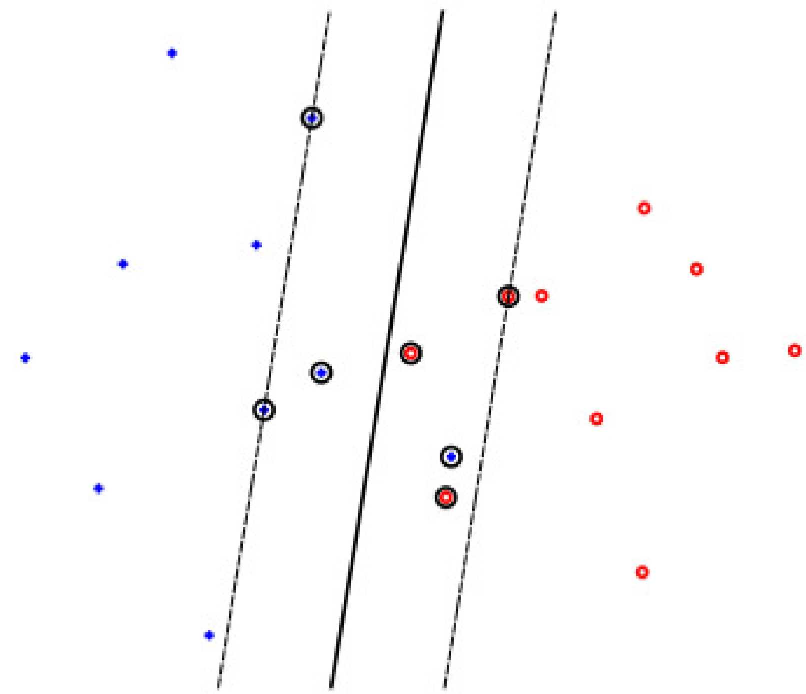
Examples

- C=10



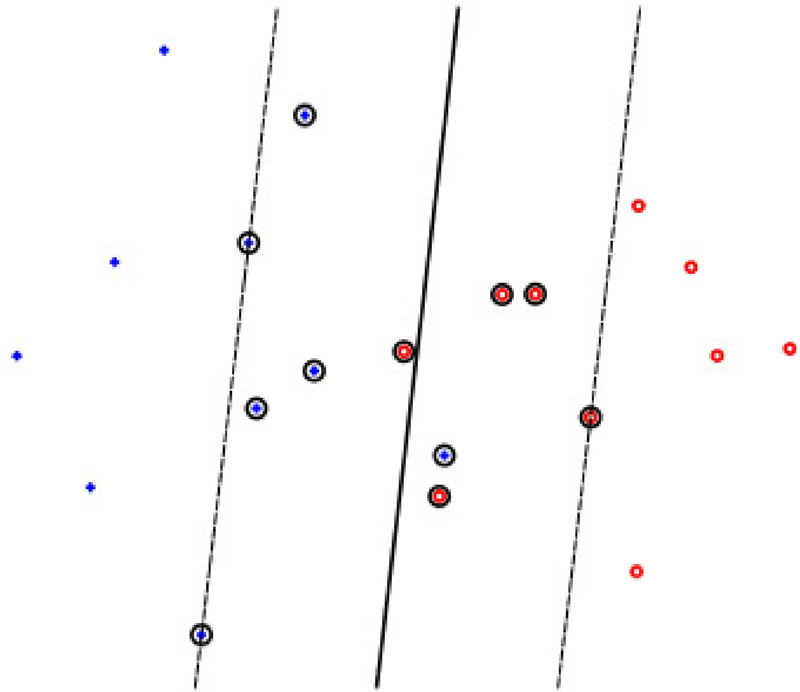
Examples

- C=1



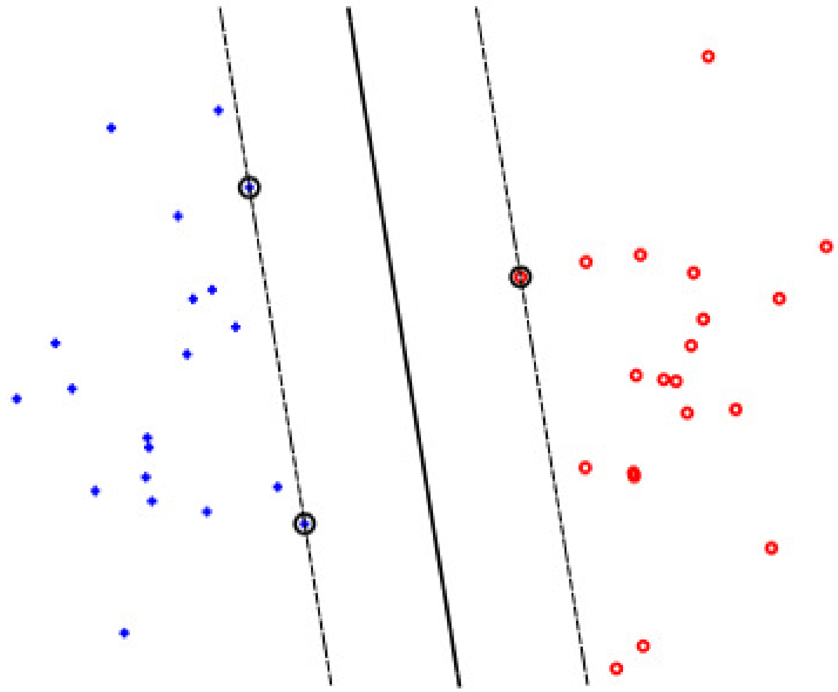
Examples

- C=0.1



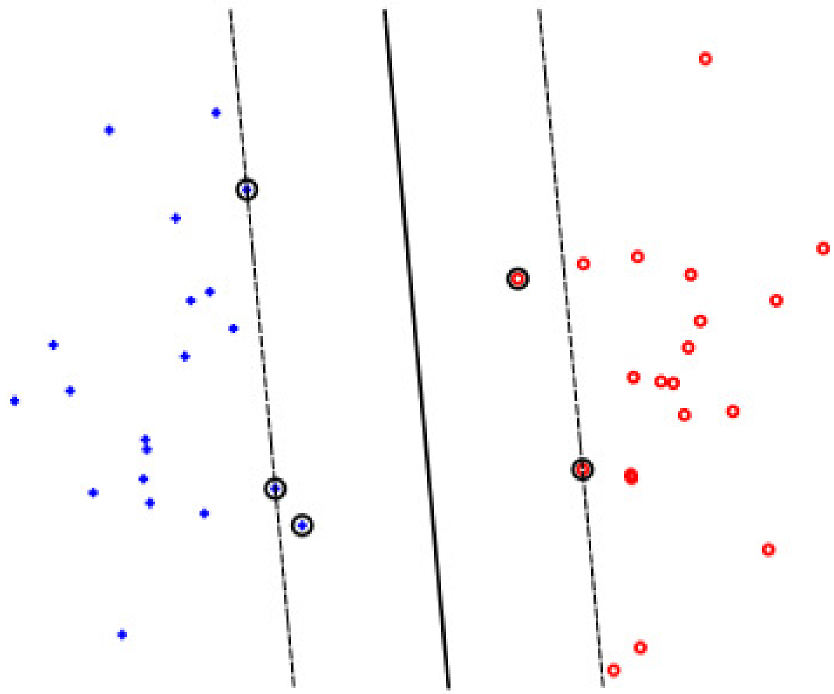
Examples

- C potentially affects the solution even in the separable case
- C = 1



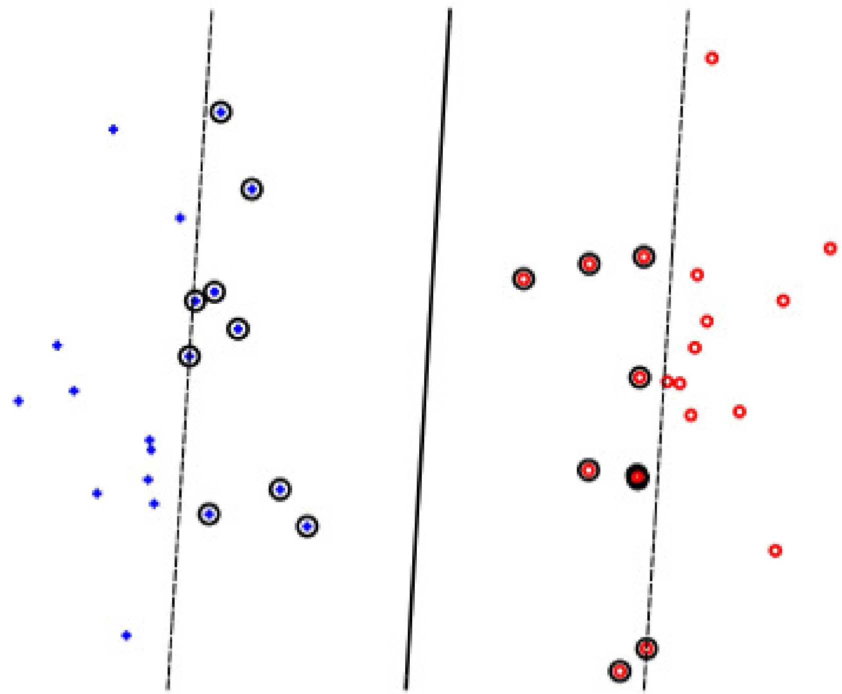
Examples

- C potentially affects the solution even in the separable case
- C = 0.1



Examples

- C potentially affects the solution even in the separable case
- C = 0.01



Original source:

MIT Course 6.867 Machine Learning (Fall 2010) by Prof. Tommi Jaakkola.