Learning From Data
Lecture 5: Support Vector Machines

Yang Li   yangli@sz.tsinghua.edu.cn

October 15, 2021

Previously on Learning from Data

Algorithms we learned so far are mostly **probabilistic linear models**:

Type	Examples
Discrimative probablistic model	linear regression, logistic regression, softmax
Generative probablistic model	GDA, naive Bayes

- Choice of model affects model performance; *may easily lead to model mismatch*
- Data are often sampled non-uniformly, forming a sparse distribution in high dimensional input space. *leading to ill-posed problems*

Possible solutions: regularization (more in later lectures), sparse kernel methods (today's lecture)

Today's Lecture

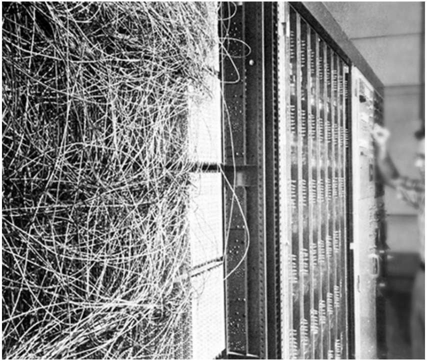
Supervised Learning (Part IV)

- Review: Perceptron Algorithm
- Support Vector Machines (SVM) ← *another discriminative algorithm for learning linear classifiers*
- Kernel SVM ← *non-linear extension of SVM*

Perceptron Learning Algorithm

The perceptron learning algorithm

- Invented in 1956 by Rosenblatt (Cornell University)
- One of the earliest learning algorithm, the first artificial neural network

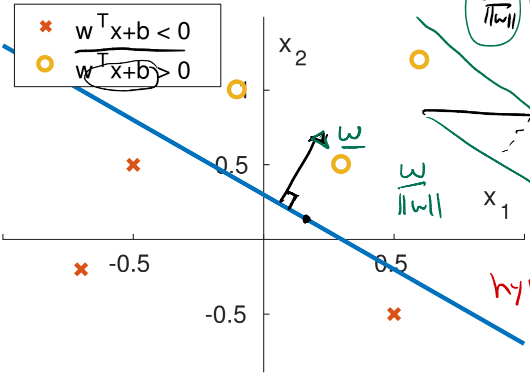


Hardware implementation: Mark I Perceptron

The perceptron learning algorithm

Given x, predict $y \in \{0, 1\}$

$$h_{w,b}(x) = \begin{cases} 1 & \text{if } w^T x + b \geq 0 \\ 0 & \text{otherwise} \end{cases}$$



The perceptron learning algorithm

Perceptron hypothesis function:

$$h_\theta(x) = \begin{cases} 1 & \text{if } \theta^T x \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

Parameter update rule:

$$\theta_j = \theta_j + \alpha \left(y^{(i)} - h_\theta(x^{(i)}) \right) x_j^{(i)} \text{ for all } j = 0, \ldots, n$$

- When prediction is correct: $\theta_j = \theta_j$
- When prediction is incorrect:
  - predicted "1": $\theta_j = \theta_j - \alpha x_j$
  - predicted "0": $\theta_j = \theta_j + \alpha x_j$

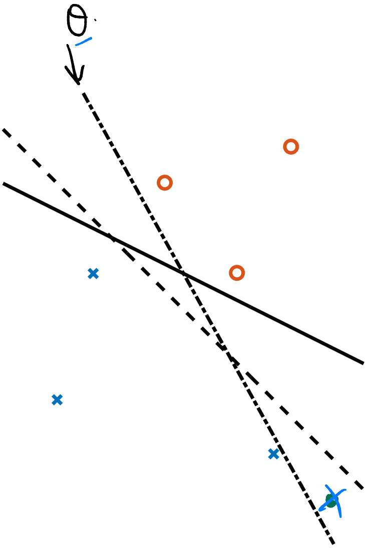
$h_\theta(x) = 1$ and $y^i = 1$.

$h_\theta(x) = 0$ and $y^i = 0$.

$y^i = 0$ $h_\theta(x) = 1$

$h_\theta(x) = 0$.

Issues with linear hyperplane perceptron:



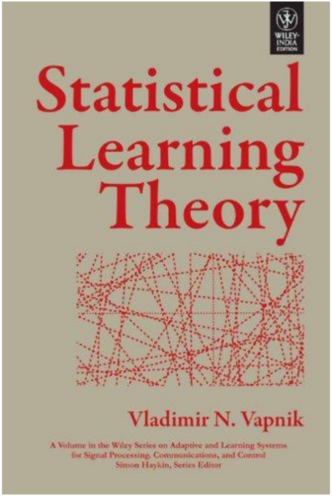
- Infinitely many solutions if data are separable
- Can not express "confidence" of the prediction

Support Vector Machines

- Optimal margin classifier
- Lagrange Duality
- Soft margin SVM

Support Vector Machines in History

- Theoretical algorithm: developed from Statistical Learning Theory (Vapnik & Chervonenkis) since 60s
- Modern SVM was introduced in COLT 92 by Boser, Guyon & Vapnik



Support Vector Machines in History

- 1995 paper by Corte & Vapnik titled "Support-Vector Networks"
- Gained popularity in 90s for giving accuracy comparable to neural networks with elaborated features in a handwriting task

Machine Learning, 20, 273–297 (1995)

Support-Vector Networks

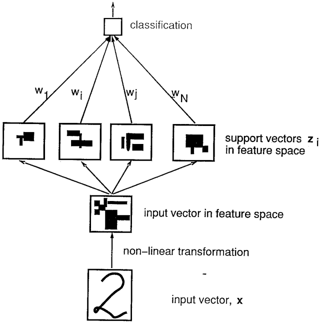
CORINNA CORTES corinna@neural.att.com
VLADIMIR VAPNIK vlad@neural.att.com
AT&T Bell Labs., Holmdel, NJ 07733, USA

Editor: Lorenza Saitta

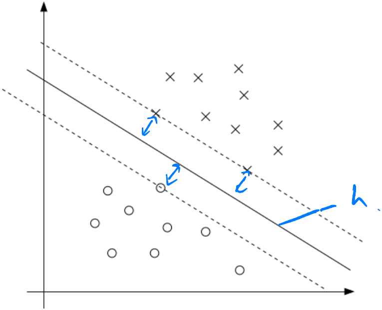
Abstract. The *support-vector network* is a new learning machine for two-group classification problems. The machine conceptually implements the following idea: input vectors are non-linearly mapped to a very high-dimension feature space. In this feature space a linear decision surface is constructed. Special properties of the decision surface ensures high generalization ability of the learning machine. The idea behind the support-vector network was previously implemented for the restricted case where the training data can be separated without errors. We here extend this result to non-separable training data.

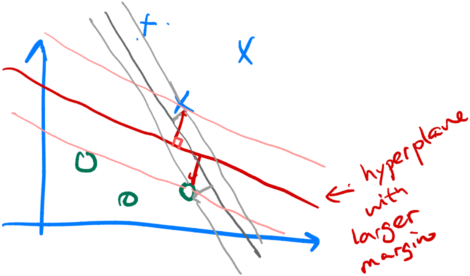
High generalization ability of support-vector networks utilizing polynomial input transformations is demonstrated. We also compare the performance of the support-vector network to various classical learning algorithms that all took part in a benchmark study of Optical Character Recognition.

Keywords: pattern recognition, efficient learning algorithms, neural networks, radial basis function classifiers, polynomial classifiers.



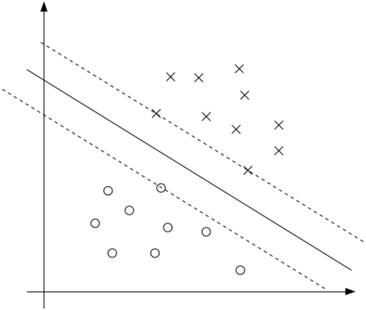
Support Vector Machine: Overview





Margin: smallest distance between the decision boundary to any samples *(Margin also represents classification confidence)*

Support Vector Machine: Overview



Margin: smallest distance between the decision boundary to any samples *(Margin also represents classification confidence)*

Linear SVM

Choose a linear classifier that maximizes the margin.

To be discussed:

- How to measure the margin? (functionally vs geometrically)
- How to find the decision boundary with optimal margin?

  + a detour on Lagrange Duality

Functional margins

perceptron: $y \in \{0, 1\}$

Class labels: $y \in \{-1, 1\}$

$$h_{w,b}(x) = \begin{cases} 1 & \text{if } w^T x + b \geq 0 \\ -1 & \text{otherwise} \end{cases}$$

Functional margins

Class labels: $y \in \{-1, 1\}$

$$h_{w,b}(x) = \begin{cases} 1 & \text{if } w^T x + b \geq 0 \\ -1 & \text{otherwise} \end{cases}$$

$$\hat{\gamma}^{(i)} = y^{(i)}(w^T x^{(i)} + b)$$

y^i	$w^T x^i + b \geq 0$	$w^T x^i + b < 0$
1	++	--
-1	--	++

Functional Margin

Given training sample $(x^{(i)}, y^{(i)})$

$$\hat{\gamma}^{(i)} = y^{(i)}\left(w^T x^{(i)} + b\right)$$

$sign(\hat{\gamma}^{(i)})$: whether the hypothesis is correct

Functional margins

Class labels: $y \in \{-1, 1\}$

$$h_{w,b}(x) = \begin{cases} 1 & \text{if } w^T x + b \geq 0 \\ -1 & \text{otherwise} \end{cases}$$

Functional Margin

Given training sample $(x^{(i)}, y^{(i)})$

$$\hat{\gamma}^{(i)} = y^{(i)} \left(w^T x^{(i)} + b \right)$$

$sign(\hat{\gamma}^{(i)})$: whether the hypothesis is correct

- $\hat{\gamma}^{(i)} >> 0$: prediction is correct with high confidence

Functional margins

Class labels: $y \in \{-1, 1\}$

$$h_{w,b}(x) = \begin{cases} 1 & \text{if } w^T x + b \geq 0 \\ -1 & \text{otherwise} \end{cases}$$

Functional Margin

Given training sample $(x^{(i)}, y^{(i)})$

$$\hat{\gamma}^{(i)} = y^{(i)} \left(w^T x^{(i)} + b \right)$$

$sign(\hat{\gamma}^{(i)})$: whether the hypothesis is correct

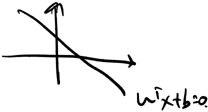
- $\hat{\gamma}^{(i)} >> 0$: prediction is correct with high confidence
- $\hat{\gamma}^{(i)} << 0$: prediction is incorrect with high confidence

Function Margins

Functional margin of (w, b) with respect to training data S:

$$\hat{\gamma} = \min_{i=1,..,m} \hat{\gamma}^{(i)} = \min_{i=1,..,m} y^{(i)} \left(w^T x^{(i)} + b \right)$$

Function Margins



Functional margin of (w, b) with respect to training data S:

$$\hat{\gamma} = \min_{i=1,..,m} \hat{\gamma}^{(i)} = \min_{i=1,..,m} y^{(i)} \left(w^T x^{(i)} + b \right)$$

$w^T x + b = 0 \Leftrightarrow 2w^T x + 2b = 0$

parameter scaling

Issue: $\hat{\gamma}$ depends on $||w||$ and b

e.g. Let $w' = 2w, b' = 2b$. The decision boundary parameterized by (w', b') and (w, b) are the same. However,

$$\hat{\gamma}'^{(i)} = y^{(i)} \left(2w^T x^{(i)} + 2b \right) = 2\underbrace{y^{(i)} (w^T x^{(i)} + b)}_{\hat{\gamma}} = 2\hat{\gamma}^{(i)}$$

Can we express the margin so that it is invariant to $||w||$ and b?

$w' = cw, \; b' = cb.$

$\hat{\gamma}' = c\hat{\gamma}$

Geometric Margins

The **geometric margin** $\gamma^{(i)}$ of a training example $(x^{(i)}, y^{(i)})$ is the distance from the hyperplane: Given (w, b) and $x^{(i)}$.

$Q = x^{(i)} + z^{(i)} = x^{(i)} - \gamma^{(i)} \frac{w}{||w||}$

$$\gamma^{(i)} = y^{(i)} \left(\frac{w}{||w||}^T x^{(i)} + \frac{b}{||w||} \right)$$

$y_i = 1$

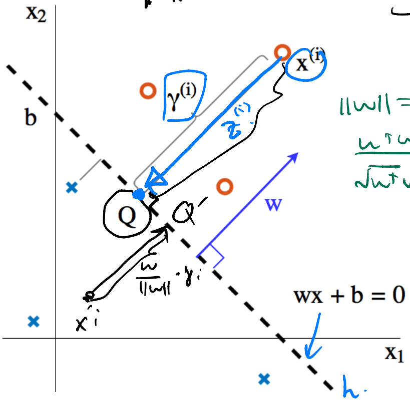
$Q = x^i - \gamma^i \frac{w}{||w||}$

Q is on h, $w^T Q + b = 0$

$w^T (x^i - \gamma^i \frac{w}{||w||}) + b = 0.$

$\gamma^i = \frac{w^T x}{||w||} + \frac{b}{||w||}$

$||w|| = \sqrt{w^T w}$, $\frac{w^T w}{\sqrt{w^T w}} = \sqrt{w^T w} = ||w||$



- w is normal to hyperplane $w^T x + b = 0$
- We want $\gamma^{(i)} > 0$ when prediction is correct

when $y^i = -1$, $Q' = x^i + \gamma^i \frac{w}{||w||}$

$\gamma^i = -\left(\frac{w^T x}{||w||} + \frac{b}{||w||}\right)$

$\gamma^i = y^i \left(\frac{w^T x^i}{||w||} + \frac{b}{||w||}\right).$

Geometric Margins

The **geometric margin** of (w, b) with respect to training data S is the minimum distance from any point to the hyperplane:

$$\gamma = \min_{i=1,..,m} \gamma^{(i)} = \min_{i=1,..,m} y^{(i)} \left(\frac{w}{||w||}^T x^{(i)} + \frac{b}{||w||} \right)$$

Geometric Margins

The **geometric margin** of (w, b) with respect to training data S is the minimum distance from any point to the hyperplane:

$$\gamma = \min_{i=1,..,m} \gamma^{(i)} = \min_{i=1,..,m} y^{(i)} \left(\frac{w}{||w||}^T x^{(i)} + \frac{b}{||w||} \right)$$
$$= \frac{1}{||w||} \min_{i=1,..,m} y^{(i)} \left(w^T x^{(i)} + b \right)$$
$$= \frac{1}{||w||} \hat{\gamma}$$

$\hat{\gamma}$

Geometric Margins

The **geometric margin** of (w, b) with respect to training data S is the minimum distance from any point to the hyperplane:

$$
\begin{aligned}
\gamma = \min_{i=1,..,m} \gamma^{(i)} &= \min_{i=1,..,m} y^{(i)} \left(\frac{w}{||w||}^T x^{(i)} + \frac{b}{||w||} \right) \\
&= \frac{1}{||w||} \min_{i=1,..,m} y^{(i)} \left(w^T x^{(i)} + b \right) \\
&= \frac{1}{||w||} \hat{\gamma}
\end{aligned}
$$

- $\hat{\gamma} = \gamma$ when $||w|| = 1$

Geometric Margins

The **geometric margin** of (w, b) with respect to training data S is the minimum distance from any point to the hyperplane:

$$\begin{aligned}\gamma = \min_{i=1,..,m} \gamma^{(i)} &= \min_{i=1,..,m} y^{(i)} \left(\frac{w}{||w||}^T x^{(i)} + \frac{b}{||w||} \right) \\ &= \frac{1}{||w||} \min_{i=1,..,m} y^{(i)} \left(w^T x^{(i)} + b \right) \\ &= \frac{1}{||w||} \hat{\gamma}\end{aligned}$$

(cw, cb) $\quad \hat{\gamma}' = c \cdot \hat{\gamma}$

- $\hat{\gamma} = \gamma$ when $||w|| = 1$
- Geometric margins are invariant to parameter scaling

$$y^{(i)} \left(\frac{cw}{||cw||}^T x^{(i)} + \frac{cb}{||cw||} \right) = y^{(i)} \left(\frac{w^T x}{||w||} + \frac{b}{||w||} \right) = \gamma^{(i)}$$

$c||w||$

Optimal Margin Classifier

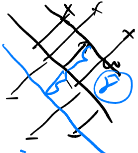
Assume data is linearly separable

Find (w, b) that maximize geometric margin $\gamma = \frac{\hat{\gamma}}{||w||}$ of the training data

$$\max_{\gamma, w, b} \frac{\hat{\gamma}}{||w||}$$

$$\text{s.t. } y^{(i)}(w^T x^{(i)} + b) \geq \hat{\gamma}, \; i = 1, \ldots, m$$

$\Rightarrow$ prediction is correct.



Optimal Margin Classifier

Assume data is linearly separable

Find (w, b) that maximize geometric margin $\gamma = \frac{\hat{\gamma}}{||w||}$ of the training data

$$\max_{\gamma, w, b} \frac{\hat{\gamma}}{||w||}$$
$$\text{s.t. } y^{(i)}(w^T x^{(i)} + b) \geq \hat{\gamma}, \; i = 1, \ldots, m$$

There exists some $\delta \in \mathbb{R}$ such that the functional margin of $(\delta w, \delta b)$ is $\hat{\gamma} = 1$

$$\max_{\gamma, w, b} \frac{1}{||w||}$$
$$\text{s.t. } y^{(i)}(w^T x^{(i)} + b) \geq 1 \; i = 1, \ldots, m$$

Optimal Margin Classifier

Assume data is linearly separable

Find (w, b) that maximize geometric margin $\gamma = \frac{\hat{\gamma}}{||w||}$ of the training data

$$\max_{\gamma,w,b} \frac{\hat{\gamma}}{||w||}$$
$$\text{s.t. } y^{(i)}(w^T x^{(i)} + b) \geq \hat{\gamma}, \; i = 1, \ldots, m$$

There exists some $\delta \in \mathbb{R}$ such that the functional margin of $(\delta w, \delta b)$ is $\hat{\gamma} = 1$

$$\max_{\gamma,w,b} \quad \frac{1}{||w||}$$
$$\text{s.t. } y^{(i)}(w^T x^{(i)} + b) \geq 1 \; i = 1, \ldots, m$$
$$\iff \min_{\gamma,w,b} \quad \frac{1}{2}||w||^2 \quad \frac{1}{2}w^T w$$
$$\text{s.t. } y^{(i)}(w^T x^{(i)} + b) \geq 1 \; i = 1, \ldots, m$$

Optimal Margin Classifier

Assume data is linearly separable

Find (w, b) that maximize geometric margin $\gamma = \frac{\hat{\gamma}}{||w||}$ of the training data

$$\max_{\gamma, w, b} \frac{\hat{\gamma}}{||w||}$$
$$\text{s.t. } y^{(i)}(w^T x^{(i)} + b) \geq \hat{\gamma}, \; i = 1, \ldots, m$$

There exists some $\delta \in \mathbb{R}$ such that the functional margin of $(\delta w, \delta b)$ is $\hat{\gamma} = 1$

$$\max_{\gamma, w, b} \frac{1}{||w||}$$
$$\text{s.t. } y^{(i)}(w^T x^{(i)} + b) \geq 1 \; i = 1, \ldots, m$$

$$\iff \min_{\gamma, w, b} \frac{1}{2}||w||^2$$
$$\text{s.t. } y^{(i)}(w^T x^{(i)} + b) \geq 1 \; i = 1, \ldots, m$$

general quadratic form → $w^T A w + B w$

can be solved using QP software

Review: Lagrange Duality

The **primal** optimization problem:

$$\begin{aligned} \min_{w} \quad & f(w) \\ s.t. \quad & g_i(w) \leq 0, i, \ldots, k \\ & h_i(w) = 0, i = 1, \ldots, l \end{aligned}$$

Review: Lagrange Duality

The **primal** optimization problem:

$$\begin{aligned} \min_{w} \quad & f(w) \\ s.t. \quad & g_i(w) \leq 0, i, \ldots, k \\ & h_i(w) = 0, i = 1, \ldots, l \end{aligned}$$

Define the **generalized Lagrange function** :

$$L(w, \alpha, \beta) = f(w) + \sum_{i=1}^{k} \alpha_i g_i(w) + \sum_{i=1}^{l} \beta_i h_i(w)$$

α_i and β_i are called the **Lagrange multipliers** Lagrange multipliers

$\alpha_i \geq 0.$

For a given w,

$$\theta_P(w) = \max_{\alpha,\beta:\alpha_i \geq 0} L(w, \alpha, \beta)$$

$$= \max_{\alpha,\beta:\alpha_i \geq 0} f(w) + \sum_{i=1}^{k} \alpha_i g_i(w) + \sum_{i=1}^{l} \beta_i h_i(w)$$

For a given w,

$$\theta_P(w) = \max_{\alpha,\beta:\alpha_i \geq 0} L(w, \alpha, \beta)$$

$$= \max_{\alpha,\beta:\alpha_i \geq 0} f(w) + \sum_{i=1}^{k} \alpha_i g_i(w) + \sum_{i=1}^{l} \beta_i h_i(w)$$

Recall the primal constraints: $g_i(w) \leq 0$ and $h_i(w) = 0$:

- $\theta_P(w) = f(w)$ if w satisfies primal constraints

For a given w,

$$\theta_P(w) = \max_{\alpha,\beta:\alpha_i \geq 0} L(w,\alpha,\beta)$$

$$= \max_{\alpha,\beta:\alpha_i \geq 0} f(w) + \sum_{i=1}^{k} \alpha_i g_i(w) + \sum_{i=1}^{l} \beta_i h_i(w)$$

Recall the primal constraints: $g_i(w) \leq 0$ and $h_i(w) = 0$:

- $\theta_P(w) = f(w)$ if w satisfies primal constraints
- $\theta_P(w) = \infty$ otherwise

The primal problem (alternative form)

$$\min_w \theta_P(w) = \min_w \max_{\alpha,\beta:\alpha_i \geq 0} L(w,\alpha,\beta)$$

The primal problem (P)

$$p^* = \min_w \theta_P(w) = \min_w \max_{\alpha,\beta:\alpha_i \geq 0} L(w,\alpha,\beta)$$

The dual problem (D)

$$d^* = \max_{\alpha,\beta:\alpha_i \geq 0} \theta_D(\alpha,\beta) = \max_{\alpha,\beta:\alpha_i \geq 0} \min_w L(w,\alpha,\beta)$$

$d^* \overset{?}{\longleftrightarrow} p^*$ max-min inequality

The primal problem (P)

$$p^* = \min_w \theta_P(w) = \min_w \max_{\alpha,\beta:\alpha_i \geq 0} L(w, \alpha, \beta)$$

The dual problem (D)

$$d^* = \max_{\alpha,\beta:\alpha_i \geq 0} \theta_D(\alpha, \beta) = \max_{\alpha,\beta:\alpha_i \geq 0} \min_w L(w, \alpha, \beta)$$

In general, $d^* \leq p^*$ (max-min inequality)

$f(w, z)$

$\max_w \min_z f(w, z) \leq \min_z \max_w f$

The primal problem (P)

$$p^* = \min_w \theta_P(w) = \min_w \max_{\alpha,\beta:\alpha_i\geq 0} L(w,\alpha,\beta)$$

The dual problem (D)

$$d^* = \max_{\alpha,\beta:\alpha_i\geq 0} \theta_D(\alpha,\beta) = \max_{\alpha,\beta:\alpha_i\geq 0} \min_w L(w,\alpha,\beta)$$

In general, $d^* \leq p^*$ (max-min inequality)

Theorem (Lagrange Duality)

Suppose f and all g_i's are convex, all h_i's are affine, and there exists some w such that $g_i(w) < 0$ for all i (strictly feasible). **There must exists w^*, α^*, β^* so that w^* is the solution to P and α^*,β^* are the solution to D, and**

$$p^* = d^* = L(w^*, \alpha^*, \beta^*)$$

Karush-Kuhn-Tucker (KKT) conditions

Under previous conditions, w^*, α^*, β^* are solutions of P and D **if and only if** they statisty the following conditions:

$$\frac{\delta}{\delta w_i} L(w^*, \alpha^*, \beta^*) = 0, \;\; i = 1, \dots n \tag{1}$$

$$\frac{\delta}{\delta \beta_i} L(w^*, \alpha^*, \beta^*) = 0, \;\; i = 1, \dots l \tag{2}$$

$$\alpha_i^* g_i(w^*) = 0, \;\; i = 1, \dots, k \tag{3}$$

$$g_i(w^*) \leq 0, \;\; i = 1, \dots, k \tag{4}$$

$$\alpha^* \geq 0, \;\; i = 1, \dots, k \tag{5}$$

Equation 3 is called the **complementary slackness condition**.

Optimal Margin Classifier

$$\left[\begin{array}{l}\min_{w} f(w) \\ \text{s.t. } g_i(w) \le 0. \quad i=1,\dots,m.\end{array}\right.$$

$$L(w,\alpha) = f(w) + \sum_{i=1}^{m} \alpha_i g_i(w) \qquad \alpha_i \ge 0.$$

Optimal margin classifier

$$\min_{\gamma,w,b} \frac{1}{2}||w||^2$$

$$\text{s.t. } y^{(i)}(w^T x^{(i)} + b) \ge 1 \; i = 1,\dots,m$$

$$-\left(y^{(i)}(w^T x^{(i)} + b) - 1\right) \le 0.$$

- $f(w) = \frac{1}{2}||w||^2$
- $g_i(w) = -\left(y^{(i)}(w^T x^{(i)} + b) - 1\right)$

Generalized Lagrangian function:

② $\frac{\partial L}{\partial b} = 0 \Rightarrow \sum_{i=1}^{m} \alpha_i y^{(i)} = 0.$

$$L(w,b,\alpha) = \frac{1}{2}||w||^2 - \sum_{i}^{m} \alpha_i \left[y^{(i)}(w^T x^{(i)} + b) - 1\right]$$

① $\frac{\partial L}{\partial w} = 0.$ $\quad \frac{\partial L(w,b,\alpha)}{\partial w_i} = w - \sum_{i=1}^{m} \alpha_i y^{(i)} x^{(i)} = 0. \Rightarrow w = \sum_{i=1}^{m} \alpha_i y^{(i)} x^{(i)}$

By the complementary slackness condition in KKT:

$$\alpha_i^* g_i(w^*) = 0, \; i = 1, \ldots, k$$

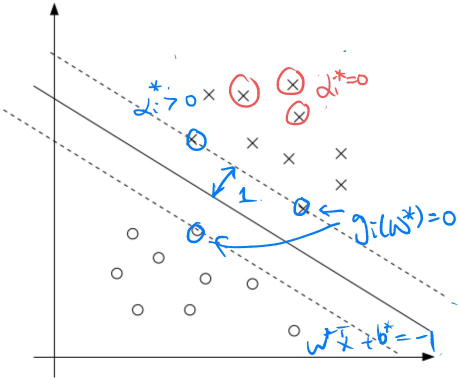
$$\alpha_i^* > 0 \iff g_i(w^*) = -y^{(i)}({w^*}^T x^{(i)} + b) + 1 = 0$$

By the complementary slackness condition in KKT:

$$\alpha_i^* g_i(w^*) = 0, \; i = 1, \ldots, k$$

$$\alpha_i^* > 0 \Longleftrightarrow g_i(w^*) = -y^{(i)}({w^*}^T x^{(i)} + b) + 1 = 0$$

Training examples $(x^{(i)}, y^{(i)})$ such that $y^{(i)}({w^*}^T x^{(i)} + b) = 1$ are called **support vectors**

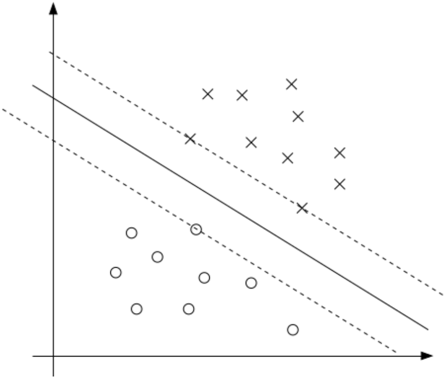


Support vectors lie on hyperplane ${w^*}^T x + b = 1$ when $y^{(i)} = 1$, or ${w^*}^T x + b = -1$ when $y^{(i)} = -1$

By the complementary slackness condition in KKT:

$$\alpha_i^* g_i(w^*) = 0, \; i = 1, \dots, k$$
$$\alpha_i^* > 0 \iff g_i(w^*) = -y^{(i)}({w^*}^T x^{(i)} + b) + 1 = 0$$

Training examples $(x^{(i)}, y^{(i)})$ such that $y^{(i)}({w^*}^T x^{(i)} + b) = 1$ are called **support vectors**



Support vectors lie on hyperplane ${w^*}^T x + b = 1$ when $y^{(i)} = 1$, or ${w^*}^T x + b = -1$ when $y^{(i)} = -1$

Constraints $g_i(w) \leq 0$ is only **active** on support vectors

Dual optimization problem: *(Check derivation)*

$$\max_{\alpha} W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} y^{(i)} y^{(j)} \alpha_i \alpha_j \langle x^{(i)}, x^{(j)} \rangle$$

$$s.t. \ \alpha_i \geq 0, i = 1, \ldots, m$$

$$\sum_{i=1}^{m} \alpha_i y^{(i)} = 0$$

① $\frac{\partial L}{\partial w} = 0 \Rightarrow w^* = \sum_{i=1}^{m} \alpha_i y^{(i)} x^{(i)}$

② $\frac{\partial L}{\partial b} = 0 \Rightarrow \sum_{i=1}^{m} \alpha_i y^{(i)} = 0$

$$L(w,b,\alpha) = \frac{1}{2} w^T w - \sum_{i=1}^{m} \alpha_i \left(y^{(i)} (w^T x^{(i)} + b) - 1 \right)$$

$\alpha_i y^{(i)} w^T x^{(i)} + \alpha_i y^{(i)} b - \alpha_i$

By (1)

$$= \frac{1}{2} w^T \left(\sum_{i=1}^{m} \alpha_i y^{(i)} x^{(i)} \right) - \sum_{i=1}^{m} \alpha_i y^{(i)} w^T x^{(i)} - \sum_{i=1}^{m} \alpha_i y^{(i)} b + \sum_{i=1}^{m} \alpha_i$$

by (2) $= 0$.

$$= -\frac{1}{2} \left(\sum_{j=1}^{m} \alpha_j y^{(j)} x^{(j)} \right)^T \left(\sum_{i=1}^{m} \alpha_i y^{(i)} x^{(i)} \right) + \sum_{i=1}^{m} \alpha_i$$

$$= -\frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} y^{(i)} y^{(j)} \alpha_i \alpha_j \left(x^{(i)T} x^{(j)} \right) + \sum_{i=1}^{m} \alpha_i = W(\alpha).$$

$\langle x^{(i)}, x^{(j)} \rangle$

Dual optimization problem: *(Check derivation)*

$$\max_{\alpha} W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} y^{(i)} y^{(j)} \alpha_i \alpha_j \langle x^{(i)}, x^{(j)} \rangle$$

$$s.t.\ \alpha_i \geq 0, i = 1, \ldots, m$$

$$\sum_{i=1}^{m} \alpha_i y^{(i)} = 0$$

Solution to the primal problem:

$$w^* = \sum_{i} \alpha_i^* y^{(i)} x^{(i)}$$

$$b^* = -\frac{1}{2} \left(\max_{i:y^{(i)}=-1} {w^*}^T x^{(i)} + \min_{i:y^{(i)}=1} {w^*}^T x^{(i)} \right)$$

Dual optimization problem: *(Check derivation)*

$$\max_{\alpha} W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} y^{(i)} y^{(j)} \alpha_i \alpha_j \langle x^{(i)}, x^{(j)} \rangle$$

$$s.t. \ \alpha_i \geq 0, i = 1, \ldots, m$$

$$\sum_{i=1}^{m} \alpha_i y^{(i)} = 0$$

Solution to the primal problem:

$$w^* = \sum_i \alpha_i^* y^{(i)} x^{(i)}$$

$$b^* = -\frac{1}{2} \left(\max_{i:y^{(i)}=-1} {w^*}^T x^{(i)} + \min_{i:y^{(i)}=1} {w^*}^T x^{(i)} \right)$$

For a new sample z, the SVM prediction is sign $\left[{w^*}^T z + b \right]$

$w^T z + b = \sum_{i=1}^{m} \alpha_i y^{(i)} \langle x^{(i)}, z \rangle + b$

Linear SVM Summary

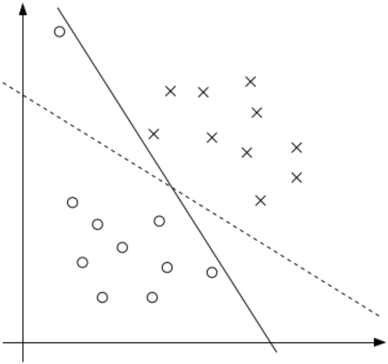
- Input:: m training samples $(x^{(i)}, y^{(i)}), y^i \in \{-1, 1\}$
- Output: optimal parameters w^*, b^*
- Step 1: solve the dual optimization problem

$$\alpha^* = \max_{\alpha} W(\alpha)$$
$$s.t.\ \alpha_i \geq 0, \sum_{i=1}^{m} \alpha_i y^{(i)} = 0, i = 1, \ldots, m$$

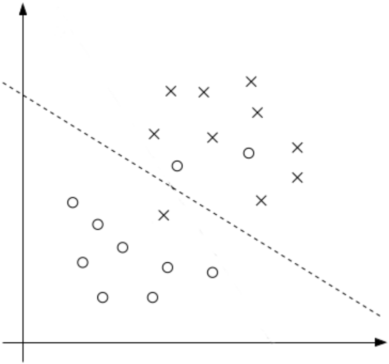
- Step 2: compute the optimal parameters w^*, b^*

$$w^* = \sum_i \alpha_i^* y^{(i)} x^{(i)}$$
$$b^* = -\frac{1}{2}\left(\max_{i:y^{(i)}=-1} {w^*}^T x^{(i)} + \min_{i:y^{(i)}=1} {w^*}^T x^{(i)} \right)$$

Limitations of the basic SVM



Outliers



Non-linearly separable cases

Soft Margin SVM

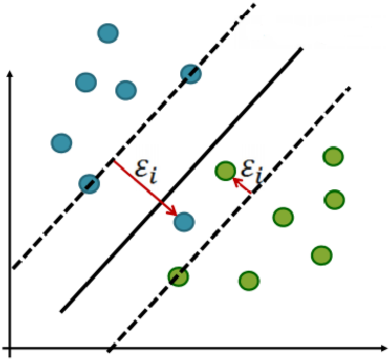
Functional margin $1 - \xi_i \leq 1$:

$$\min_{w,b,\xi} \frac{1}{2}||w||^2 + C\sum_{i=1}^{m} \xi_i$$

$$s.t.\ y^{(i)}(w^T x^{(i)} + b) \geq 1 - \xi_i$$

$$\xi_i \geq 0, i = 1, \ldots, m$$

- C: relative weight on the regularizer
- L_1 regularization let most $\xi_i = 0$, such that their functional margins $1 - \xi_i = 1$



Soft Margin SVM

The generalized Lagrangian function:

$$L(w, b, \xi, \alpha, r) = \frac{1}{2}||w||^2 + C\sum_{i=1}^{m}\xi_i - \sum_{i}^{m}\alpha_i\left[y^{(i)}(w^T x^{(i)} + b) - 1 + \xi_i\right]$$
$$- \sum_{i=1}^{m} r_i\xi_i$$

Soft Margin SVM

The generalized Lagrangian function:

$$L(w, b, \xi, \alpha, r) = \frac{1}{2}||w||^2 + C\sum_{i=1}^{m}\xi_i - \sum_{i}^{m}\alpha_i\left[y^{(i)}(w^T x^{(i)} + b) - 1 + \xi_i\right] - \sum_{i=1}^{m} r_i \xi_i$$

Dual problem:

Soft Margin SVM

The generalized Lagrangian function:

$$L(w, b, \xi, \alpha, r) = \frac{1}{2}||w||^2 + C\sum_{i=1}^{m}\xi_i - \sum_{i}^{m}\alpha_i\left[y^{(i)}(w^T x^{(i)} + b) - 1 + \xi_i\right] - \sum_{i=1}^{m} r_i\xi_i$$

Dual problem:

$$\max_{\alpha} W(\alpha) = \sum_{i=1}^{m}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{m} y^{(i)}y^{(j)}\alpha_i\alpha_j\langle x^{(i)}, x^{(j)}\rangle$$

$$s.t.\ 0 \leq \alpha_i \leq C, i = 1, \ldots, m$$

$$\sum_{i=1}^{m}\alpha_i y^{(i)} = 0$$

w^* is the same as the non-regularizing case, but b^* has changed.

Soft Margin SVM

Dual problem:

$$\max_{\alpha} W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} y^{(i)} y^{(j)} \alpha_i \alpha_j \langle x^{(i)}, x^{(j)} \rangle$$

$$s.t.\ 0 \leq \alpha_i \leq C, i = 1, \ldots, m$$

$$\sum_{i=1}^{m} \alpha_i y^{(i)} = 0$$

By the KKT dual-complentary conditions, for all i, $\alpha_i^* g_i(w^*) = 0$

$$\alpha_i = 0 \iff$$
$$\alpha_i = C \iff$$
$$0 < \alpha_i < C \iff$$

Soft Margin SVM

Dual problem:

$$\max_{\alpha} W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} y^{(i)} y^{(j)} \alpha_i \alpha_j \langle x^{(i)}, x^{(j)} \rangle$$

$$s.t.\ 0 \leq \alpha_i \leq C, i = 1, \ldots, m$$

$$\sum_{i=1}^{m} \alpha_i y^{(i)} = 0$$

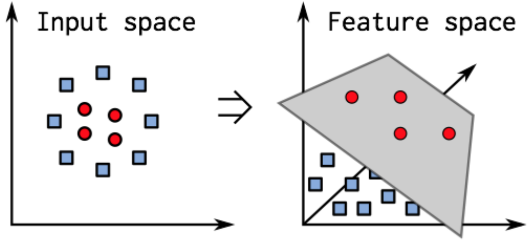
By the KKT dual-complentary conditions, for all i, $\alpha_i^* g_i(w^*) = 0$

$\alpha_i = 0$	$\iff$	$y^{(i)}(w^T x^{(i)} + b) \geq 1$	correct side of margin
$\alpha_i = C$	$\iff$	$y^{(i)}(w^T x^{(i)} + b) \leq 1$	wrong side of margin
$0 < \alpha_i < C$	$\iff$	$y^{(i)}(w^T x^{(i)} + b) = 1$	at margin

Kernel SVM

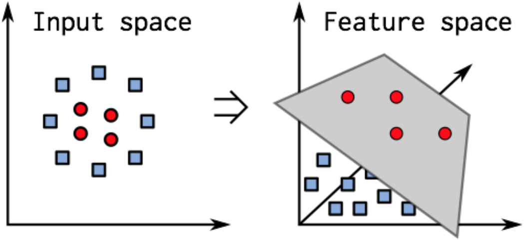
Non-linear SVM

For non-separable data, we can use the **kernel trick**: Map input values $x \in \mathbb{R}^d$ to a higher dimension $\phi(x) \in \mathbb{R}^D$, such that the data becomes separable.



Non-linear SVM

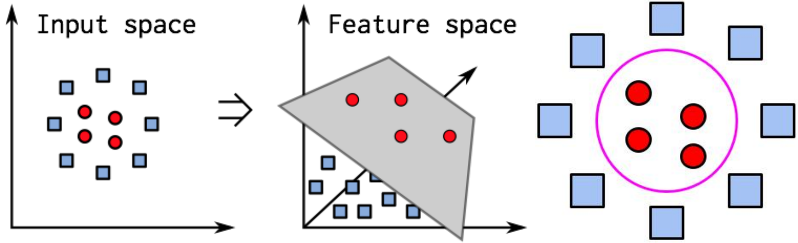
For non-separable data, we can use the **kernel trick**: Map input values $x \in \mathbb{R}^d$ to a higher dimension $\phi(x) \in \mathbb{R}^D$, such that the data becomes separable.



- ϕ is called a **feature mapping**.

Non-linear SVM

For non-separable data, we can use the **kernel trick**: Map input values $x \in \mathbb{R}^d$ to a higher dimension $\phi(x) \in \mathbb{R}^D$, such that the data becomes separable.



- ϕ is called a **feature mapping**.
- The classification function $w^T x + b$ becomes nonlinear:$w^T \phi(x) + b$

Kernel Function

Given a feature mapping ϕ, we define the **kernel function** to be

$$K(x, z) = \phi(x)^T \phi(z)$$

Kernel Function

Given a feature mapping ϕ, we define the **kernel function** to be

$$K(x, z) = \phi(x)^T \phi(z)$$

Some kernel functions are easier to compute than $\phi(x)$, e.g.

$$K(x, z) = (x^T z)^2$$

Kernel Function

Given a feature mapping ϕ, we define the **kernel function** to be

$$K(x, z) = \phi(x)^T \phi(z)$$

Some kernel functions are easier to compute than $\phi(x)$, e.g.

$$\begin{aligned} K(x, z) = (x^T z)^2 &= \sum_{i=1}^{n} x_i, z_i \sum_{j=1}^{n} x_j, z_j = \sum_{i=1}^{n} \sum_{j=1}^{n} x_i, x_j, z_i, z_j \\ &= \phi(x)^T \phi(z) \end{aligned}$$

Kernel Function

Given a feature mapping ϕ, we define the **kernel function** to be

$$K(x, z) = \phi(x)^T \phi(z)$$

Some kernel functions are easier to compute than $\phi(x)$, e.g.

$$\begin{aligned} K(x, z) = (x^T z)^2 &= \sum_{i=1}^{n} x_i, z_i \sum_{j=1}^{n} x_j, z_j = \sum_{i=1}^{n} \sum_{j=1}^{n} x_i, x_j, z_i, z_j \\ &= \phi(x)^T \phi(z) \end{aligned}$$

where $\phi(x) = \begin{bmatrix} x_1 x_1 \\ x_1 x_2 \\ \vdots \\ x_n x_{n-1} \\ x_n x_n \end{bmatrix}$ takes $O(n^2)$ operations to compute,
while $(x^T z)^2$ only takes $O(n)$

Kernel SVM

In the dual problem, replace $\langle x_i, y_j \rangle$ with $\langle \phi(x_i), \phi(y_i) \rangle = K(x_i, x_j)$

$$\max_{\alpha} W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} y^{(i)} y^{(j)} \alpha_i \alpha_j K(x_i, x_j)$$

$$s.t.\ 0 \leq \alpha_i \leq C, i = 1, \ldots, m$$

$$\sum_{i=1}^{m} \alpha_i y^{(i)} = 0$$

Kernel SVM

In the dual problem, replace $\langle x_i, y_j \rangle$ with $\langle \phi(x_i), \phi(y_i) \rangle = K(x_i, x_j)$

$$\max_{\alpha} W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} y^{(i)} y^{(j)} \alpha_i \alpha_j K(x_i, x_j)$$

$$s.t.\ 0 \le \alpha_i \le C, i = 1, \ldots, m$$

$$\sum_{i=1}^{m} \alpha_i y^{(i)} = 0$$

No need to compute $w^* = \sum_{i=1}^{m} \alpha_i^* y^{(i)} \phi(x^{(i)})$ explicitly since

$$f(x) = w^T \phi(x) + b = \left(\sum_{i=1}^{m} \alpha_i y^{(i)} \phi(x^{(i)}) \right)^T \phi(x) + b$$

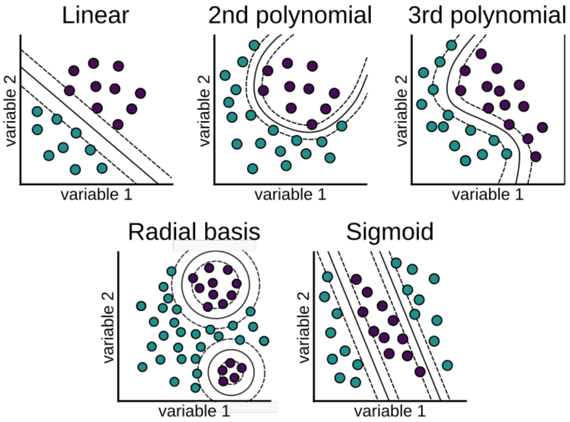
$$= \sum_{i=1}^{m} \alpha_i y^{(i)} \langle \phi(x^{(i)}), \phi(x) \rangle + b$$

$$= \sum_{i=1}^{m} \alpha_i y^{(i)} K(x^{(i)}, x) + b$$

Kernel Matrix

kernel functions measure the similarity between samples x, z, e.g.

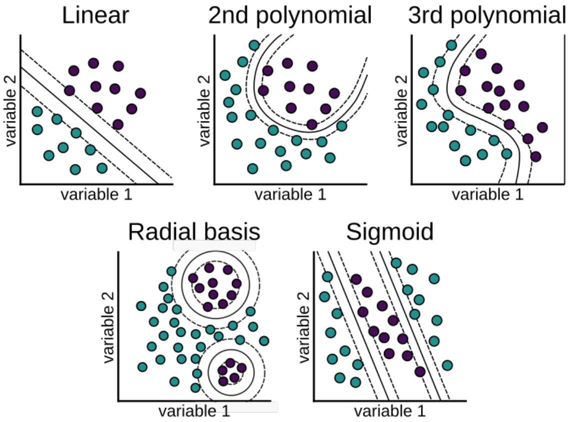
- Linear kernel: $K(x, z) = (x^T z)$
- Polynomial kernel: $K(x, z) = (x^T z + 1)^p$
- Gaussian / radial basis function (RBF) kernel:
  $K(x, z) = \exp\left(-\frac{||x-z||^2}{2\sigma^2}\right)$



Kernel Matrix

kernel functions measure the similarity between samples x, z, e.g.

- Linear kernel: $K(x, z) = (x^T z)$
- Polynomial kernel: $K(x, z) = (x^T z + 1)^p$
- Gaussian / radial basis function (RBF) kernel: $K(x, z) = \exp\left(-\frac{||x-z||^2}{2\sigma^2}\right)$



Can any function $K(x, y)$ be a kernel function?

Kernel Matrix

Represent kernel function as a matrix $K \in \mathbb{R}^{n \times n}$ where $K_{i,j} = K(x_i, x_j) = \phi(x_i)\phi(x_j)$.

Kernel Matrix

Represent kernel function as a matrix $K \in \mathbb{R}^{n \times n}$ where $K_{i,j} = K(x_i, x_j) = \phi(x_i)\phi(x_j)$.

Theorem (Mercer)

Let $K : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ Then K is a valid (Mercer) kernel if and only if for any finite training set $\{x^{(i)}, \ldots, x^{(m)}\}$, K is symmetric positive semi-definite.

i.e. $K_{i,j} = K_{j,i}$ and $x^T K x \geq 0$ for all $x \in \mathbb{R}^n$

Kernel SVM Summary

- Input: m training samples $(x^{(i)}, y^{(i)}), y^i \in \{-1, 1\}$, kernel function $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, constant $C > 0$
- Output: non-linear decision function $f(x)$
- Step 1: solve the dual optimization problem for α^*

$$\max_{\alpha} W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} y^{(i)} y^{(j)} \alpha_i \alpha_j K(x^{(i)}, x^{(j)})$$

$$s.t.\ 0 \leq \alpha_i \leq C, \sum_{i=1}^{m} \alpha_i y^{(i)} = 0, i = 1, \ldots, m$$

- Step 2: compute the optimal decision function

$$b^* = y^{(j)} - \sum_{i=1}^{m} \alpha_i^* y^{(i)} K(x^{(i)}, x^{(j)}) \text{ for some } 0 \leq \alpha_j \leq C$$

$$f(x) = \sum_{i=1}^{m} \alpha_i y^{(i)} K(x^{(i)}, x) + b^*$$

In practice, it's more efficient to compute kernel matrix K in advance.

SVM in Practice

Sequential Minimal Optimization: a fast algorithm for training soft margin kernel SVM

- Break a large SVM problem into smaller chunks, update two α_i's at a time
- Implemented by most SVM libraries.

SVM in Practice

Sequential Minimal Optimization: a fast algorithm for training soft margin kernel SVM

- Break a large SVM problem into smaller chunks, update two α_i's at a time
- Implemented by most SVM libraries.

Other related algorithms

- Support Vector Regression (SVR)
- Multi-class SVM (Koby Crammer and Yoram Singer. 2002. *On the algorithmic implementation of multiclass kernel-based vector machines*. J. Mach. Learn. Res. 2 (March 2002), 265-292.)