

Pengantar Support Vector Machine*

Anto Satriyo Nugroho[†]

February 8, 2007

1 Pengantar

Pattern Recognition (PR) didefinisikan sebagai proses pemetaan suatu data ke dalam konsep tertentu yang telah didefinisikan sebelumnya[1]. Konsep tertentu ini disebut *class* atau *category*. Contoh sederhana dari Pattern Recognition misalnya dalam problem klasifikasi yang melibatkan dua buah konsep/class: "Atlet Sumo" dan "Pemain Sepak Bola". Seorang atlet Sumo (Gulat Jepang) dapat dibedakan dengan seorang pemain Sepak Bola dengan mengukur tinggi badan dan beratnya. Umumnya seorang atlit Sumo memiliki tinggi badan dan berat yang lebih besar daripada seorang pemain sepak bola. Tinggi dan berat badan ini adalah dua buah "feature" yang menentukan apakah seorang atlit diidentifikasi/dipetakan ke konsep "Atlet Sumo" atautkah "Pemain Sepak Bola".

Aplikasi pattern recognition sangat luas, di antaranya mengenali suara dalam sistem sekuriti, membaca huruf dalam OCR, mengklasifikasikan penyakit secara otomatis berdasarkan hasil diagnosa kondisi medis pasien dan sebagainya. Berbagai metode dikenal dalam pattern recognition, seperti linear discrimination analysis, hidden markov model hingga metode yang dikembangkan atas motivasi meniru proses informasi pada manusia, misalnya seperti artificial neural network. Salah satu metode PR yang akhir-akhir ini banyak mendapat perhatian adalah Support Vector Machine (SVM)[2]. Hal ini tercermin dari hasil studi yang dilakukan oleh ICDM pada tahun 2006, mengenai top 10 algoritma dalam Data Mining. Studi yang dilakukan oleh Xindong Wu dan Vipin Kumar ini mengidentifikasi SVM pada peringkat ke-3, setelah C4.5 dan k-Nearest Neighbor Classifier.

Support Vector Machine (SVM) dikembangkan oleh Boser, Guyon, Vapnik, dan pertama kali dipresentasikan pada tahun 1992 di Annual Workshop on Computational Learning Theory. Konsep dasar SVM sebenarnya merupakan kombinasi harmonis dari teori-teori komputasi yang telah ada puluhan tahun sebelumnya, seperti margin hyperplane (Duda & Hart tahun 1973, Cover tahun 1965, Vapnik 1964, dsb.), kernel diperkenalkan oleh Aronszajn tahun 1950, dan demikian juga dengan konsep-konsep pendukung yang lain. Akan tetapi hingga tahun 1992, belum pernah ada upaya merangkaikan komponen-komponen tersebut[4][5].

*disampaikan pada e-tutorial SVM di milis indo-dm@yahoogroups.com 5-18 Februari 2007

[†]Penulis adalah peneliti di BPP Teknologi, dan bekerja sebagai Visiting Professor di Chukyo University Japan, tahun 2003-2007, URL: <http://asnugroho.net>

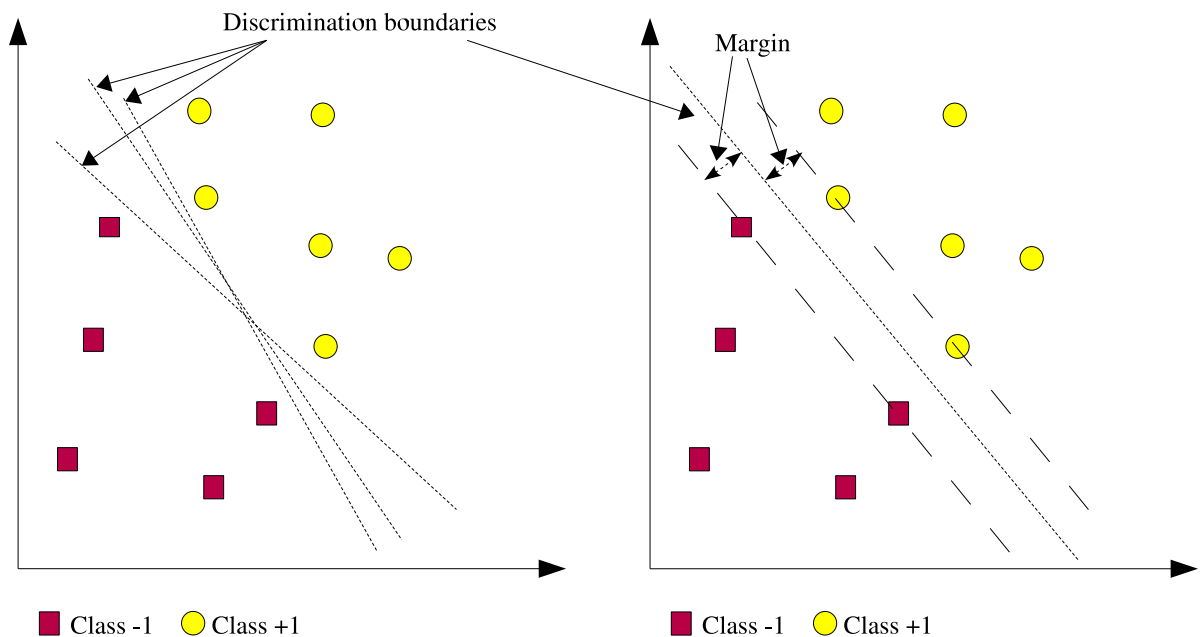


Figure 1: SVM berusaha menemukan hyperplane terbaik yang memisahkan kedua class -1 dan $+1$

Berbeda dengan strategi neural network yang berusaha mencari hyperplane pemisah antar class, SVM berusaha menemukan hyperplane yang terbaik pada input space. Prinsip dasar SVM adalah linear classifier, dan selanjutnya dikembangkan agar dapat bekerja pada problem non-linear. dengan memasukkan konsep kernel trick pada ruang kerja berdimensi tinggi.

Pada e-tutorial ini akan dibahas tiga pertanyaan sebagai berikut:

1. Apakah SVM itu ?
2. Apa sajakah contoh aplikasi SVM ?
3. Referensi apa sajakah yang direkomendasikan untuk mempelajari SVM ?

Pertanyaan pertama dibahas pada Sec.2 dan Sec.3. Pertanyaan kedua dibahas pada Sec.4, sedangkan pertanyaan ketiga dibahas pada Sec.5, yang akan ditutup dengan kesimpulan.

2 Linear Support Vector Machine

Konsep SVM dapat dijelaskan secara sederhana sebagai usaha mencari hyperplane terbaik yang berfungsi sebagai pemisah dua buah class pada input space. Gambar 1 memperlihatkan beberapa pattern yang merupakan anggota dari dua buah class : $+1$ dan -1 . Pattern yang tergabung pada class -1 disimbolkan dengan warna merah (kotak), sedangkan pattern pada class $+1$, disimbolkan dengan warna kuning(lingkaran).

Problem klasifikasi dapat diterjemahkan dengan usaha menemukan garis (hyperplane) yang memisahkan antara kedua kelompok tersebut. Berbagai alternatif garis pemisah (discrimination boundaries) ditunjukkan pada gambar 1. Hyperplane pemisah terbaik antara kedua class dapat ditemukan dengan mengukur margin hyperplane tsb. dan mencari titik maksimalnya. Margin adalah jarak antara hyperplane tersebut dengan pattern terdekat dari masing-masing class. Pattern yang paling dekat ini disebut sebagai support vector. Garis solid pada gambar 1 sebelah kanan menunjukkan hyperplane yang terbaik, yaitu yang terletak tepat pada tengah-tengah kedua class, sedangkan titik merah dan kuning yang berada dalam lingkaran hitam adalah support vector. Usaha untuk mencari lokasi hyperplane ini merupakan inti dari proses pembelajaran pada SVM.

Tiap data (example) dinotasikan sebagai $\mathbf{x}_i \in \mathcal{R}^D, i = 1, 2, \dots, N$. N adalah banyaknya data. Positive class dinotasikan sebagai $+1$, dan negative class sebagai -1 . Dengan demikian, tiap data dan label class-nya dinotasikan sebagai : $y_i \in \{-1, +1\}$.

Diasumsikan bahwa kedua class tersebut dapat dipisahkan secara sempurna oleh hyperplane di D -dimensional feature space. Hyperplane tersebut didefinisikan sbb.

$$\mathbf{w} \cdot \mathbf{x}_i + b = 0 \quad (1)$$

Data \mathbf{x}_i yang tergolong ke dalam negative class adalah mereka yang memenuhi pertidaksamaan berikut :

$$\mathbf{w} \cdot \mathbf{x}_i + b \leq -1 \quad (2)$$

Adapun data \mathbf{x}_i yang tergolong ke dalam positive class, adalah mereka yang memenuhi pertidaksamaan :

$$\mathbf{w} \cdot \mathbf{x}_i + b \geq 1 \quad (3)$$

Optimal margin dihitung dengan memaksimalkan jarak antara hyperplane dan pattern terdekat. Jarak ini dirumuskan sebagai $1/\|\mathbf{w}\|$ ($\|\mathbf{w}\|$ adalah norm dari weight vector \mathbf{w}).

Selanjutnya, masalah ini diformulasikan ke dalam Quadratic Programming (QP) problem, dengan meminimalkan Eq.4 dibawah constraint Eq.5.

Minimize:

$$\|\mathbf{w}\|^2 \quad (4)$$

Subject to:

$$y_i(\mathbf{w} \cdot \mathbf{x}_i + b) - 1 \geq 0, \forall i \quad (5)$$

Optimisasi ini dapat diselesaikan dengan Lagrange Multipliers:

$$L(\mathbf{w}, b, \alpha) = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^N \alpha_i y_i (\mathbf{w} \cdot \mathbf{x}_i + b - 1) \quad (6)$$

α_i adalah Lagrange multiplier yang berkorespondensi dengan \mathbf{x}_i . Nilai α_i adalah nol atau positif. Untuk menyelesaikan masalah tsb. pertama-tama meminimalkan L terhadap

\mathbf{w} , dan memaksimalkan L terhadap α_i . Dengan memodifikasi persamaan 6, maximization problem di atas dapat direpresentasikan dalam α_i , sbb.

Maximize:

$$\sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i,j=1}^N \alpha_i \alpha_j y_i y_j \mathbf{x}_i \mathbf{x}_j \quad (7)$$

Subject to:

$$\alpha_i \geq 0, \sum_{i=1}^N \alpha_i y_i = 0 \quad (8)$$

Solusi dari problem ini akan menghasilkan banyak α_i dengan nilai nol. Data yang berkorespondensi dengan α_i yang tidak nol, merupakan support vectors, yaitu data yang memiliki jarak terdekat dengan hyperplane.

2.1 Soft Margin

Penjelasan di atas berdasarkan asumsi bahwa kedua belah class dapat terpisah secara sempurna oleh hyperplane. Akan tetapi, pada umumnya kedua buah class tersebut tidak dapat terpisah secara sempurna. Hal ini menyebabkan proses optimisasi tidak dapat diselesaikan, karena tidak ada \mathbf{w} dan b yang memenuhi pertidaksamaan 5.

Untuk itu, pertidaksamaan 5 dimodifikasi dengan memasukkan slack variable ξ_i ($\xi_i \geq 0$), menjadi

$$y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1 - \xi_i, \forall i \quad (9)$$

Demikian juga dengan 4, sehingga diperoleh :

Minimize

$$\frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N \xi_i \quad (10)$$

Parameter C bertugas mengontrol tradeoff antara margin and classification error. Semakin besar nilai C , semakin besar penalty yang dikenakan untuk tiap classification error.

3 Non Linear Support Vector Machine

Sebagaimana dijelaskan pada bab sebelumnya, SVM merupakan satu varian dari linear machine sehingga hanya dapat dipakai untuk menyelesaikan masalah yang sifatnya linearly separable. Untuk dapat dipakai dalam kasus non-linear, pertama-tama data yang berada pada ruang vektor awal ($\{\mathbf{x}_i \in \mathbb{R}^D\}$) berdimensi D , harus dipetakan ke ruang vektor baru yang berdimensi lebih tinggi ($\{\mathbf{x}'_i \in \mathbb{R}^Q\}$). Misalnya saja fungsi pemetaan tersebut dinotasikan sebagai $\Phi(\mathbf{x})$. Pemetaan ini bertujuan untuk merepresentasikan data ke dalam format yang linear separable pada ruang vektor yang baru. Ilustrasi proses ini ditunjukkan pada gambar 2.

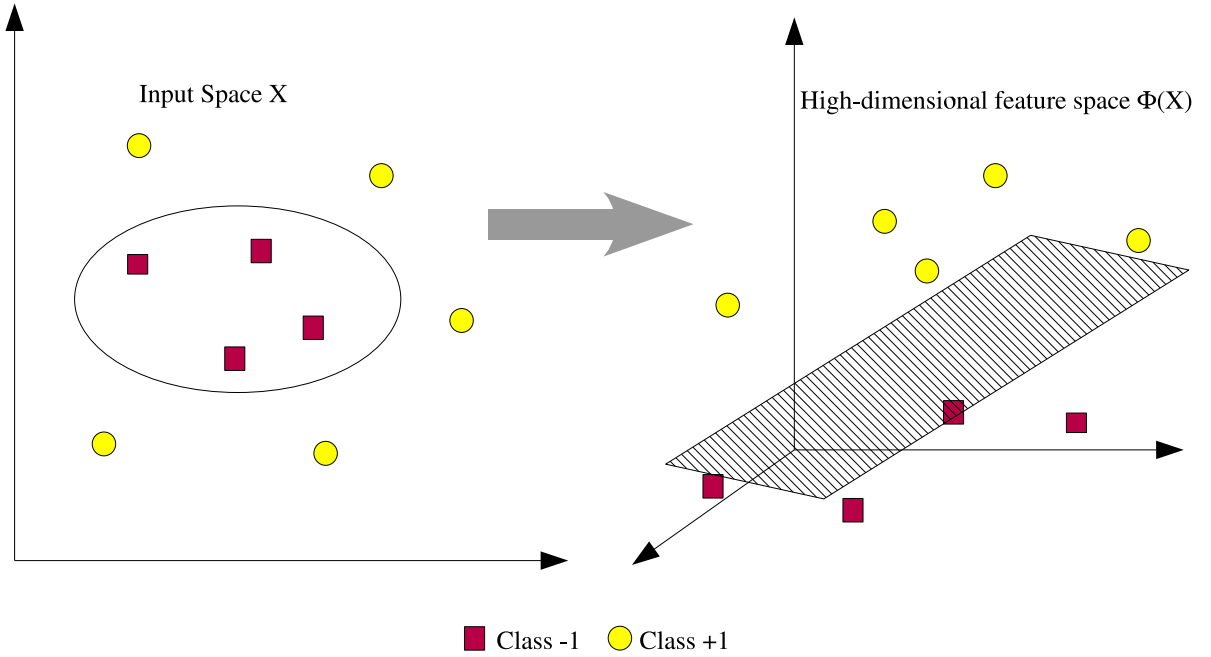


Figure 2: Fungsi Φ memetakan data ke ruang vektor yang berdimensi lebih tinggi, sehingga kedua buah class dapat dipisahkan secara linear oleh sebuah hyperplane

$$\Phi : \mathbb{R}^D \rightarrow \mathbb{R}^Q \quad D < Q \quad (11)$$

Selanjutnya dilakukan proses training sama sebagaimana pada Linear SVM. Proses optimisasi pada fase ini memerlukan perhitungan dot product dua buah example pada ruang vektor baru. Dot product kedua buah vektor (\mathbf{x}_i) dan (\mathbf{x}_j) dinotasikan sebagai $\Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j)$. Nilai dot product kedua buah vektor ini dapat dihitung secara tak langsung, yaitu tanpa mengetahui fungsi transformasi Φ . Teknik komputasi ini disebut *Kernel Trick*, yaitu menghitung dot product dua buah vector di ruang vektor baru dengan memakai komponen kedua buah vektor tersebut di ruang vektor asal sbb.

$$K(\mathbf{x}_i, \mathbf{x}_j) = \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j) \quad (12)$$

Berbagai jenis fungsi dapat dipakai sebagai kernel K , sebagaimana tercantum pada Tabel1.

Selanjutnya klasifikasi non linear pada SVM terhadap test sample \mathbf{x} dirumuskan sbb.

$$f(\Phi(\mathbf{x})) = \mathbf{w} \cdot \Phi(\mathbf{x}) + b \quad (13)$$

Table 1: Fungsi kernel yang sering dipakai dalam SVM

| Nama Kernel | Definisi |
|-------------|--|
| Polynomial | $K(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i \cdot \mathbf{x}_j + 1)^p$ |
| Gaussian | $K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\frac{\ \mathbf{x}_i - \mathbf{x}_j\ ^2}{2\sigma^2})$ |
| Sigmoid | $K(\mathbf{x}_i, \mathbf{x}_j) = \tanh(\alpha \mathbf{x}_i \cdot \mathbf{x}_j + \beta)$ |

$$= \sum_{i=1, x_i \in SV}^N \alpha_i y_i \Phi(\mathbf{x}) \cdot \Phi(\mathbf{x}_i) + b \quad (14)$$

$$= \sum_{i=1, x_i \in SV}^N \alpha_i y_i K(\mathbf{x}, \mathbf{x}_i) + b \quad (15)$$

Support vector SV adalah subset dari data training set, dengan α yang tidak negatif $\{\mathbf{x}_i | \alpha_i \neq 0\}$ ($i = 1, 2, \dots, l$).

References

- [1] J. Toriwaki, Ninshiki Kougaku, Korona-sha, 1993
- [2] H. Byun, S.W. Lee, "A Survey on Pattern Recognition Applications of Support Vector Machines", International Journal of Pattern Recognition and Artificial Intelligence, Vol.17, No.3, pp.459–486, 2003
- [3] <http://www.cs.uvm.edu/icdm/algorithms/index.shtml>
- [4] N. Cristianini, J.S. Taylor, "An Introduction to Support Vector Machine and Other Kernel-Based Learning Methods", Cambridge Press University, 2000.
- [5] T. Hastie et al., and P. W. Daly, *The Elements of Statistical Learning : Datamining, Interference, and Prediction*, Springer Verlag, 2001.
- [6] K. Tsuda, "Overview of Support Vector Machine," Journal of IEICE, Vol.83, No.6, pp.460-466, 2000
- [7] V.N. Vapnik, *The Nature of Statistical Learning Theory*, Springer-Verlag , New York Berlin Heidelberg, 1999 (2nd ed.)