Support Vector Machines

Ali Akbar Septiandri

Universitas Al-Azhar Indonesia

aliakbars@live.com

December 27, 2017

Selayang Pandang

Bahan Bacaan

1. VanderPlas, J. (2016). Python Data Science Handbook. (In Depth: Support Vector Machines) https://jakevdp.github.io/PythonDataScienceHandbook/05.07-support-vector-machines.html
2. Witten, I. H., Frank, E., Hall, M. A., & Pal, C. J. (2016). Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann. (Section 6.3)

Pendahuluan

Apa itu SVM?

- Support vector machines (SVM) adalah salah satu yang paling sering digunakan dan cukup efektif
- Merupakan kombinasi dari dua ide
  - Klasifikasi margin maksimum
  - "Kernel trick"
- SVMs adalah *linear classifier*, seperti regresi logistik

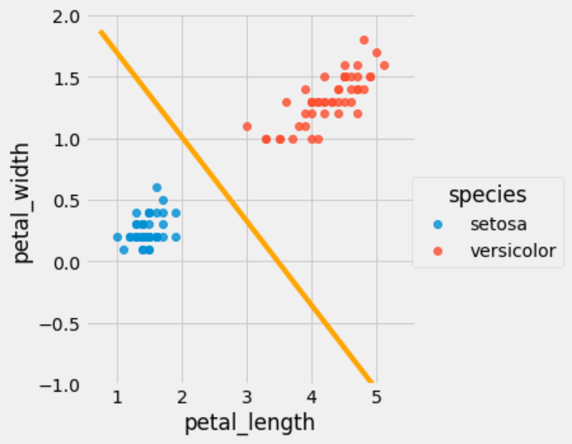
$\mathbf{w}^T\mathbf{x}$ adalah panjang dari proyeksi $\mathbf{x}$ ke $\mathbf{w}$
(jika $\mathbf{w}$ adalah vektor unit), i.e. $b = \mathbf{w}^T\mathbf{x}$.

Hyperplane

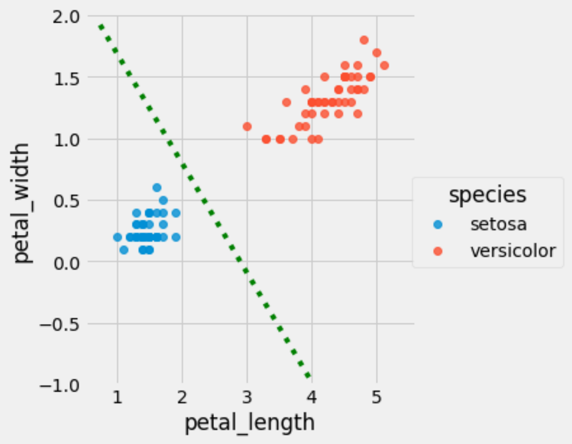
Untuk setiap *linear classifier*,

- Data latih $(\mathbf{x}_i, y_i), i = 1, ..., n.y_i \in \{-1, +1\}$
- Hyperplane $\mathbf{w}^T\mathbf{x} + w_0 = 0$
- Dalam kasus SVM, kita akan menggunakan -1 alih-alih 0 karena lebih memudahkan matematikanya

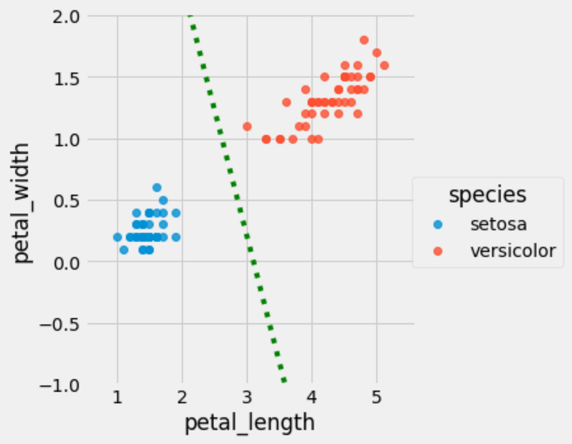
Batas Keputusan



Batas Keputusan



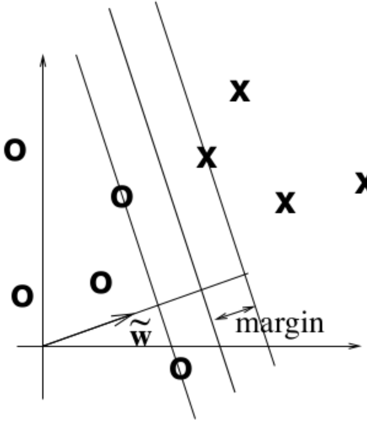
Batas Keputusan



Batas Keputusan dan Margin Maksimum

Ide: Maksimalkan Marginnya

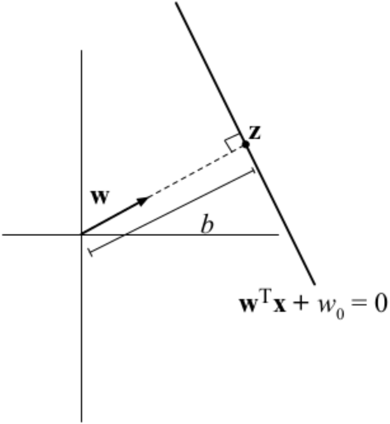
Margin adalah jarak antara batas keputusan dengan data latih terdekat



Menghitung Margin

- Dilakukan dengan menghitung jarak terdekat dari titik pusat O(0,0) ke batas keputusan terlebih dahulu
- Memanfaatkan fakta bahwa **w** tegak lurus dengan batas keputusan

Menghitung Jarak ke Hyperplane



Diberikan...

- $\mathbf{w}^T\mathbf{x} + w_0 = 0$

Diberikan...

- $\mathbf{w}^T\mathbf{x} + w_0 = 0$
- panjang $\|z\| = b$ sehingga $b\frac{\mathbf{w}}{\|\mathbf{w}\|} = \mathbf{z}$

Diberikan...

- $\mathbf{w}^T\mathbf{x} + w_0 = 0$
- panjang $\|z\| = b$ sehingga $b\frac{\mathbf{w}}{\|\mathbf{w}\|} = \mathbf{z}$
- berarti $\mathbf{w}^T\mathbf{z} + w_0 = 0$

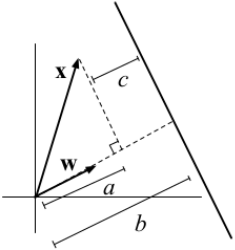
Diberikan...

- $\mathbf{w}^T\mathbf{x} + w_0 = 0$
- panjang $\|z\| = b$ sehingga $b\frac{\mathbf{w}}{\|\mathbf{w}\|} = \mathbf{z}$
- berarti $\mathbf{w}^T\mathbf{z} + w_0 = 0$
- substitusi nilai $\mathbf{z}$, kita akan mendapatkan

$$\begin{aligned} \mathbf{w}^T\frac{b\mathbf{w}}{\|\mathbf{w}\|} + w_0 &= 0 \\ \frac{b\mathbf{w}^T\mathbf{w}}{\|\mathbf{w}\|} + w_0 &= 0 \\ b &= -\frac{w_0}{\|\mathbf{w}\|} \end{aligned}$$

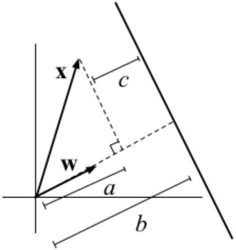
- ingat bahwa $\|\mathbf{w}\| = \sqrt{\mathbf{w}^T\mathbf{w}}$

Menghitung Jarak ke Hyperplane



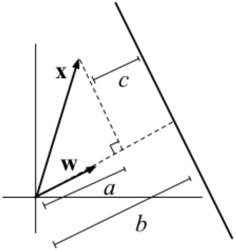
- Kita ingin c, jarak dari $\mathbf{x}$ ke hyperplane

Menghitung Jarak ke Hyperplane



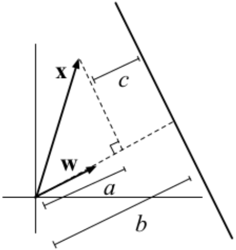
- Kita ingin c, jarak dari $\mathbf{x}$ ke hyperplane
- Kita tahu $c = |b - a|$, dengan a adalah panjang proyeksi $\mathbf{x}$ ke $\mathbf{w}$

Menghitung Jarak ke Hyperplane



- Kita ingin c, jarak dari $\mathbf{x}$ ke hyperplane
- Kita tahu $c = |b - a|$, dengan a adalah panjang proyeksi $\mathbf{x}$ ke $\mathbf{w}$
- Berapa nilai a?

Menghitung Jarak ke Hyperplane



- Kita ingin c, jarak dari $\mathbf{x}$ ke hyperplane
- Kita tahu $c = |b - a|$, dengan a adalah panjang proyeksi $\mathbf{x}$ ke $\mathbf{w}$
- Berapa nilai a?
- $a = \frac{\mathbf{w}^T \mathbf{x}}{\|\mathbf{w}\|}$

Persamaan untuk Margin

- Jarak tegak lurus dari titik $\mathbf{x}$ ke hyperplane $\mathbf{w}^T\mathbf{x} + w_0 = 0$ adalah
$$\frac{1}{\|\mathbf{w}\|}|\mathbf{w}^T\mathbf{x} + w_0|$$
- Margin adalah jarak dari titik terdekat ke hyperplane

$$\min_i \frac{1}{\|\mathbf{w}\|}|\mathbf{w}^T\mathbf{x}_i + w_0|$$

Max Margin Optimization Problem

Tujuan kita adalah memaksimalkan margin dengan

$$\begin{aligned} &\max_{\mathbf{w}} \frac{1}{\|\mathbf{w}\|} \\ \text{subject to } & \mathbf{w}^T\mathbf{x}_i + w_0 \geq 0, && \text{for all } i \text{ with } y_i = 1 \\ & \mathbf{w}^T\mathbf{x}_i + w_0 \leq 0, && \text{for all } i \text{ with } y_i = -1 \\ & \min_i |\mathbf{w}^T\mathbf{x}_i + w_0| = 1 \end{aligned}$$

Max Margin Optimization Problem

Batasan pertama dan kedua dapat diubah menjadi

$$
\begin{aligned}
&\max_{\mathbf{w}} \ \frac{1}{\|\mathbf{w}\|} \\
\text{subject to } & \mathbf{w}^T\mathbf{x}_i + w_0 \geq +1, && \text{for all } i \text{ with } y_i = 1 \\
& \mathbf{w}^T\mathbf{x}_i + w_0 \leq -1, && \text{for all } i \text{ with } y_i = -1 \\
& \min_i \left|\mathbf{w}^T\mathbf{x}_i + w_0\right| = 1
\end{aligned}
$$

Max Margin Optimization Problem

Batasan ketiga dapat dibuang karena redundan

$$\max_{\mathbf{w}} \frac{1}{\|\mathbf{w}\|}$$

$$\begin{aligned}
\text{subject to } \mathbf{w}^T\mathbf{x}_i + w_0 &\geq +1, && \text{for all } i \text{ with } y_i = 1 \\
\mathbf{w}^T\mathbf{x}_i + w_0 &\leq -1, && \text{for all } i \text{ with } y_i = -1
\end{aligned}$$

Max Margin Optimization Problem

Dapat disederhanakan lagi menjadi

$$\max_{\mathbf{w}} \frac{1}{\|\mathbf{w}\|}$$
$$\text{subject to } y_i(\mathbf{w}^T\mathbf{x}_i + w_0) \geq +1, \qquad \text{for all } i$$

Perhatikan bahwa memaksimalkan nilai $\frac{1}{\|\mathbf{w}\|}$ sama dengan *meminimalkan* $\|\mathbf{w}\|^2$

Optimasi SVM

Jadi, optimasi vektor bobot pada SVM dapat didefinisikan sebagai

$$\min_{\mathbf{w}} \ \|\mathbf{w}\|^2$$
$$\text{subject to } y_i(\mathbf{w}^T\mathbf{x}_i + w_0) \geq +1 \qquad , \text{for all } i$$

Ini dikenal sebagai *contrained optimization problem*.

Mencari Nilai Optimal

- Persoalan tadi dapat diselesaikan dengan Lagrange multipliers sehingga parameter optimalnya adalah

$$\mathbf{w} = \sum_i \alpha_i y_i \mathbf{x}_i$$

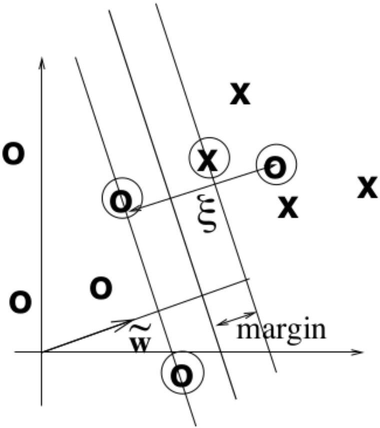
- Perhatikan bahwa hyperplane yang optimal hanya dibentuk dari beberapa titik saja, yang disebut sebagai support vectors
- $\alpha_i = 0$ untuk *non-support*
- Tidak ada minimum lokal

Prediksi

Prediksi pada data baru $\mathbf{x}$ adalah

$$
\begin{aligned}
f(\mathbf{x}) &= sign(\mathbf{w}^T\mathbf{x} + w_0) \\
&= sign(\sum_{i=1}^{n} \alpha_i y_i (\mathbf{x}^T\mathbf{x}) + w_0)
\end{aligned}
$$

Apa yang terjadi kalau kasusnya seperti ini?



Slack

- Solusi: Gunakan “slack” variabel $\xi_i \geq 0$ untuk tiap data latih
- Idenya seperti memberikan penalti pada *ridge regression*

SVM Non-linear

SVM Non-linear

- Dapat dibuat non-linear dengan fungsi basis seperti pada regresi

SVM Non-linear

- Dapat dibuat non-linear dengan fungsi basis seperti pada regresi
- Ada cara khusus menggunakan kernel

SVM Non-linear

- Dapat dibuat non-linear dengan fungsi basis seperti pada regresi
- Ada cara khusus menggunakan kernel
- Komputasinya lebih cepat

SVM Non-linear

- Dapat dibuat non-linear dengan fungsi basis seperti pada regresi
- Ada cara khusus menggunakan kernel
- Komputasinya lebih cepat
- Topik matematika lanjut, jadi dibahas ide umumnya saja

Transformasi

- Transformasi $\mathbf{x}$ menjadi $\phi(\mathbf{x})$
- Algoritma yang linear hanya bergantung pada $\mathbf{x}^T\mathbf{x}$ sehingga tranformasinya hanya bergantung pada $\phi(\mathbf{x})^T\phi(\mathbf{x})$
- Gunakan fungsi kernel $k(\mathbf{x}_i, \mathbf{x}_j)$ sedemikian sehingga

$$k(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x})^T\phi(\mathbf{x})$$

Support Vector Machine

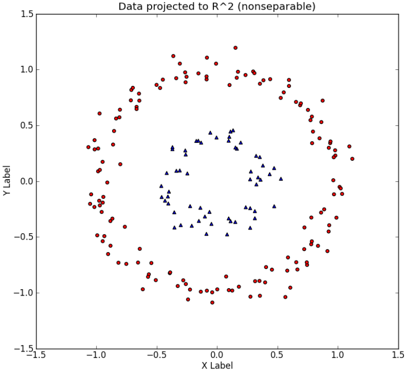
- Support vector machine pada dasarnya adalah *kernelized maximum margin classifier*
- Ingat kembali bahwa prediksi pada SVM adalah

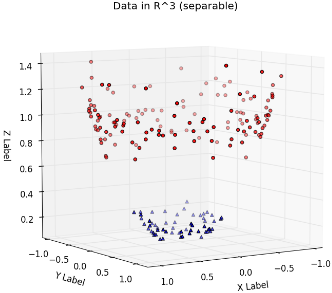
$$\hat{y} = sign(\mathbf{w}^T\mathbf{x} + w_0) = sign(\sum_i \alpha_i y_i \mathbf{x}_i^T \mathbf{x} + w_0)$$

- Dengan kernel, didapat

$$\hat{y} = sign(\sum_i \alpha_i y_i k(\mathbf{x}_i, \mathbf{x}) + w_0)$$

Kernel Trick





Kernel popular lain: RBF

$$k(\mathbf{x}_i, \mathbf{x}_j) = \exp - \frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\alpha^2}$$

Pros & Cons

Pros:

- Modelnya padat, hanya perlu memori yang sedikit
- Fase prediksi bisa sangat cepat
- Bisa bekerja dengan baik dalam dimensi tinggi
- "Kernel tricks"

Cons:

- $O(N^3)$ dalam kasus terburuk, hanya bisa diperbaiki hingga $O(N^2)$
- Sangat bergantung pada parameter ξ
- Tidak probabilistik

Salindia ini dipersiapkan dengan
banyak mengadaptasi dari Nigel Goddard (2014)

Terima kasih