Machine Learning

CSCI 5622 Fall 2020

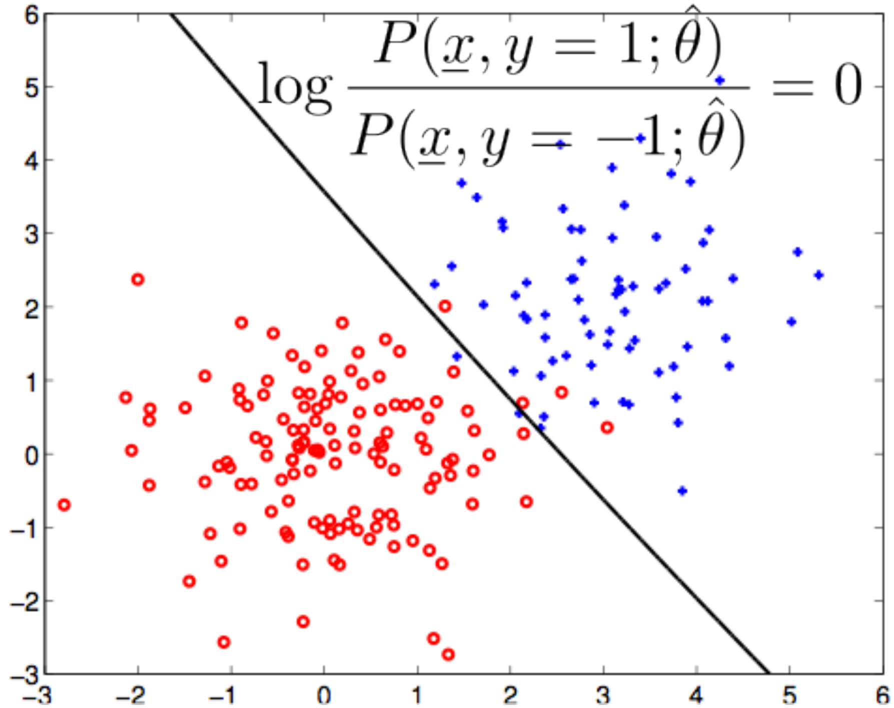
Prof. Claire Monteleoni

Today

- Discriminative learning II
  - Logistic regression
  - Empirical risk minimization
  - Regularization
  - Support vector machines (SVM)

with much credit to S. Dasgupta and T. Jaakkola

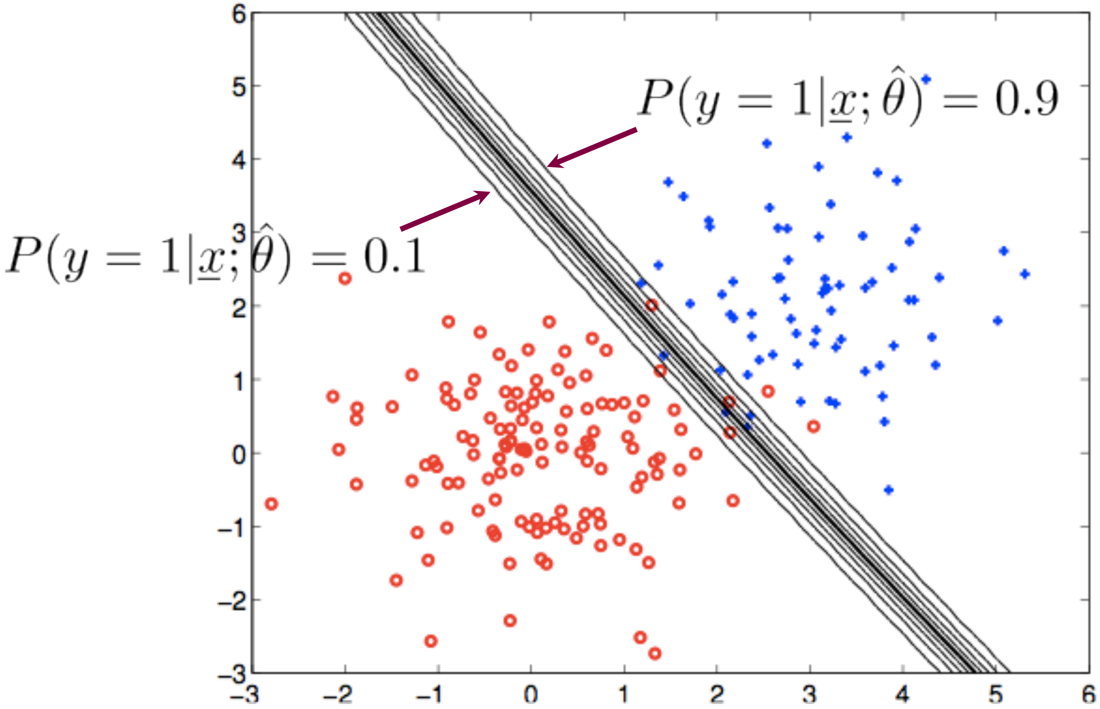
Decision boundary



Probability predictions

- The model also permits us to evaluate probabilities over the possible class labels such as

$$P(y=1|\underline{x};\hat{\theta}) = \frac{P(\underline{x},y=1;\hat{\theta})}{\sum_{y'\in\{-1,1\}} P(\underline{x},y';\hat{\theta})}$$



Logistic regression

Data in R^d, with labels {+1, -1}

What model do we use for P(y|x)?

Recall: for Gaussian classes with common covariance,

$$\log \frac{P(y=+1|x)}{P(y=-1|x)} = w \cdot x + \mathrm{b}$$

Use b = 0 for convenience
(we'll put it in later – or else add an extra feature to each x)

Rearranging, $P(y|x) = \frac{1}{1+e^{-y(w \cdot x)}}$

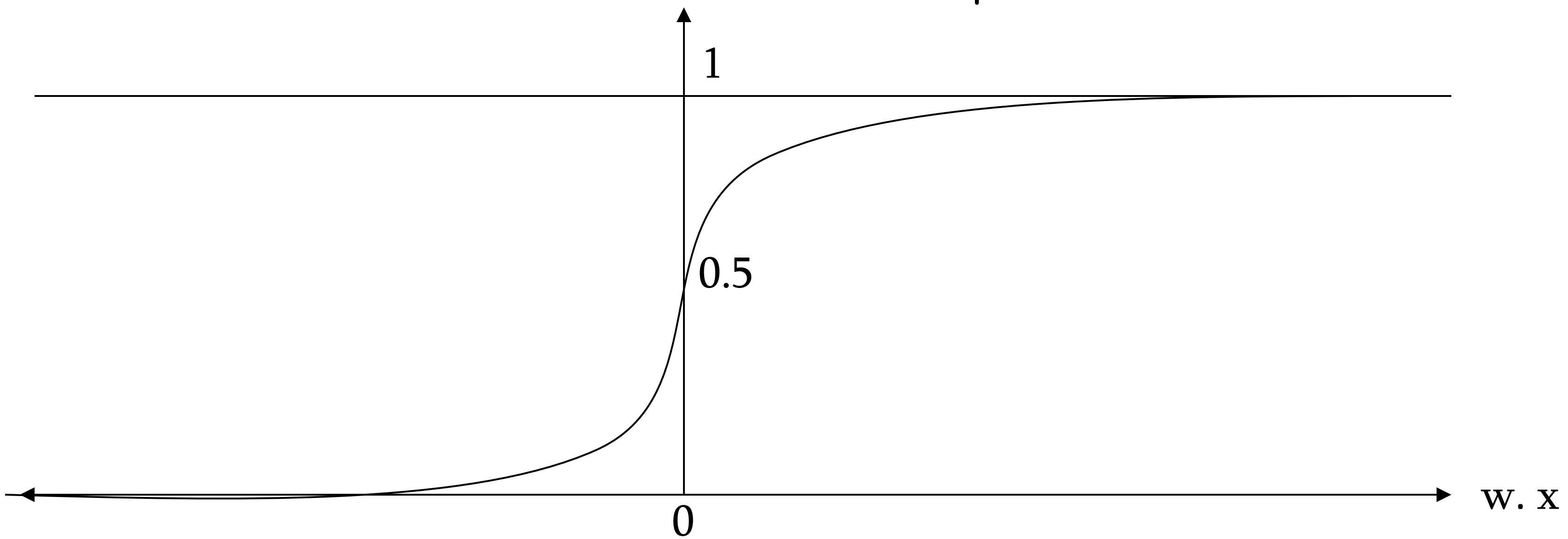
This is the logistic regression model

Logistic function

$$P(y|x) = \frac{1}{1 + e^{-y(w \cdot x)}}$$

Convert linear function (w . x) into probability values via the Logistic function, g:

$$g : \mathbf{R} \rightarrow [0, 1], \qquad g(z) = \frac{1}{1 + e^{-z}}$$



Logistic regression

Notation: let θ = w, θ_0 = b, and g is the logistic function.

The logistic regression (probabilistic) classifier is

$$P(y|\mathbf{x},\theta,\theta_0) = g\left(\, y(\theta^T\mathbf{x} + \theta_0)\,\right)$$

To learn this model from data, we want to choose parameters to maximize this probability. For a labeled, i.i.d. training set of size n, maximize (conditional) log-likelihood:

$$L(\theta,\theta_0) = \prod_{t=1}^{n} P(y_t|\mathbf{x}_t,\theta,\theta_0)$$

Logistic regression

We compute with logs for simplicity: $l(\theta, \theta_0) =$

$$= \log \prod_{t=1}^{n} P(y_t|\mathbf{x}_t, \theta, \theta_0) = \sum_{t=1}^{n} \log P(y_t|\mathbf{x}_t, \theta, \theta_0)$$

And we will actually *minimize* the negative log-likelihood: $-l(\theta, \theta_0)$.

$$-\,l(\theta, \theta_0) \;=\; \sum_{t=1}^{n} \overbrace{-\log P(y_t|\mathbf{x}_t, \theta, \theta_0)}^{\text{log-loss}}$$

Logistic regression

$$\begin{aligned}
-\,l(\theta,\theta_0) &= \sum_{t=1}^{n} \overbrace{-\log P(y_t|\mathbf{x}_t,\theta,\theta_0)}^{\text{log-loss}} \\
&= \sum_{t=1}^{n} -\log g\left(\,y_t(\theta^T\mathbf{x}_t+\theta_0)\,\right) \\
&= \sum_{t=1}^{n} \log\left[1+\exp\left(\,-y_t(\theta^T\mathbf{x}_t+\theta_0)\,\right)\right]
\end{aligned}$$

This is a convex objective function, so min value is unique.

- Can minimize it iteratively, using (stochastic) gradient descent

Logistic regression

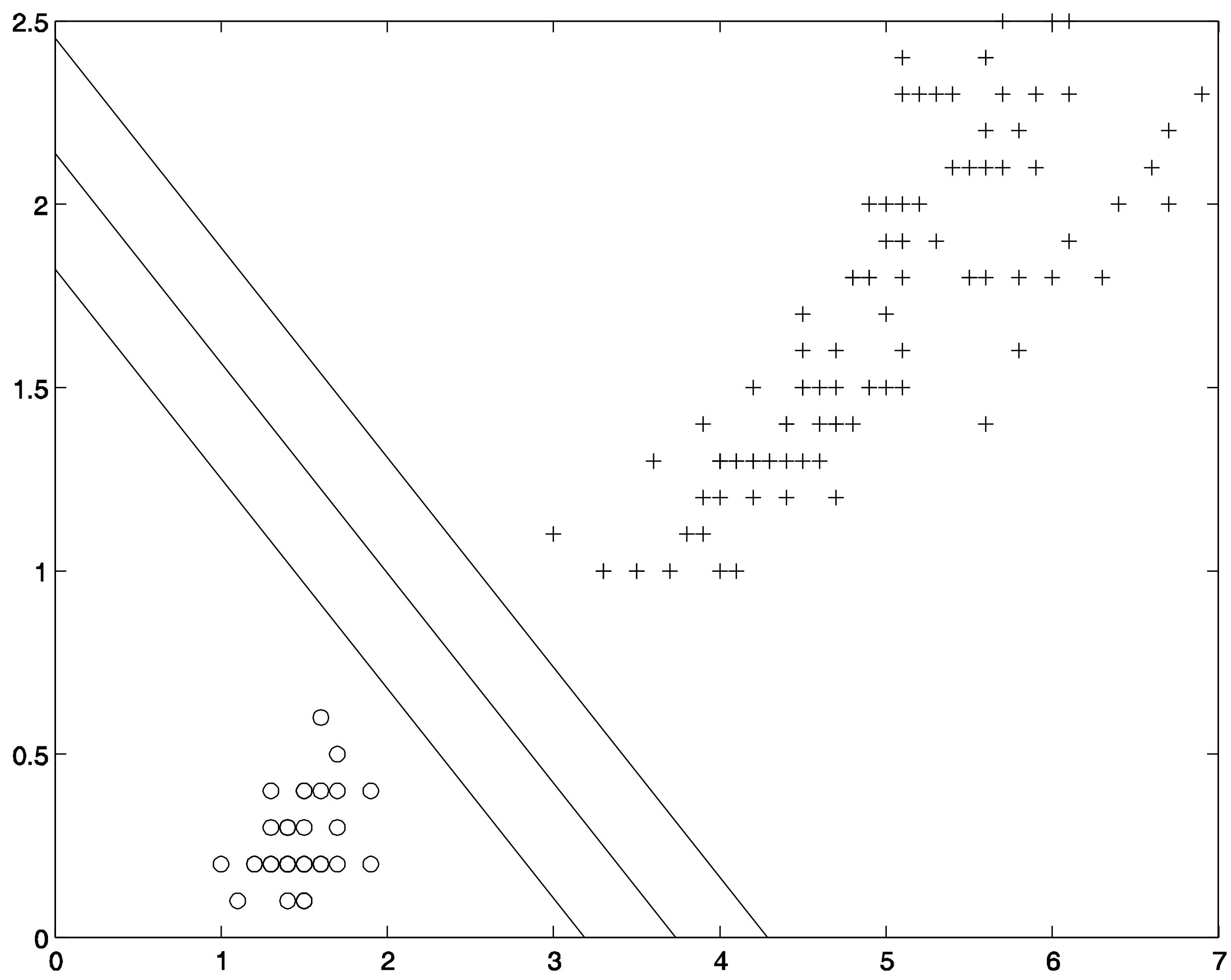
Stochastic gradient descent:

- Choose a random example from the training set (x_t, y_t)
- For each parameter take a step in the opposite direction from the derivative at (x_t, y_t). [partial deriv. w.r.t. θ, θ_0]

$$
\begin{aligned}
\theta_0 &\leftarrow \theta_0 + \eta \cdot y_t[1 - P(y_t|\mathbf{x}_t, \theta, \theta_0)] \\
\theta &\leftarrow \theta + \eta \cdot y_t\mathbf{x}_t[1 - P(y_t|\mathbf{x}_t, \theta, \theta_0)]
\end{aligned}
$$

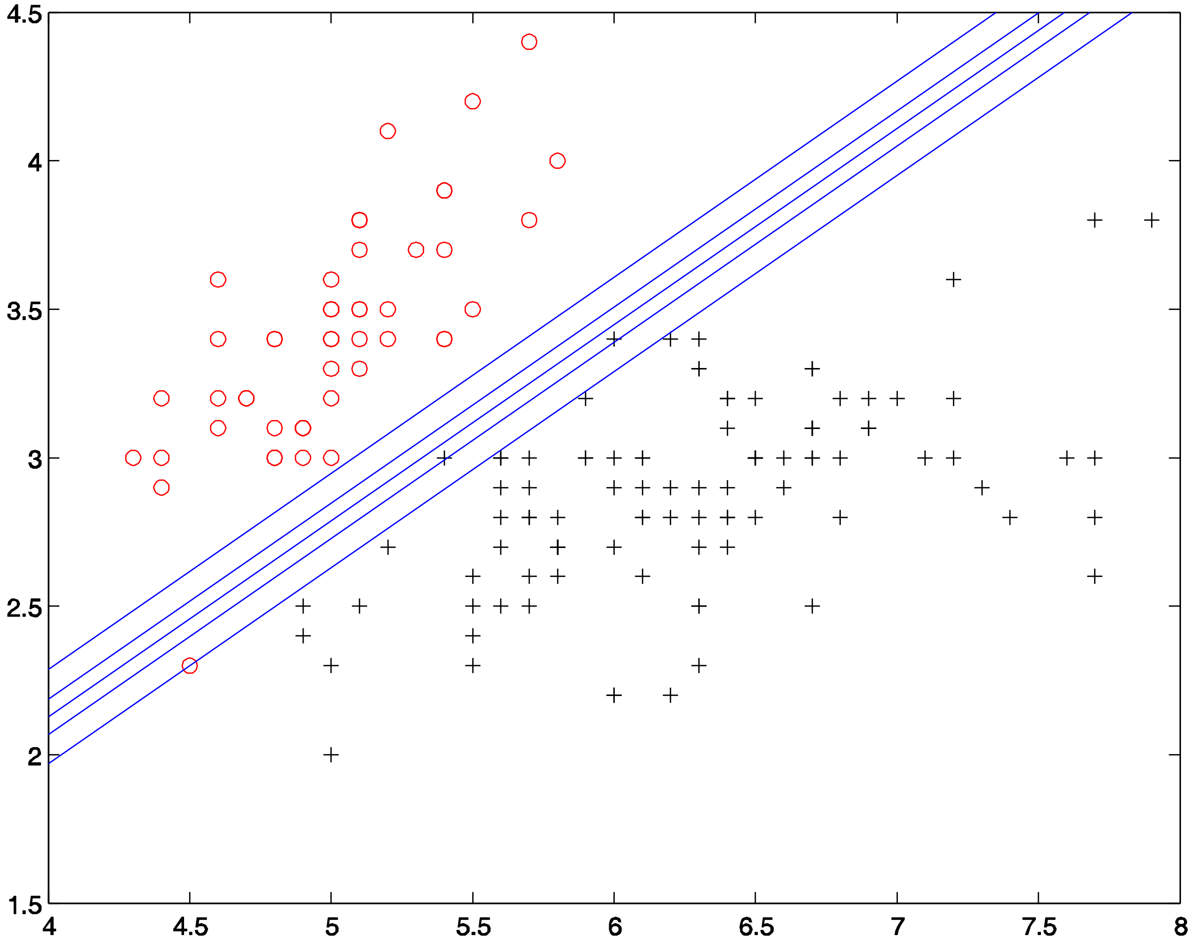
- The learning rate η governs step-size.
- [Note: Similar to Perceptron, but updates take into account probability of making a mistake.]

Logistic regression



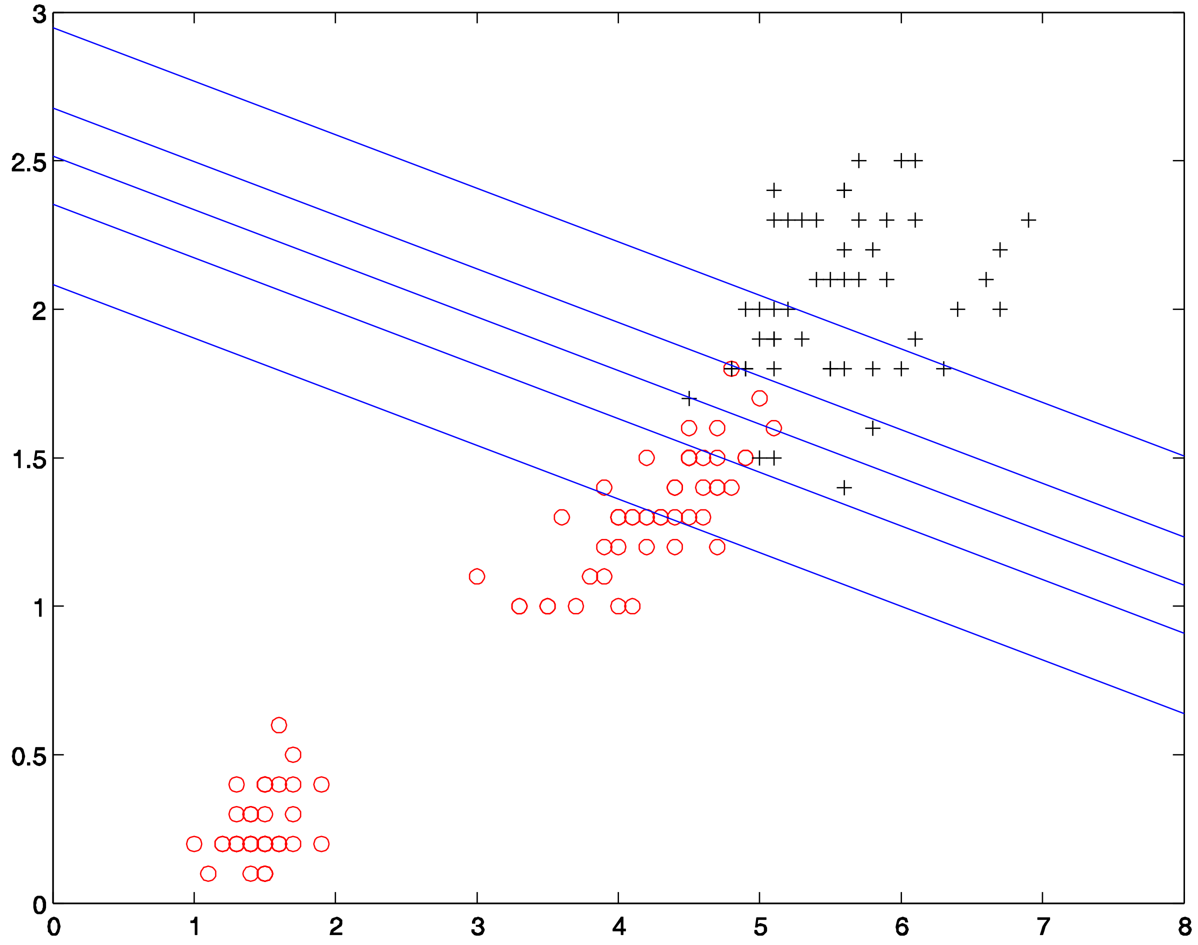
2 iterations of Newton-Raphson

Logistic regression



2 iterations of Newton-Raphson

Logistic regression



81 iterations of Newton-Raphson

Regularization

- Simpler classifiers tend to have better generalization properties. So to reduce overfitting, add a regularization term.
- Also known as a complexity penalty on θ.
- For regularized logistic regression, minimize:

$$\frac{\lambda}{2}\|\theta\|^2 + \sum_{t=1}^{n} \log\left[1 + \exp\left(-y_t(\theta^T \mathbf{x}_t + \theta_0)\right)\right]$$

λ, the regularization constant, manages this trade-off.

Empirical Risk Minimization

Methods that output a classifier by optimizing an objective of the form:

$$\hat{\theta} = \arg\min_{\theta} \lambda \cdot \text{Complexity}(\theta) + \frac{1}{n}\sum_{t=1}^{n} \text{Loss}(\theta, (x_t, y_t))$$

are a class of ML algorithms known as (regularized) Empirical Risk Minimization (ERM).

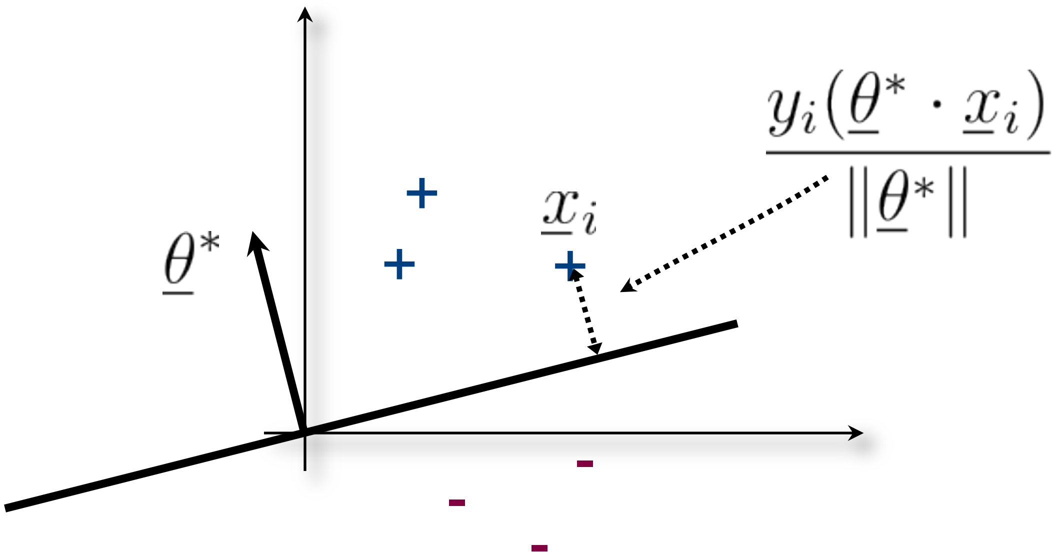
First term is known as Regularizer, second term as Empirical Risk (or Empirical Loss).

Empirical Risk Minimization

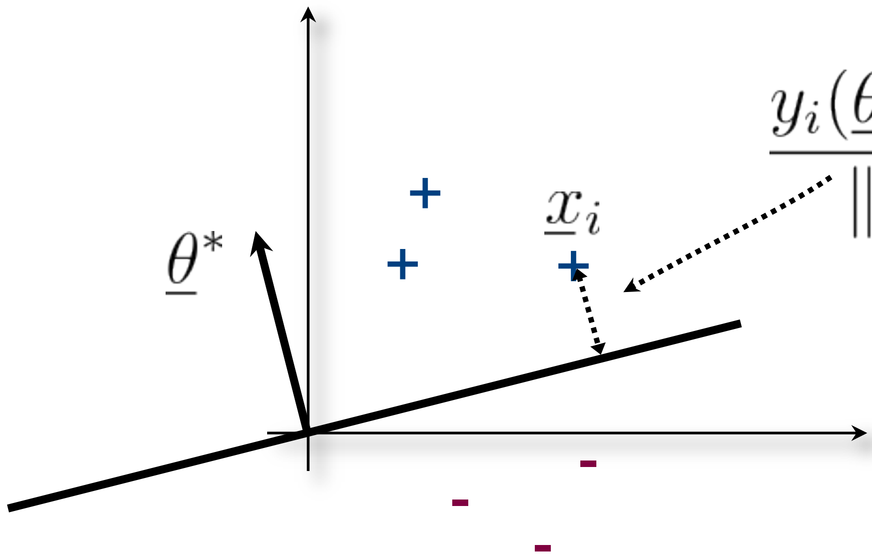
Widely-used (regularized) ERM methods:

- Logistic Regression
- Support Vector Machine (SVM)
  - We will motivate SVM with large margin classification.

Maximum margin classifier



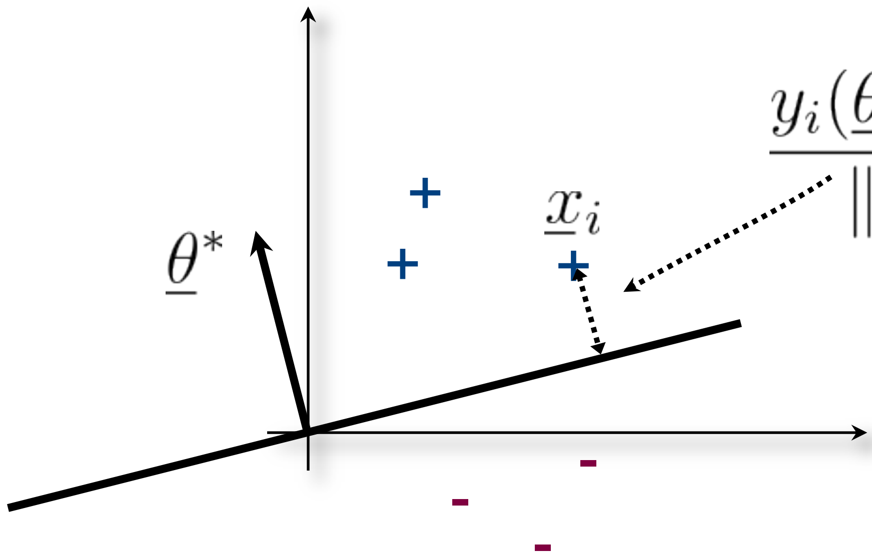
Maximum margin classifier



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

geometric margin

Maximum margin classifier



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

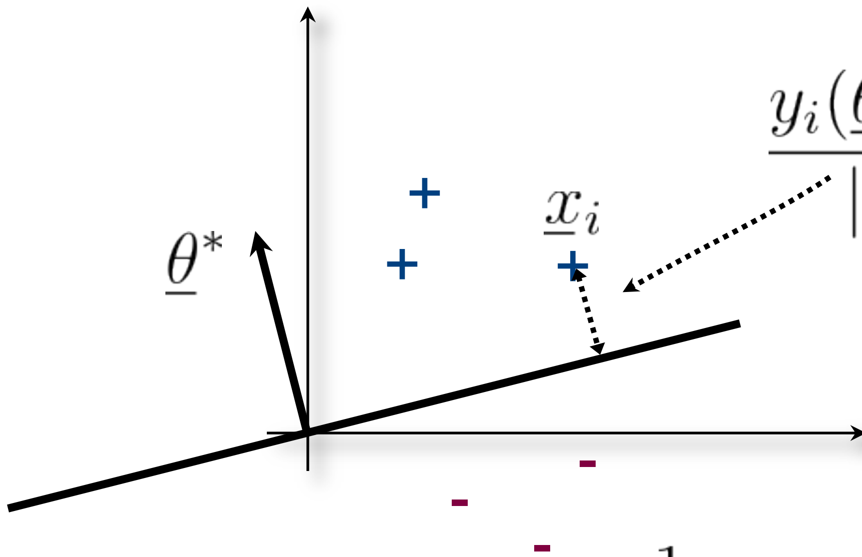
geometric margin

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

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

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

Maximum margin classifier



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

geometric margin

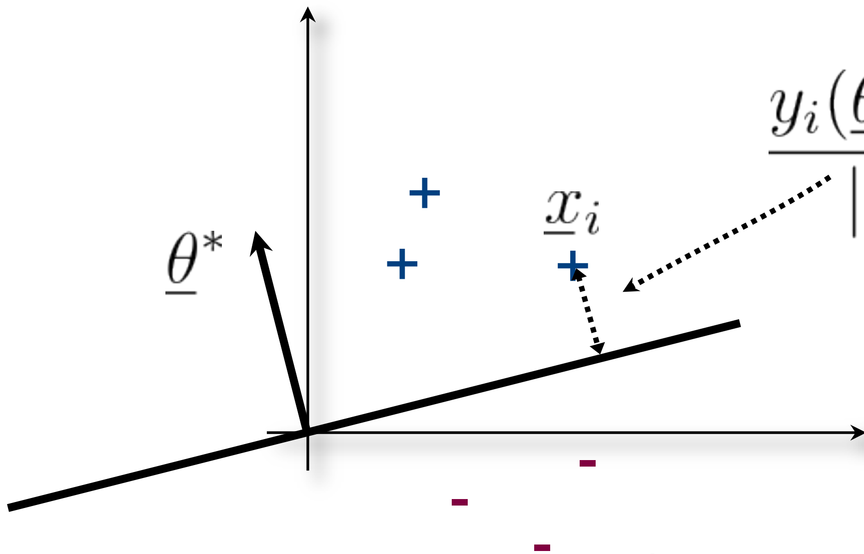
$$\gamma_g = \frac{1}{\|\underline{\theta}^*\|}$$

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

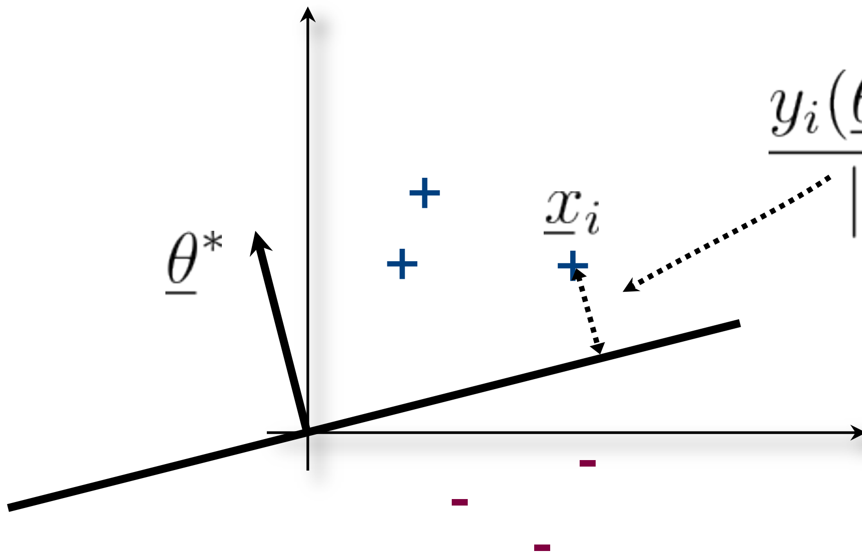
$$\gamma_g = \frac{1}{\|\underline{\theta}^*\|}$$

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



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

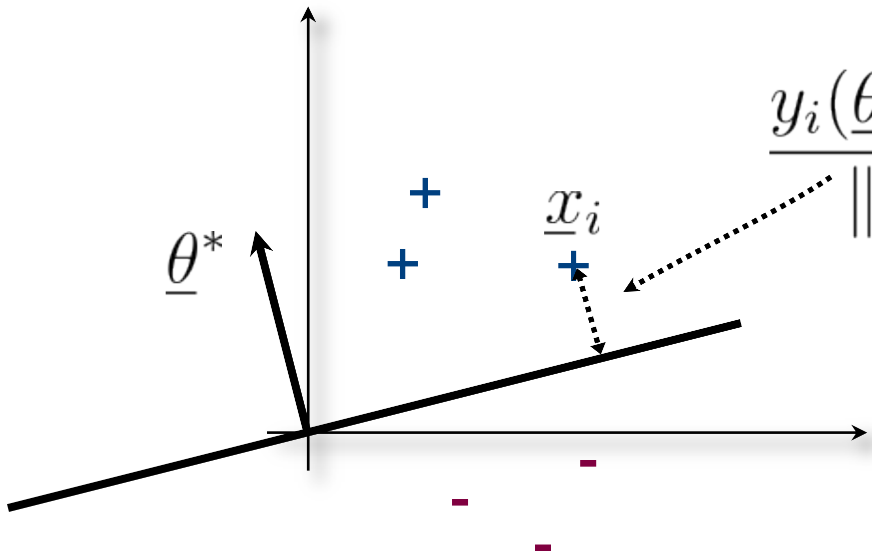
geometric margin

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

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

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

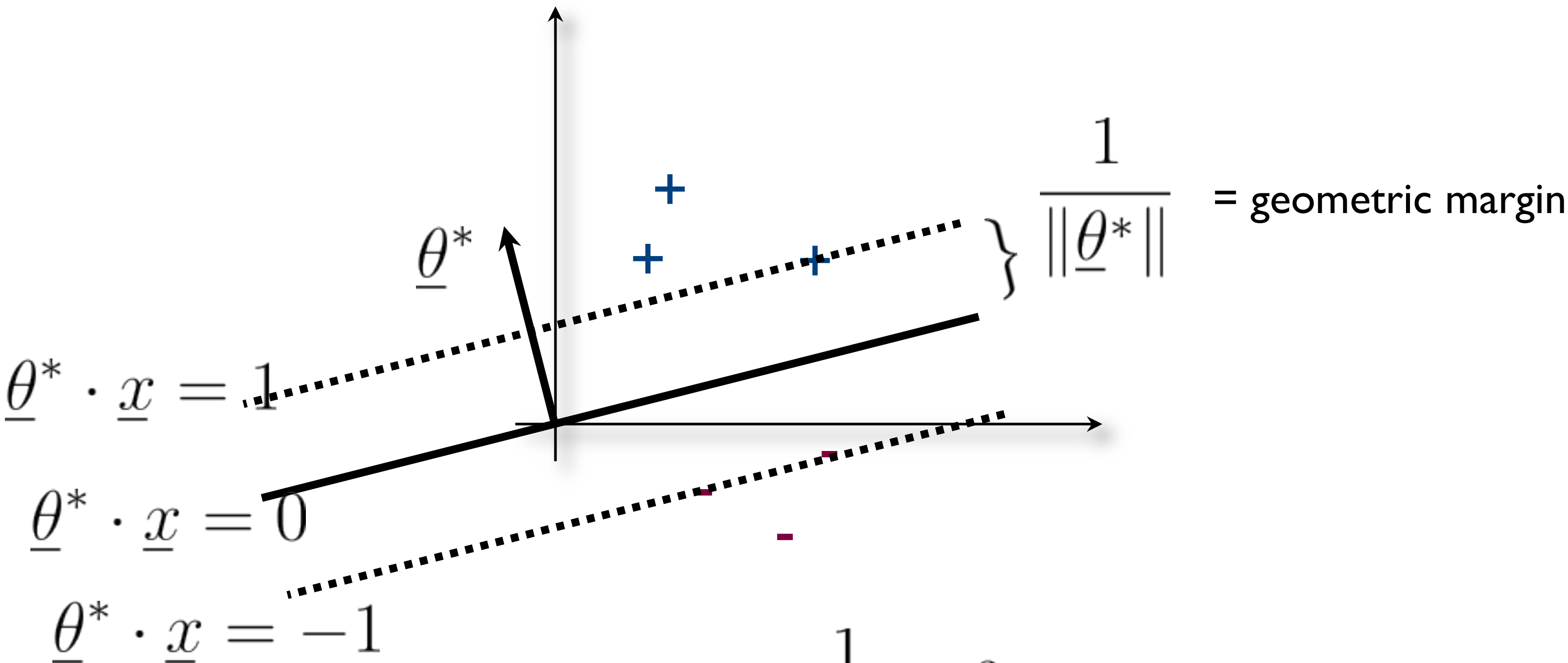
To find $\underline{\theta}^*$:

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

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

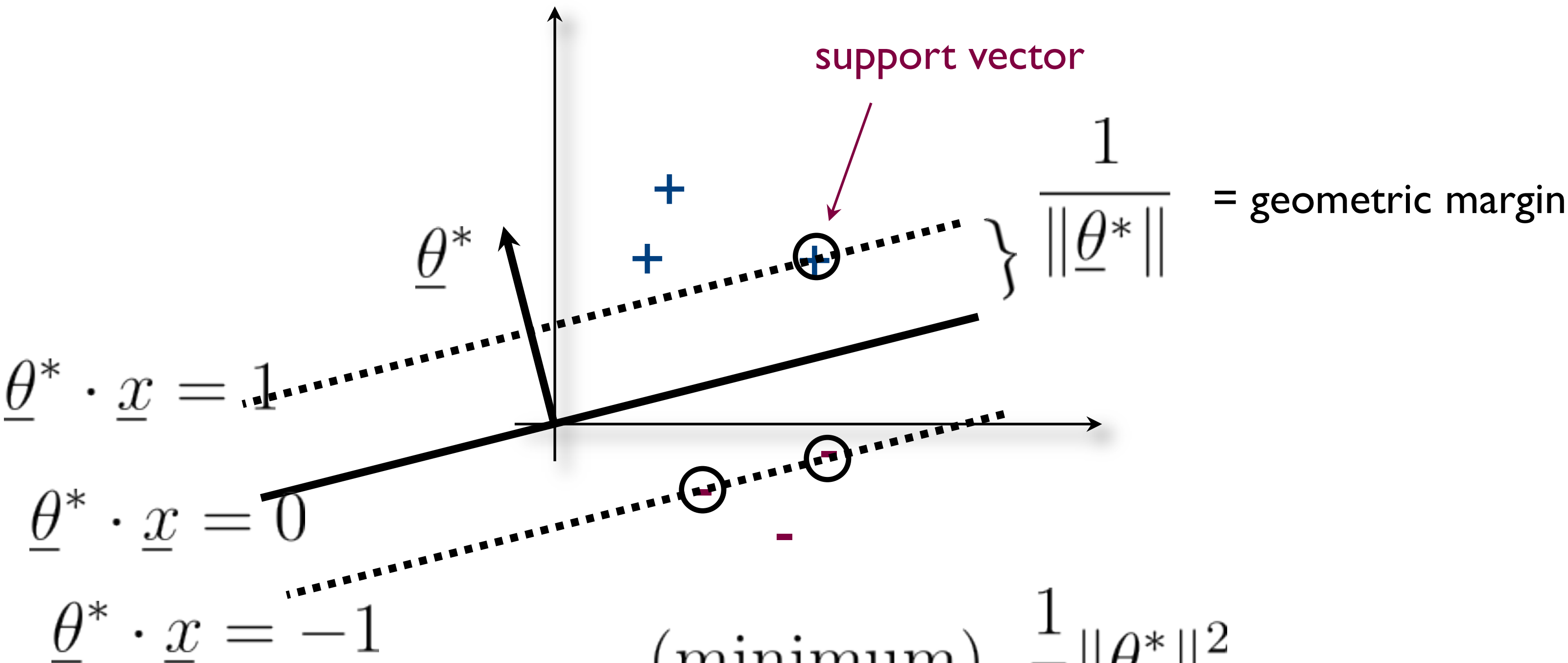
- This is a quadratic programming problem (quadratic objective, linear constraints)
- The solution is unique, typically obtained in the dual

Support vector machine



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

Support vector machine



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

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

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

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

$$\ldots$$

active constraints
= support vectors

The solution is **sparse**

Sparse solution

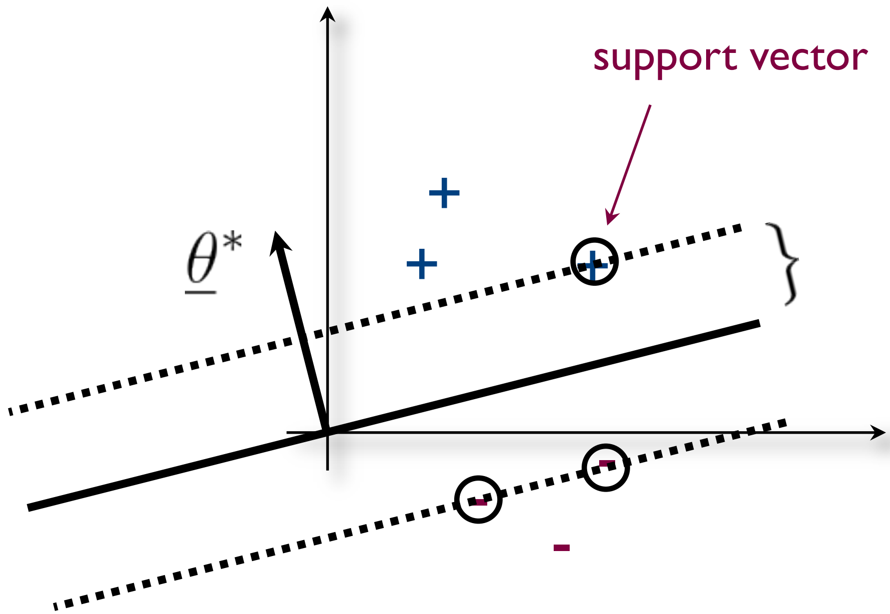
The solution in **sparse** in two ways:

1) The number of support vectors is small

2) The resulting $\underline{\theta}^*$ vector will have small L-2 norm

- usually just a few parameters with values significantly > 0.

Is sparse solution good?



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

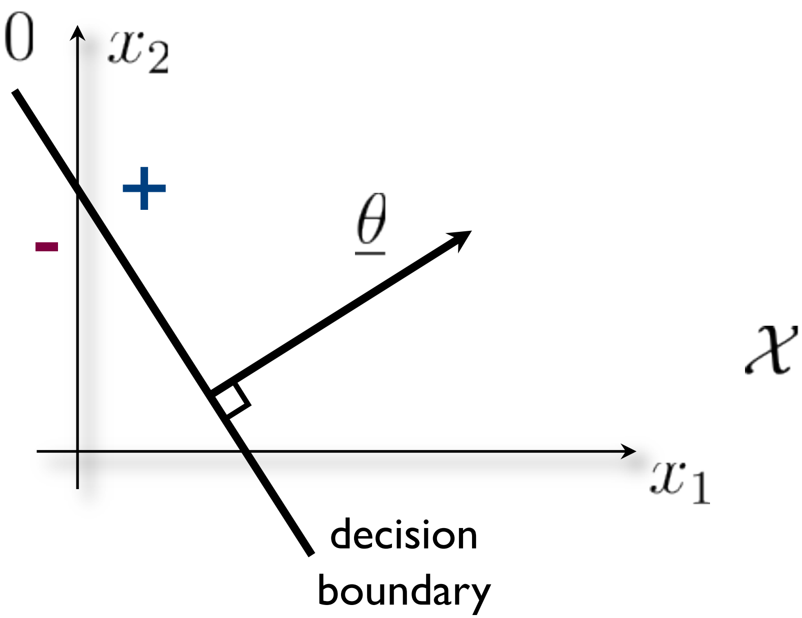
$$\mathrm{LOOCV}(\underline{\theta}^*) \leq \frac{\#\text{ of support vectors}}{n}$$

Linear classifiers (with offset)

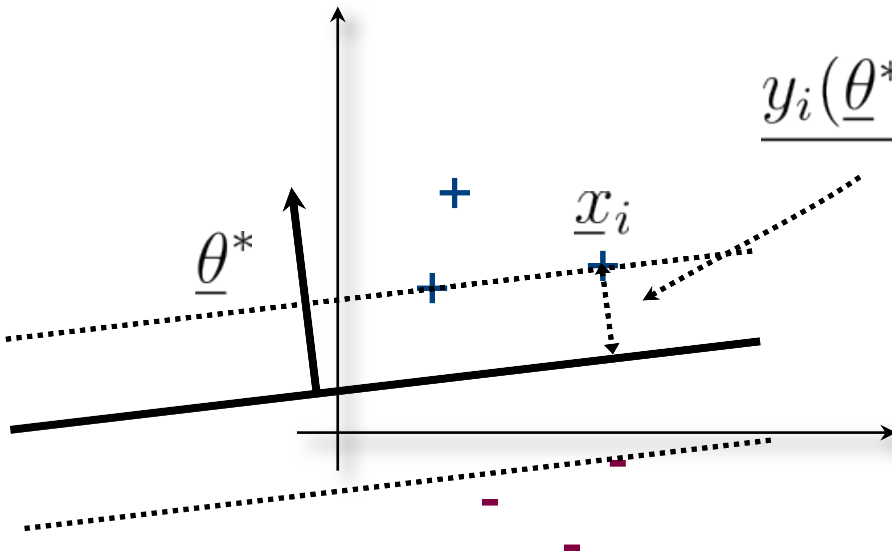
- A linear classifier with parameters $(\underline{\theta}, \theta_0)$

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

$$\underline{\theta}\cdot\underline{x}+\theta_0=0$$



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 $\underline{\theta}^*, \theta_0^*$:
$$\text{minimize } \frac{1}{2}\|\underline{\theta}\|^2 \text{ subject to}$$
$$y_i(\underline{\theta} \cdot \underline{x}_i + \theta_0) \geq 1, \quad i = 1, \ldots, n$$

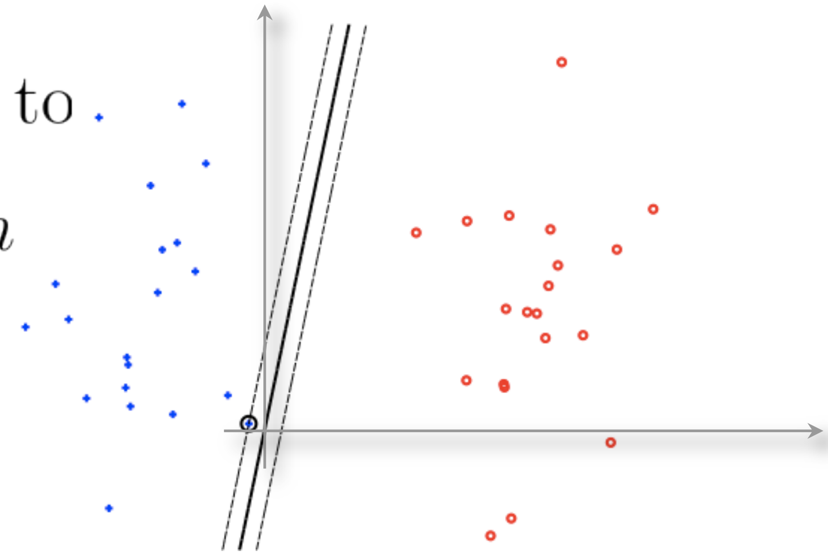
- Still a quadratic programming problem (quadratic objective, linear constraints)

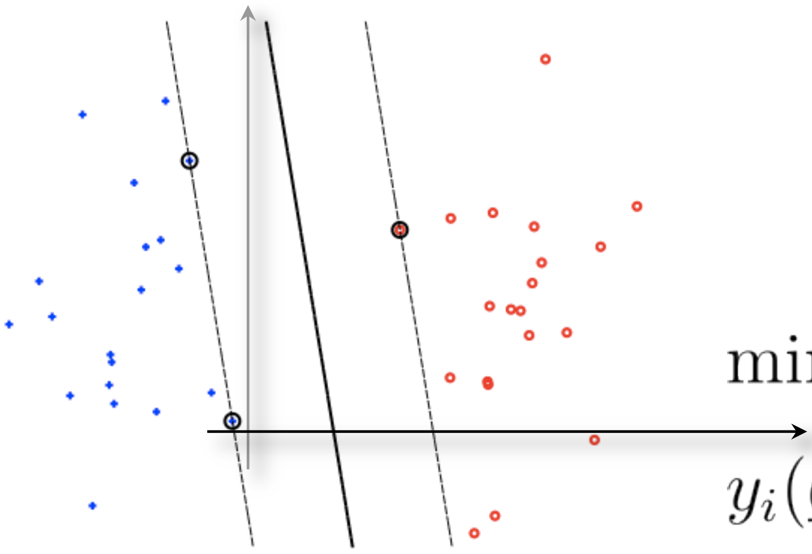
The impact of offset

- 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 (version so far)

- 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, and 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

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

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

slack variables
permit us to violate
some of the margin
constraints

Support vector machine

- Relaxed quadratic optimization problem

penalty for constraint violation

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

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

slack variables permit us to violate some of the margin constraints

$$\text{large } C \Rightarrow \text{few (if any) violations}$$

$$\text{small } C \Rightarrow \text{many violations}$$

Support vector machine

- Relaxed quadratic optimization problem

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

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

slack variables permit us to violate some of the margin constraints

$$\text{large } C \Rightarrow \text{few (if any) violations}$$
$$\text{small } C \Rightarrow \text{many violations}$$

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

Soft-margin SVM

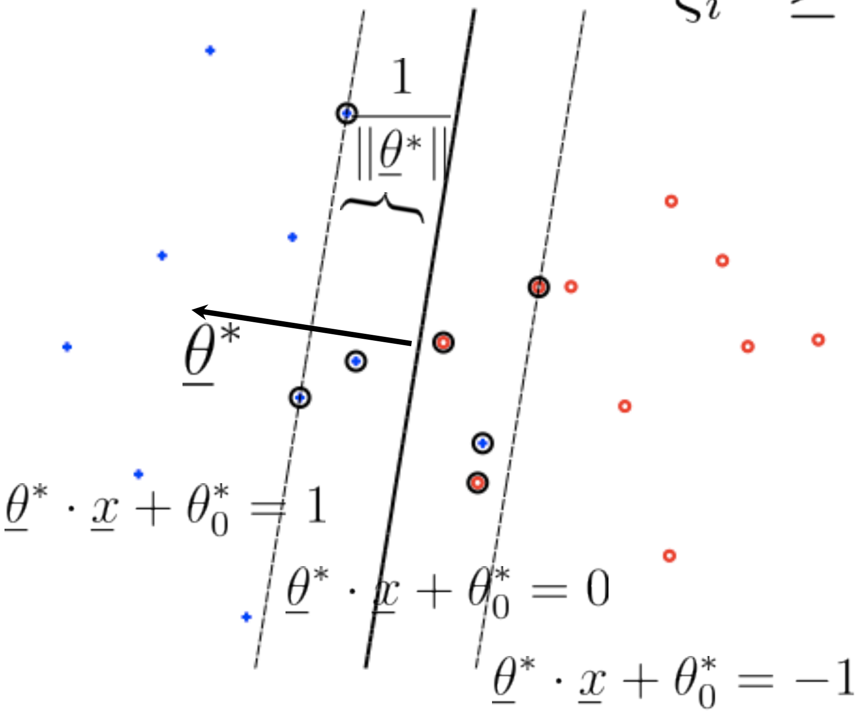
- We relaxed the optimization problem by adding slack variables
- So not all the constraints need to be met
- The solution therefore need not:
  - Classify all training points with a margin
  - Correctly classify all training points
- The margin is still the region within $\dfrac{1}{\|\underline{\theta}^*\|}$ of the decision boundary

Support vector machine

- Relaxed quadratic optimization problem

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

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



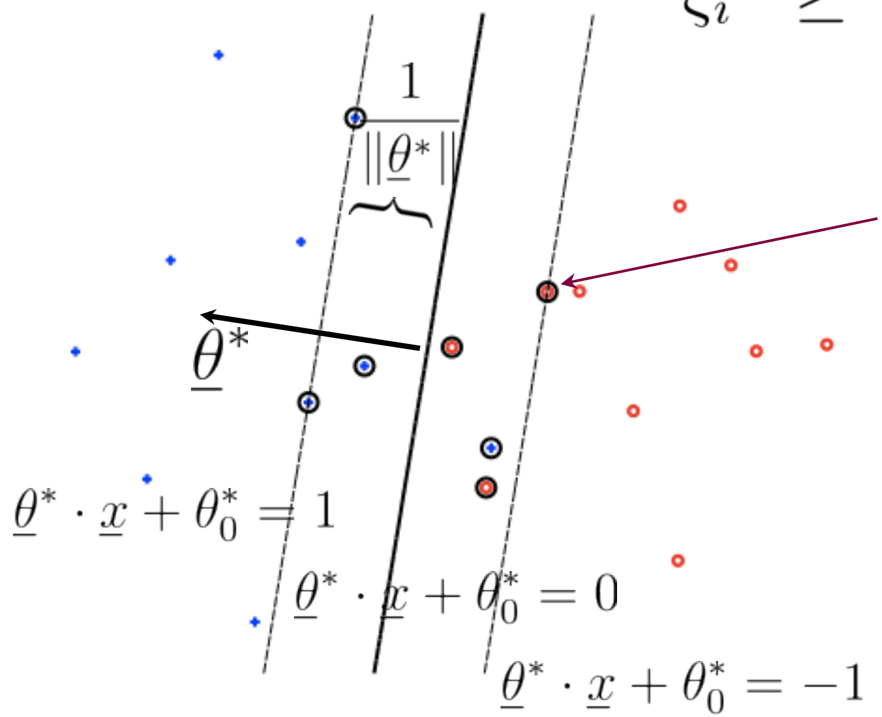
Support vectors and slack

- The solution now has three types of support vectors

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

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

$$\xi_i \quad \geq \quad 0, \quad i=1,\ldots,n$$

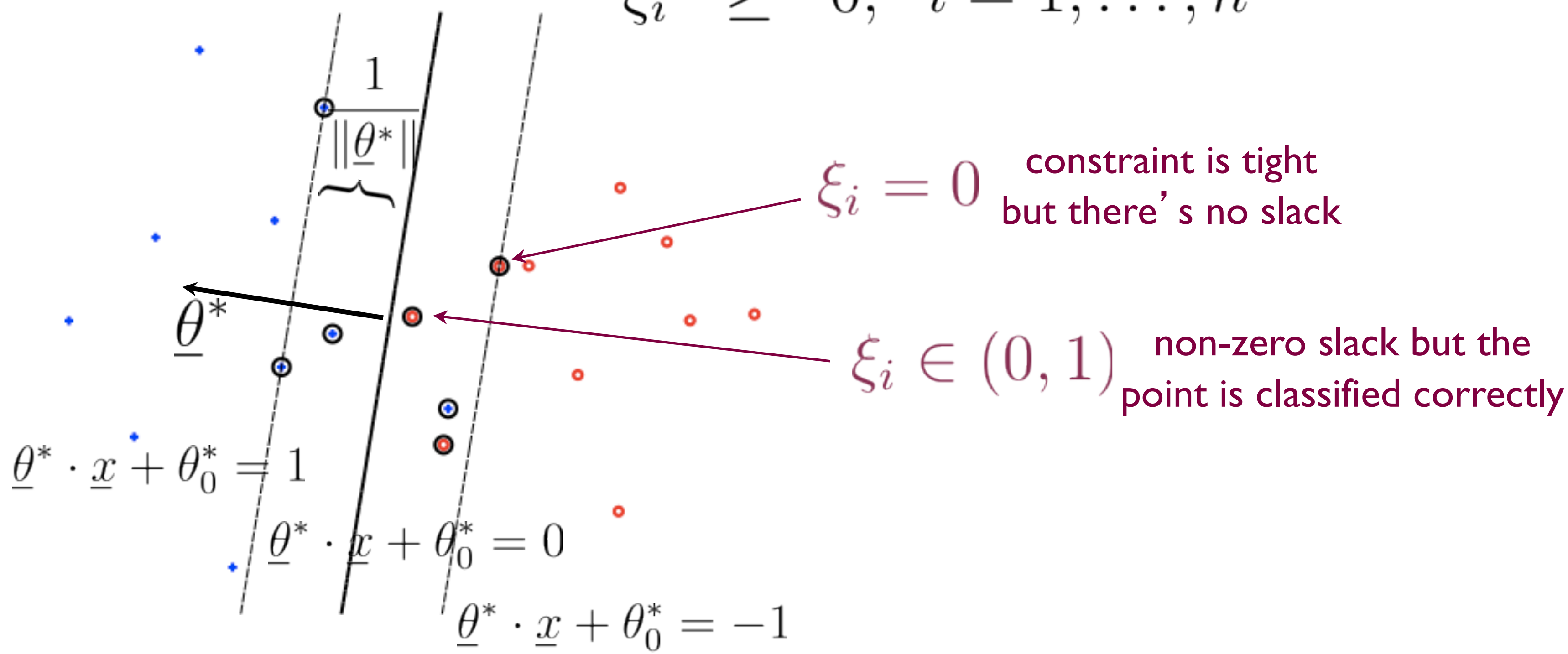


$\xi_i = 0$ constraint is tight and there's no slack

Support vectors and slack

- The solution now has three types of support vectors

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

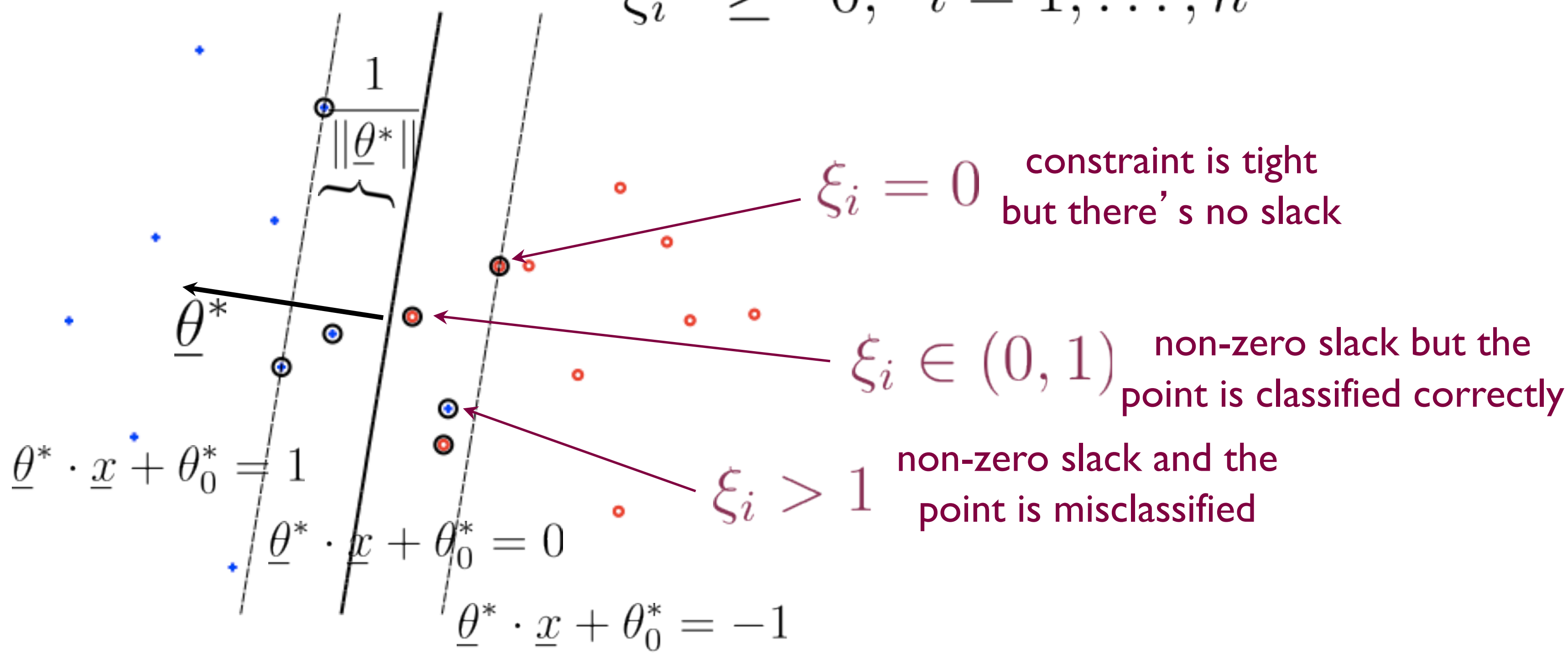


Support vectors and slack

- The solution now has three types of support vectors

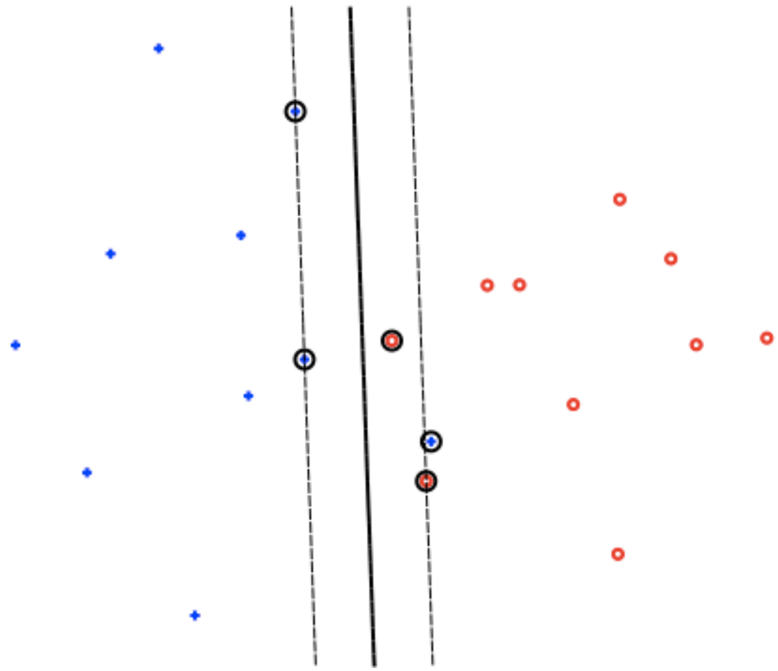
$$\text{minimize} \quad \frac{1}{2}\|\underline{\theta}\|^2 \; + \; C\sum_{i=1}^{n}\xi_i \quad \text{subject to}$$

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



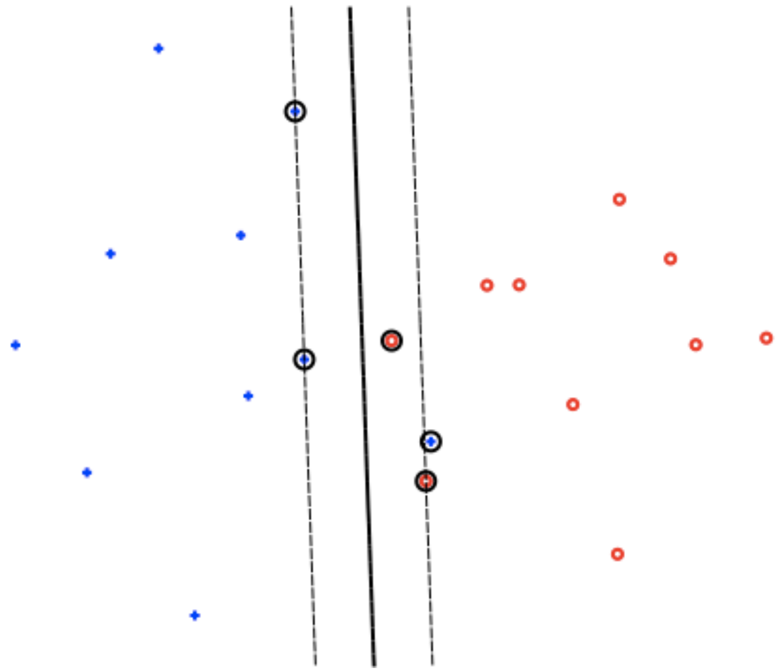
Examples

- C=100



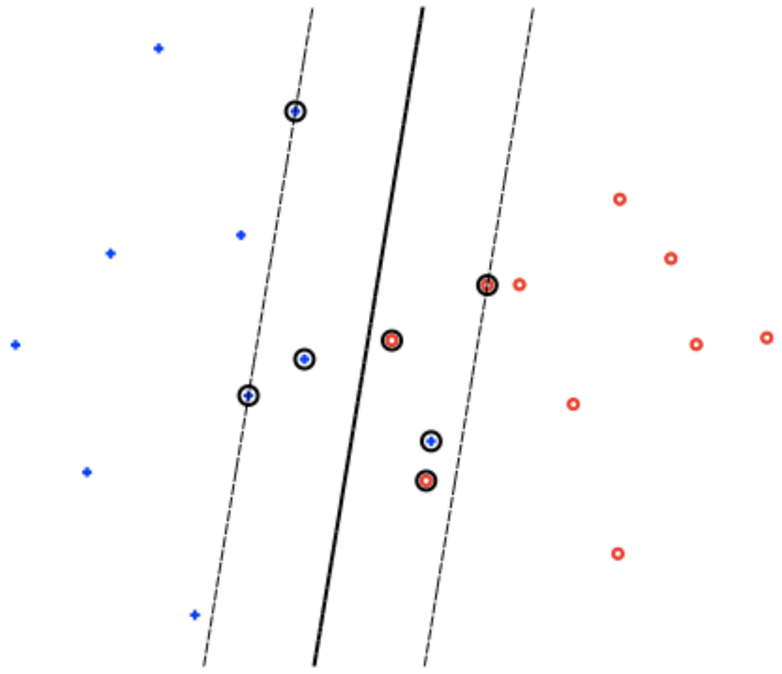
Examples

- C=10



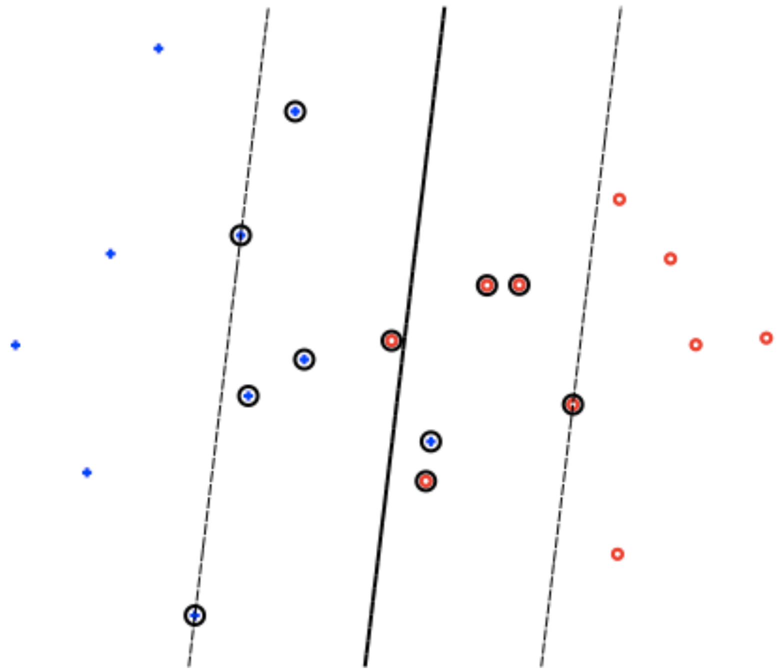
Examples

- C=1



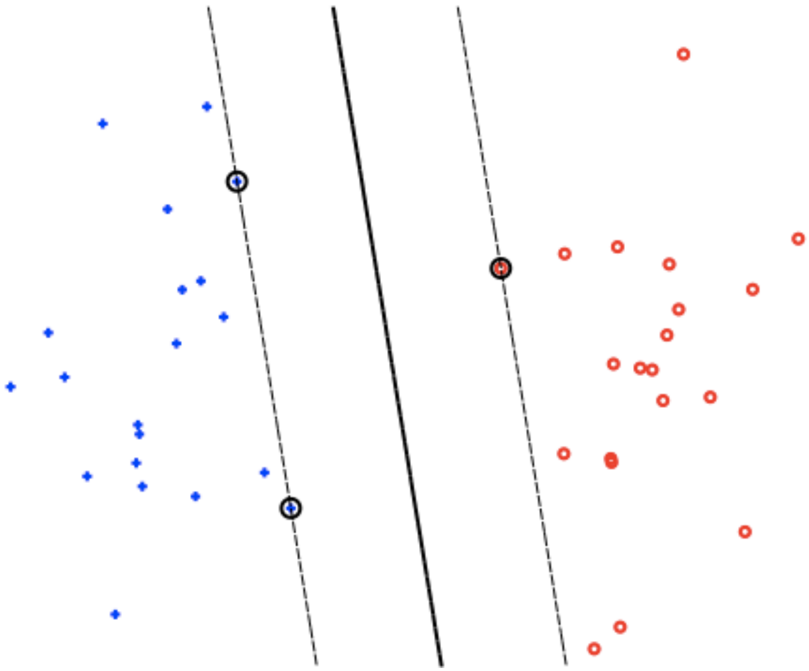
Examples

- C=0.1



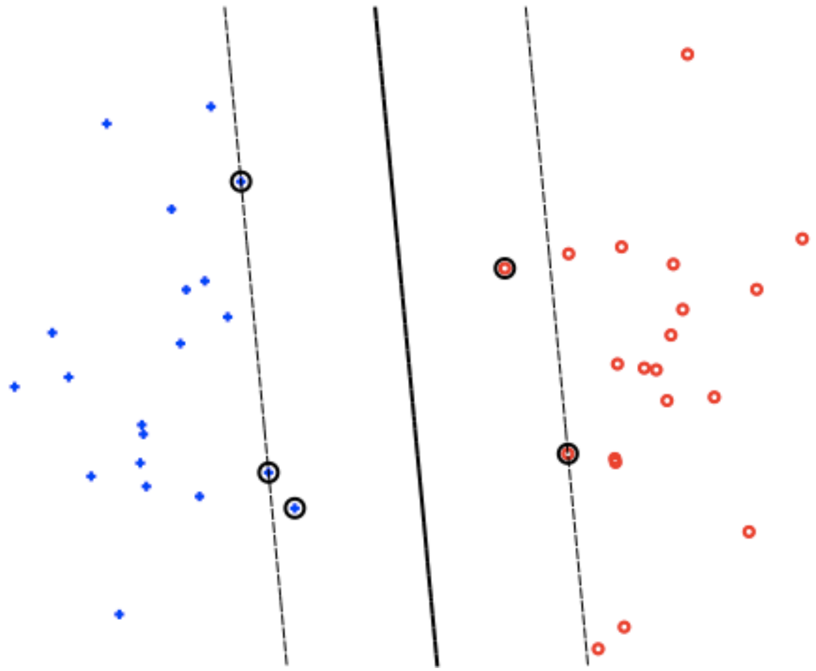
Examples

- 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