

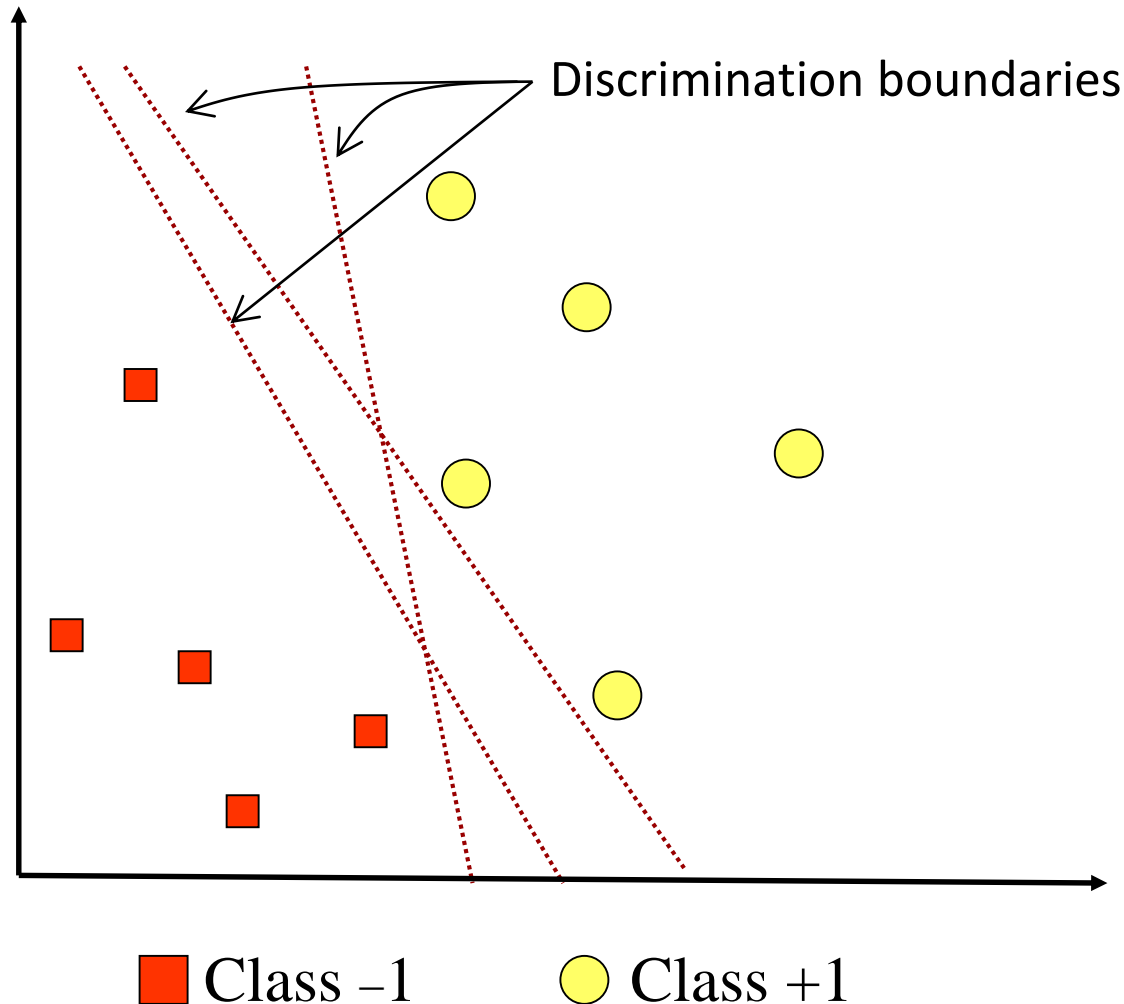
PERTEMUAN 9

Support Vector Machine

Support Vector Machine

- Diperkenalkan oleh Vapnik (1992)
- Support Vector Machine memenuhi 3 syarat utama sebuah metode PR
 - Robustness
 - Theoretically Analysis
 - Feasibility
- Pada prinsipnya bekerja sebagai binary classifier. Saat ini tengah dikembangkan untuk multiclass problem
- Structural-Risk Minimization

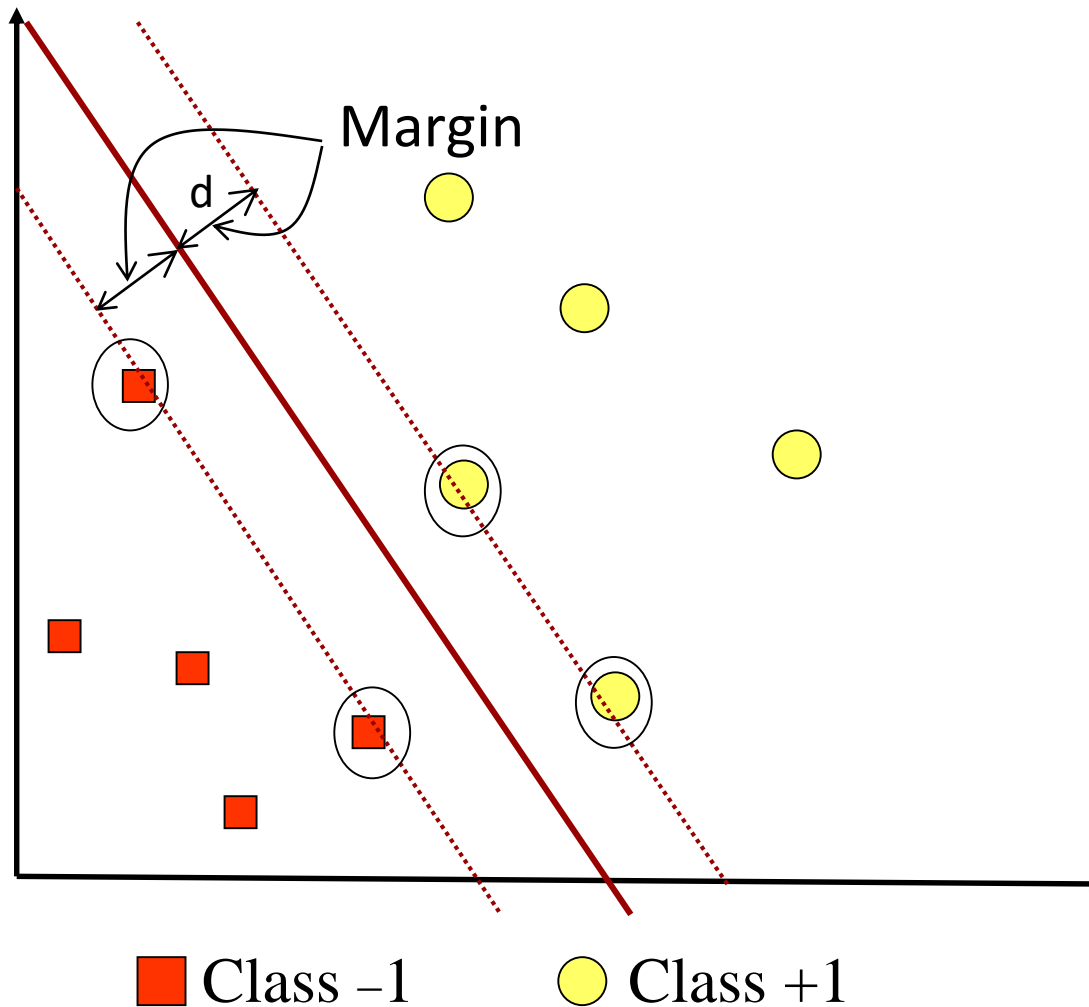
Binary Classification



Optimal Hyperplane by SVM

- Margin (d) = minimum distance antara hyperplane and training samples
- Hyperplane yang paling baik diperoleh dengan memaksimalkan nilai margin
- Hyperplane yang paling baik itu akan melewati pertengahan antara kedua class
- Sample yang paling dekat lokasinya terhadap hyperplane disebut support vector
- Proses learning dalam SVM : mencari support vector untuk memperoleh hyperplane yang terbaik

Optimal Hyperplane by SVM



Kernel & Non-Linear SVM

- Latar belakang
- Kelemahan Linear Learning-Machines
- Representasi data & Kernel
- Non linear SVM

Latar belakang

- Machine Learning
 - Supervised learning: berikan satu set input-output data, dan buatlah satu model yang mampu memprediksi dengan benar output terhadap data baru. Contoh : pattern classification, regression
 - Unsupervised learning: berikan satu set data (tanpa output yang bersesuaian), dan ekstraklah suatu informasi bermanfaat. Contoh : clustering, Principal Component Analysis
- Apabila banyaknya data yang diberikan “cukup banyak”, metode apapun yang dipakai akan menghasilkan model yang bagus
- Tetapi jika data yang diberikan sangat terbatas, untuk mendapatkan performa yang baik, mutlak perlu memakai informasi spesifik masalah yang dipecahkan (prior knowledge of the problem domain). Contoh : masalah yg dipecahkan apakah berupa character recognition, analisa sekuens DNA, voice dsb. Prior knowledge seperti “masalah yg dianalisa adalah DNA” ini tidak dapat dinyatakan dengan angka.

- Pemanfaatan prior knowledge :
 - Fungsi Kernel (kemiripan sepasang data)
 - Probabilistic model of data distribution (Gaussian, Markov model, HMM, dsb)
- Pemakaian Kernel :

user memanfaatkan pengetahuannya mengenai domain masalah yang dipecahkan dengan mendefinisikan fungsi kernel untuk mengukur kemiripan sepasang data

Linear Learning Machines

- Kelebihan :
 - Algoritma pembelajarannya simple dan mudah dianalisa secara matematis
- Kelemahan
 - Perceptron (salah satu contoh linear learning machine) hanya mampu memecahkan problem klasifikasi linear (Minsky & Papert)
 - Umumnya masalah dari *real-world domain* bersifat non-linear dan kompleks, sehingga linear learning machines tidak mampu dipakai memecahkan masalah riil.

Representasi Data & Kernel

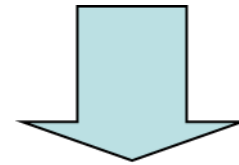
- Representasi data seringkali mampu menyederhanakan satu masalah
- Data yang dipetakan ke ruang vektor berdimensi lebih tinggi, memiliki potensi lebih besar untuk dapat dipisahkan secara linear (Cover theorem)
- Masalah : semakin tinggi dimensi suatu data, akan mengakibatkan tertimpa kutukan dimensi tinggi Curse of dimensionality.
 - turunnya generalisasi model
 - meningkatnya komputasi yang diperlukan
- Pemakaian konsep Kernel akan mengatasi masalah di atas

Representasi Data & Kernel

- Representasi data seringkali mampu menyederhanakan satu masalah
- Formula sebagaimana pada persamaan (20) tidak dapat dipecahkan dengan linear machines
- Representasi dengan menghasilkan (21) yang berupa persamaan linear, sehingga bisa dipecahkan dengan linear machines

Newton's law gravitation

$$f(m_1, m_2, r) = C \frac{m_1 m_2}{r^2} \quad (20)$$



$$\begin{aligned} g(x, y, z) &= \ln f(m_1, m_2, r) \\ &= \ln C + \ln m_1 + \ln m_2 - 2 \ln r \\ &= c + x + y - 2z \end{aligned} \quad (21)$$

Representasi Data & Kernel

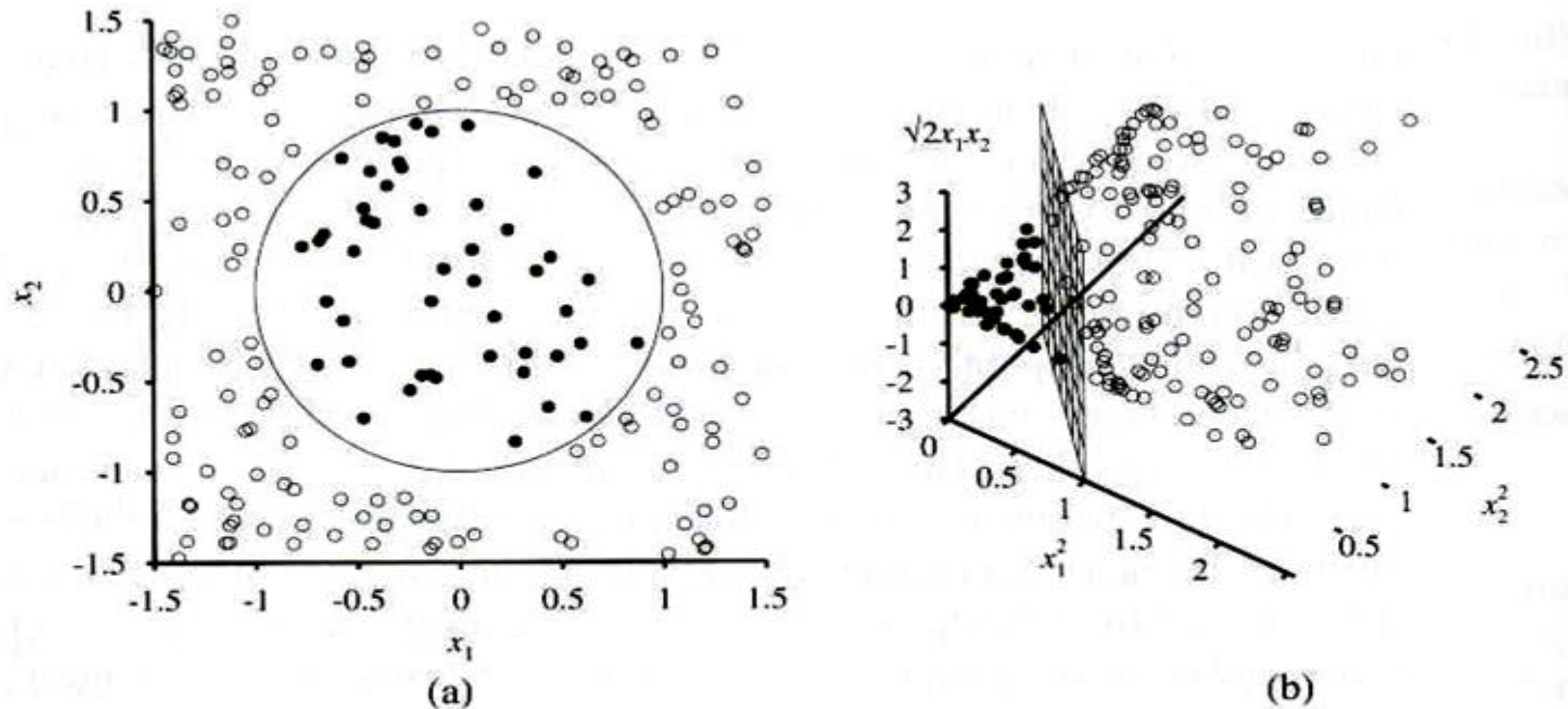
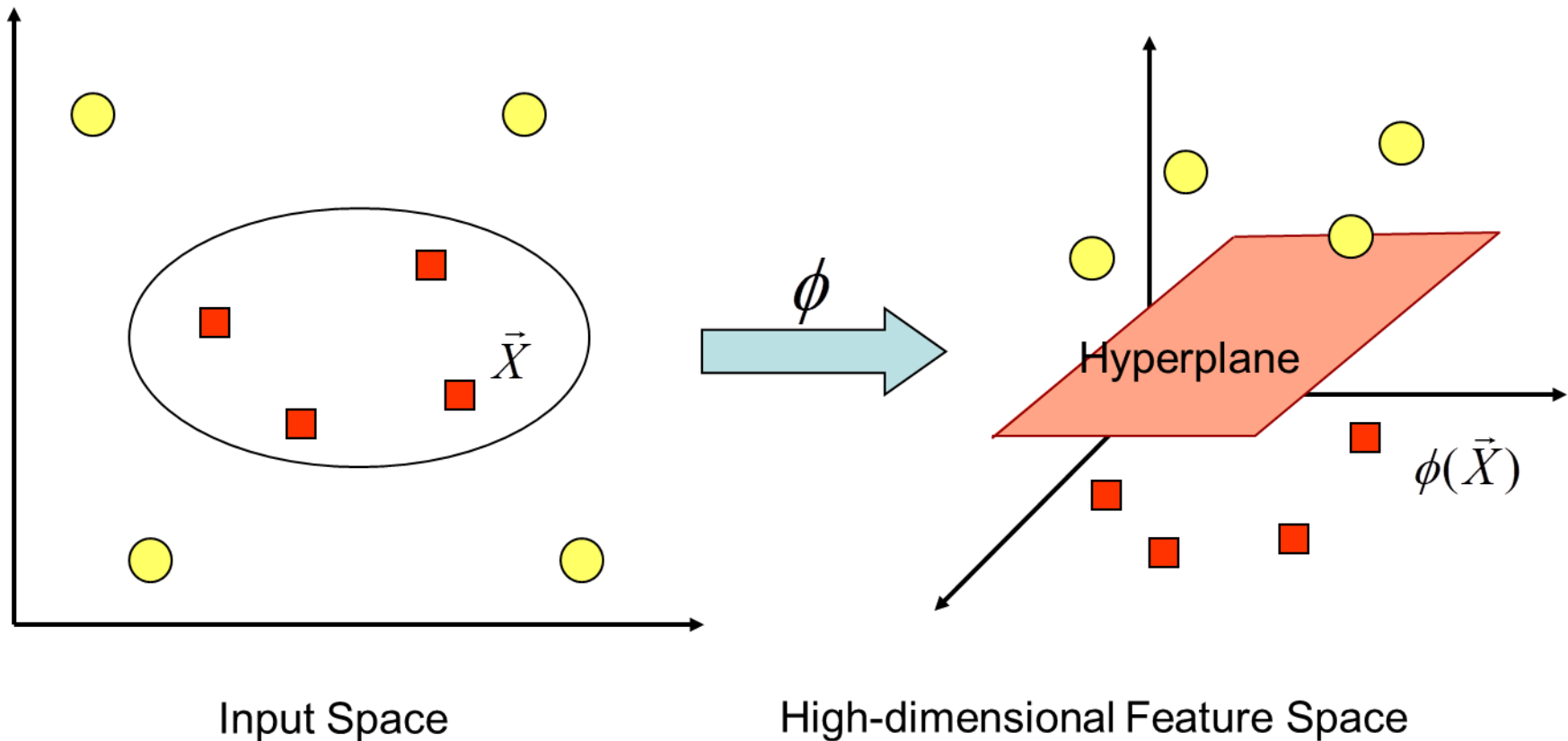


Figure 20.27 (a) A two-dimensional training with positive examples as black circles and negative examples as white circles. The true decision boundary, $x_1^2 + x_2^2 \leq 1$, is also shown. (b) The same data after mapping into a three-dimensional input space $(x_1^2, x_2^2, \sqrt{2}x_1x_2)$. The circular decision boundary in (a) becomes a linear decision boundary in three dimensions.

Non Linear Classification dalam SVM

Non Linear Classification dalam SVM



- Linear learning machines dapat ditulis dalam dua bentuk: primal form & dual form

$$\bar{w} = \sum_{i=1}^l \alpha_i y_i \vec{x}_i$$

$$\text{primal} \quad f(\vec{x}) = \langle \bar{w} \cdot \vec{\phi}(x) \rangle + b = \sum_{i=1}^{\dim} w_i \phi_i(x) + b$$

$$\text{dual} \quad f(\vec{x}) = \sum_{i=1}^l \alpha_i y_i \langle \phi(\vec{x}_i) \cdot \phi(\vec{x}) \rangle + b$$

- Hypotheses function dapat direpresentasikan sebagai kombinasi linear training points. Sehingga decision rule dapat dievaluasi berdasarkan inner product (dot product) antara test point & training points
- Keuntungan dual form : dimensi feature space tidak mempengaruhi perhitungan. Informasi yang dipakai hanya Gram matrix

Gram Matrix

$$G = (\langle \vec{x}_i \cdot \vec{x}_j \rangle)_{i,j=1}^l$$

$$\begin{pmatrix} \langle \vec{x}_1 \cdot \vec{x}_1 \rangle & \cdots & \langle \vec{x}_l \cdot \vec{x}_1 \rangle \\ \vdots & \ddots & \vdots \\ \langle \vec{x}_1 \cdot \vec{x}_l \rangle & \cdots & \langle \vec{x}_l \cdot \vec{x}_l \rangle \end{pmatrix}$$

Fungsi Kernel

Representasi dual form

$$f(\vec{x}) = \sum_{i=1}^l \alpha_i y_i \langle \phi(\vec{x}_i) \cdot \phi(\vec{x}) \rangle + b$$



$$\langle \phi(\vec{x}_i) \cdot \phi(\vec{x}) \rangle$$

Bisa dihitung secara IMPLISIT. Yaitu tidak perlu mengetahui wujud fungsi pemetaan melainkan langsung menghitungnya lewat fungsi KERNEL

$$\langle \phi(\vec{x}_i) \cdot \phi(\vec{x}) \rangle = K(\vec{x}_i, \vec{x})$$

Contoh-contoh Fungsi Kernel

Polynomial

$$K(\vec{x}, \vec{y}) = \langle \vec{x} \cdot \vec{y} \rangle^d$$

Gaussian

$$K(\vec{x}, \vec{y}) = \exp\left(-\frac{\|\vec{x} - \vec{y}\|^2}{2\sigma^2}\right)$$

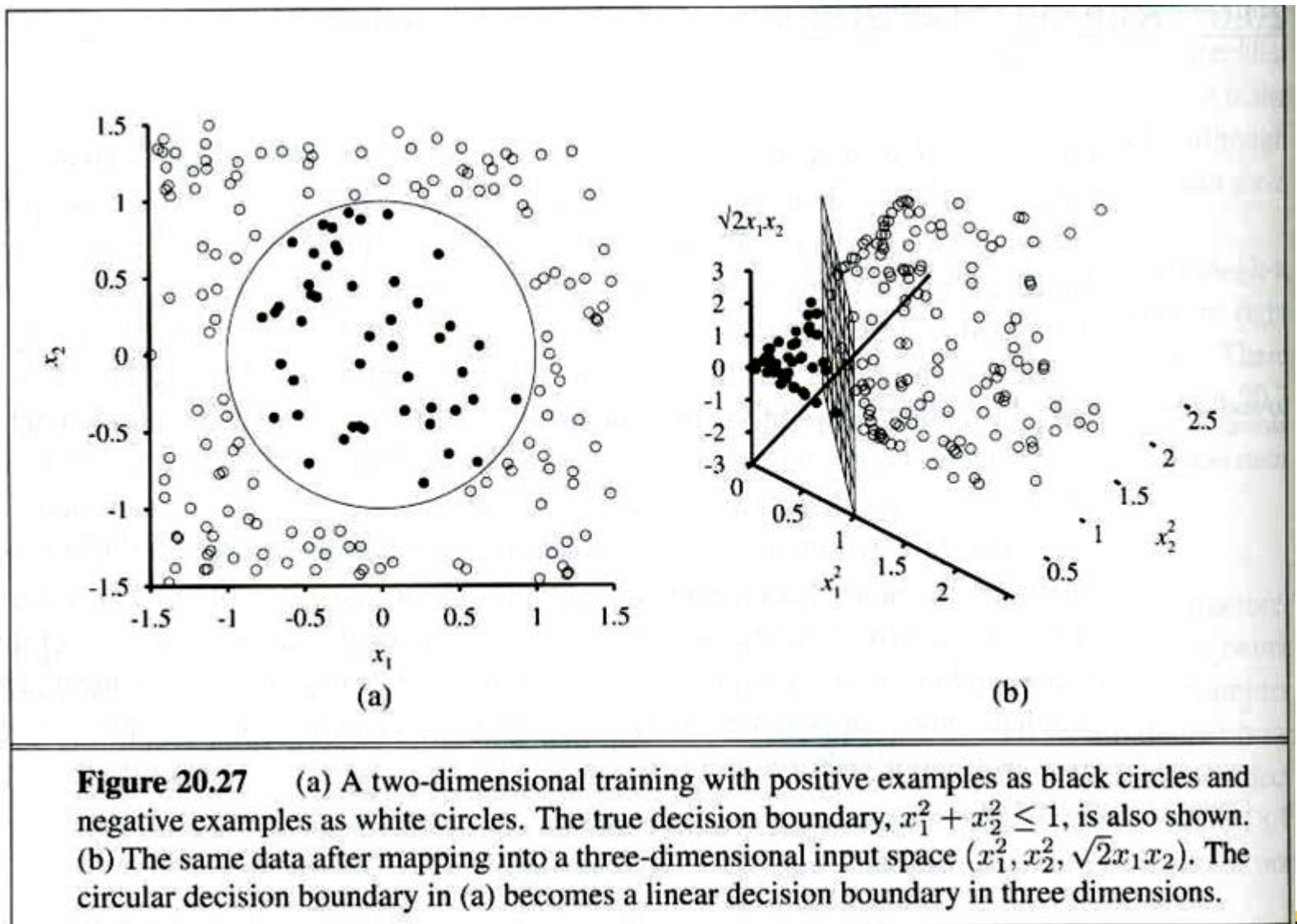
where $\sigma > 0$

Sigmoid

$$K(\vec{x}, \vec{y}) = \tanh(\kappa \langle \vec{x} \cdot \vec{y} \rangle + \mathcal{G})$$

where $\kappa > 0$ and $\mathcal{G} < 0$

Representasi Data & Kernel



Representasi Data & Kernel

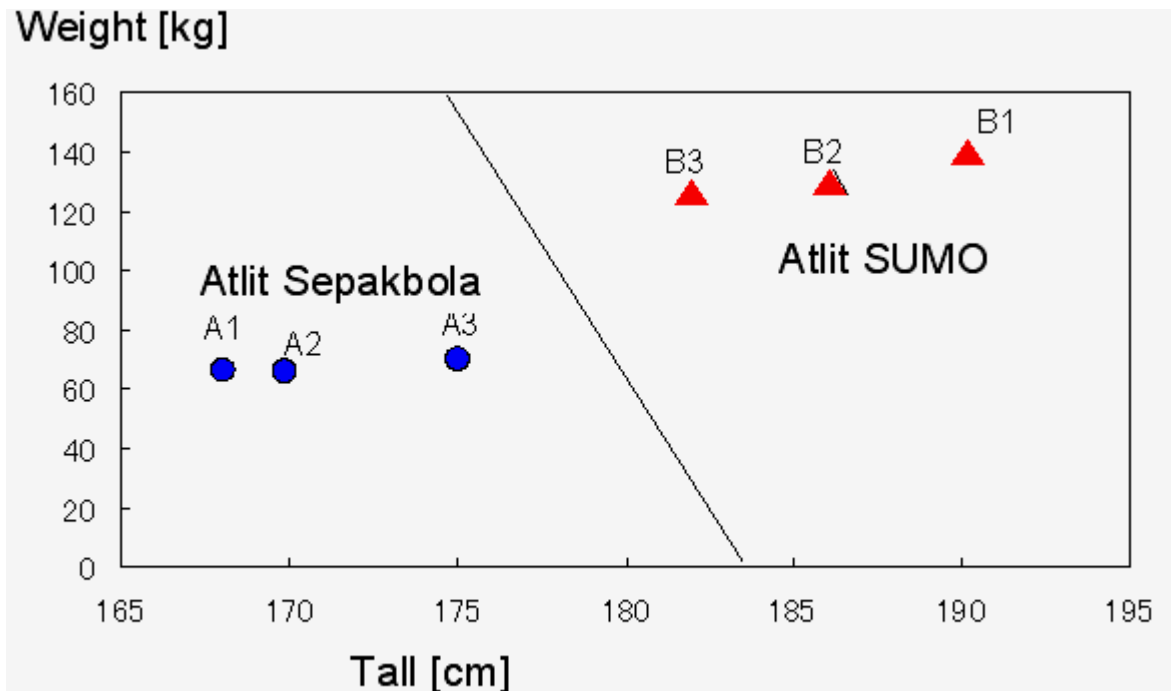
$$(x_1, x_2) \mapsto (x_1^2, x_2^2, \sqrt{2}x_1x_2)$$

$$\begin{aligned} \langle \phi(\vec{x}) \cdot \phi(\vec{y}) \rangle &= (x_1^2, x_2^2, \sqrt{2}x_1x_2)(y_1^2, y_2^2, \sqrt{2}y_1y_2)^T \\ &= ((x_1, x_2)(y_1, y_2)^T)^2 \\ &= (\langle \vec{x} \cdot \vec{y} \rangle)^2 \\ &=: K(\vec{x}, \vec{y}) \end{aligned}$$

Representasi Data & Kernel

- Umumnya data direpresentasikan secara individual. Misalnya, untuk membedakan atlet Sumo dan atlet sepakbola, bisa dengan mengukur berat badan dan tinggi mereka

67 kg
167 cm



Representasi Data & Kernel

- Metode Kernel : data tidak direpresentasikan secara individual, melainkan lewat perbandingan antara sepasang data

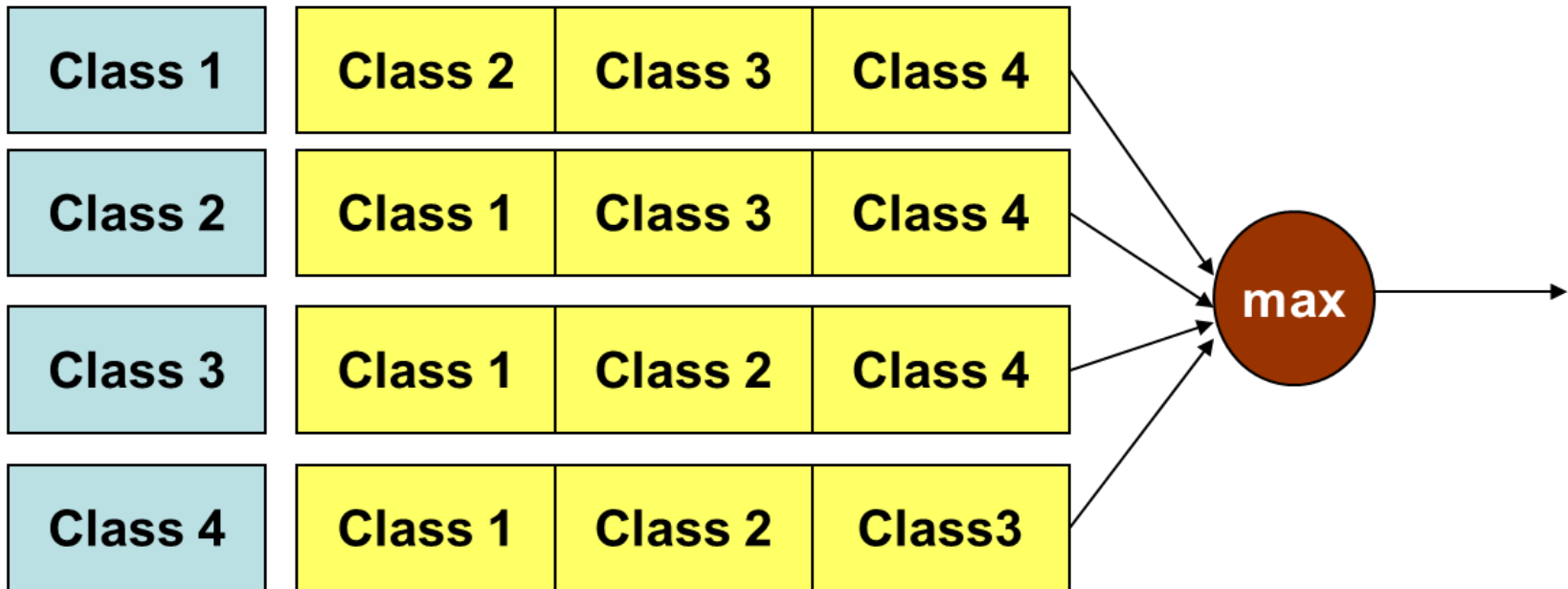
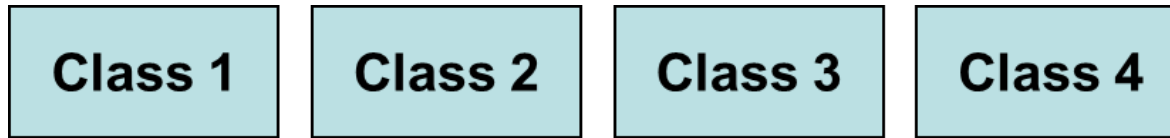
	A1	A2	A3	B1	B2	B3
A1	$K(A1,A1)$	$K(A1,A2)$	$K(A1,A3)$	$K(A1,B1)$	$K(A1,B2)$	$K(A1,B3)$
A2	$K(A2,A1)$	$K(A2,A2)$	$K(A2,A3)$	$K(A2,B1)$	$K(A2,B2)$	$K(A2,B3)$
A3	$K(A3,A1)$	$K(A3,A2)$	$K(A3,A3)$	$K(A3,B1)$	$K(A3,B2)$	$K(A3,B3)$
B1	$K(B1,A1)$	$K(B1,A2)$	$K(B1,A3)$	$K(B1,B1)$	$K(B1,B2)$	$K(B1,B3)$
B2	$K(B2,A1)$	$K(B2,A2)$	$K(B2,A3)$	$K(B2,B1)$	$K(B2,B2)$	$K(B2,B3)$
B3	$K(B3,A1)$	$K(B3,A2)$	$K(B3,A3)$	$K(B3,B1)$	$K(B3,B2)$	$K(B3,B3)$

Pemakaian SVM pada Multiclass Problem

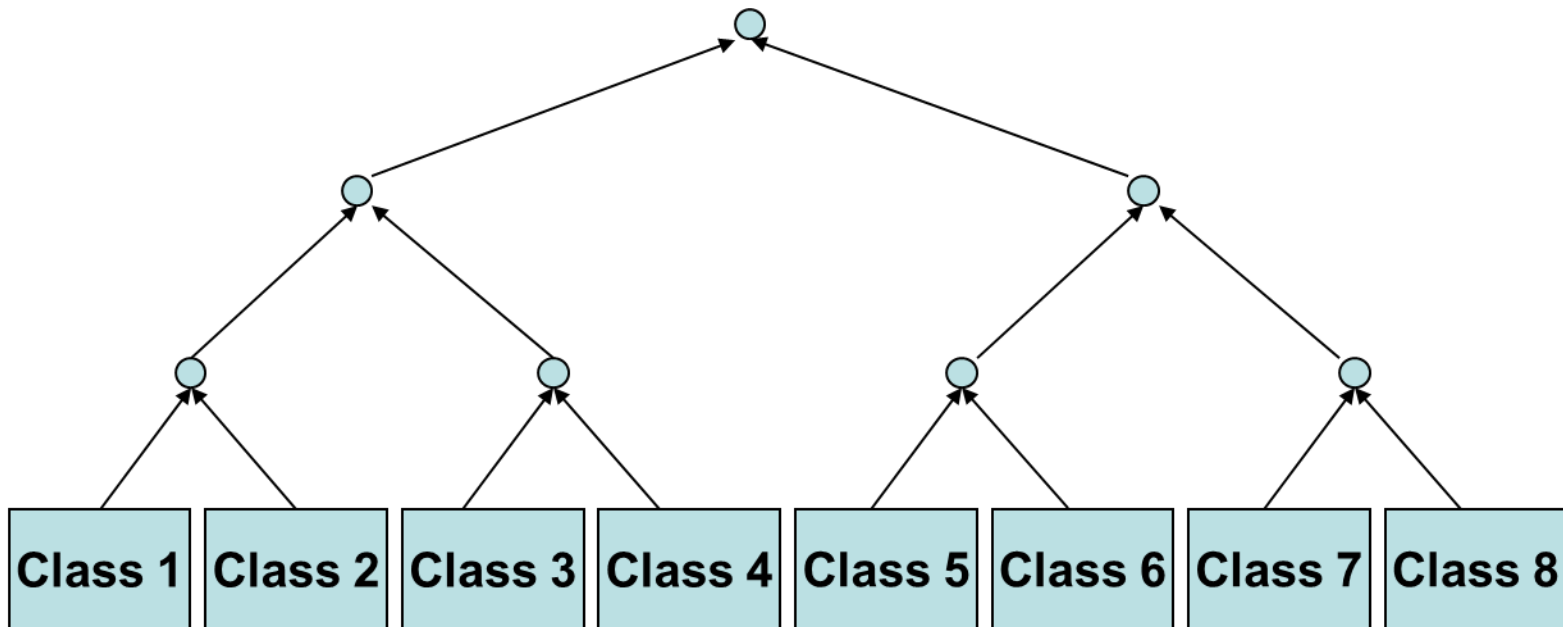
Multiclass Problems

- Pada prinsipnya SVM adalah binary classifier
- Expansion to multiclass classifier:
 1. One vs Others Approach
 2. One vs One : tree structured approach
 1. Bottom-up tree (Pairwise)
 2. Top-down tree (Decision Directed Acyclic Graph)
- Dari sisi training effort : One to Others lebih baik daripada One vs One
- Runtime : keduanya memerlukan evaluasi q SVMs (q = num. of classes)

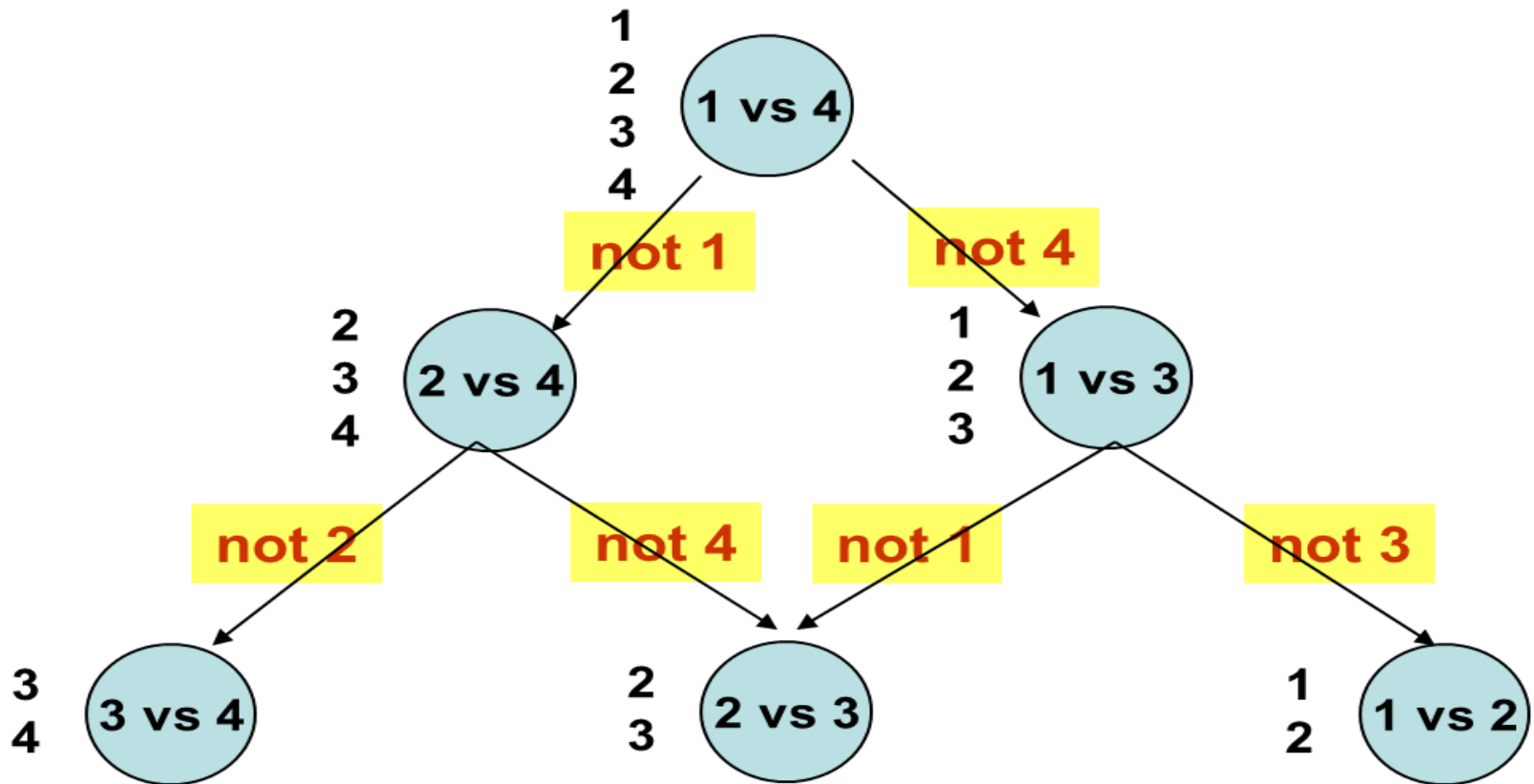
One vs Others



Bottom-Up Tree



Top-Down Tree (DDAG)



Experiment : Digit Recognition

- Num. of class : 10
- Num. of samples
 - Training Set : 100 samples/class
 - Test Set : 100 samples/class
- Num. of attributes (Dimension) : 64
- Feature Extraction : Mesh 8x8
- Database source : SANYO Handwriting Numeral Database (we used only printed-font characters)

Part of patterns in training set

0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9

Part of patterns in test set

0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9

SVM Experimental Results

- SVM Parameters :
 - $\gamma=0.01$ $\lambda:3.0$ $C:1.0$
 - Vijayakumar Algorithm max iteration : 100
 - Gaussian Kernel with $\sigma=0.5$
- Recognition rate :
 - Training Set : 100%
 - Test set : 100%

KARAKTERISTIK SVM

Karakteristik SVM sebagaimana telah dijelaskan sebelumnya, dirangkumkan sebagai berikut:

1. SVM adalah suatu teknik untuk melakukan prediksi, baik dalam kasus klasifikasi maupun regresi. SVM berada dalam satu kelas dengan Artificial Neural Network (ANN) dalam hal fungsi dan kondisi permasalahan yang bisa diselesaikan. Keduanya masuk dalam kelas *supervised learning*.
2. Secara prinsip SVM adalah linear classifier

3. Pattern recognition dilakukan dengan mentransformasikan data pada input space ke ruang yang berdimensi lebih tinggi, dan optimisasi dilakukan pada ruang vector yang baru tersebut. Hal ini membedakan SVM dari solusi pattern recognition pada umumnya, yang melakukan optimisasi parameter pada ruang hasil transformasi yang berdimensi lebih rendah daripada dimensi input space.
4. Menerapkan strategi *Structural Risk Minimization (SRM)*
5. Prinsip kerja SVM pada dasarnya hanya mampu menangani klasifikasi dua class.

KELEBIHAN DAN KEKURANGAN SVM

Kelebihan SVM antara lain sbb:

1. *Generalisasi*

kemampuan suatu metode (SVM, neural network, dsb.) untuk mengklasifikasikan suatu pattern, yang tidak termasuk data yang dipakai dalam fase pembelajaran metode itu.

Vapnik menjelaskan bahwa generalization error dipengaruhi oleh dua faktor: error terhadap training set, dan error dipengaruhi oleh dimensi VC (Vapnik-Chervokinensis).

Strategi pembelajaran pada neural network dan umumnya metode learning machine difokuskan pada usaha untuk meminimalkan error pada training-set. Strategi ini disebut *Empirical Risk Minimization (ERM)*.

Adapun SVM selain meminimalkan error pada training-set, juga meminimalkan faktor kedua. Strategi ini disebut *Structural Risk Minimization (SRM)*, dan dalam SVM diwujudkan dengan memilih hyperplane dengan margin terbesar. Pendekatan SRM pada SVM memberikan error generalisasi yang lebih kecil daripada yang diperoleh dari strategi ERM pada neural network maupun metode yang lain.

2. *Curse of dimensionality*

Curse of dimensionality didefinisikan sebagai masalah yang dihadapi suatu metode pattern recognition dalam mengestimasi parameter (misalnya jumlah hidden neuron pada neural network, stopping criteria dalam proses pembelajaran dsb.) dikarenakan jumlah sampel data yang relatif sedikit dibandingkan dimensional ruang vektor data tersebut.

Semakin tinggi dimensi dari ruang vektor informasi yang diolah, membawa konsekuensi dibutuhkannya jumlah data dalam proses pembelajaran. Tingkat generalisasi yang diperoleh oleh SVM tidak dipengaruhi oleh dimensi dari input vector. SVM merupakan salah satu metode yang tepat dipakai untuk memecahkan masalah berdimensi tinggi, dalam keterbatasan sampel data yang ada.

3. *Feasibility*

SVM dapat diimplementasikan relatif mudah, karena proses penentuan support vector dapat dirumuskan dalam QP problem. Dengan demikian jika memiliki *library* untuk menyelesaikan QP problem, dengan sendirinya SVM dapat diimplementasikan dengan mudah. Selain itu dapat diselesaikan dengan metode sekuensial sebagaimana penjelasan sebelumnya.

SVM memiliki kelemahan atau keterbatasan, antara lain:

1. Sulit dipakai dalam problem berskala besar. Skala besar dalam hal ini dimaksudkan dengan jumlah sample yang diolah.
2. SVM secara teoritik dikembangkan untuk problem klasifikasi dengan dua class. Dewasa ini SVM telah dimodifikasi agar dapat menyelesaikan masalah dengan class lebih dari dua, antara lain strategi One versus rest dan strategi Tree Structure.