Support Vector Machines

M. Gruber

KW 42

Definition 1 (Lineare Trennbarkeit) *Eine Menge* $\left\{[x^{(1)}, y^{(1)}], \ldots, [x^{(N)}, y^{(N)}]\right\} \subset \mathbf{R}^d \times \{-1, 1\}$ *ist linear trennbar, wenn mindestens ein Wertepaar* $[w, b] \in \mathbf{R}^{(d+1)}$ *existiert mit* $\operatorname{sgn}(w^T x^{(n)} + b) = y^{(n)}$ *für alle* n*. Ein solches Wertepaar* $[w, b]$ *nennt man trennend.*

Satz 1 *Sei* $\left\{[x^{(1)}, y^{(1)}], \ldots, [x^{(N)}, y^{(N)}]\right\} \subset \mathbf{R}^d \times \{-1, 1\}$ *linear trennbar und sei* $[w, b] \in \mathbf{R}^{(d+1)}$ *trennend. Dann hat jeder Punkt* $x^{(n)}$ *von der Trennebene* $E_{[w,b]} = \left\{x \mid w^T x + b = 0\right\}$ *den Abstand*

$$(1/||w||) \cdot y^{(n)}(w^T x^{(n)} + b). \tag{1}$$

Beweis

- Die Gerade $G_w = \{x \mid x = \alpha w,\ \alpha \in \mathbf{R}\}$ ist orthogonal zur Trennebene $E_{[w,b]} = \left\{x \mid w^T x + b = 0\right\}$.
- Der Abstand eines Datenpunktes $x^{(n)}$ von der Trennebene $E_{[w,b]}$ ist gleich dem Abstand seines Projektionsbilds auf der Geraden G_w und dem Punkt, an dem die Gerade G_w die Trennebene trifft.
- Der Projektor $P_w : \mathbf{R}^d \to G_w$ ist $x \mapsto w(w^T w)^{-1} w^T x$ (Lineare Algebra).
- $P_w(x^{(n)}) = w(w^T w)^{-1} w^T x^{(n)} = \alpha_1 w$ mit $\alpha_1 = (w^T w)^{-1} w^T x^{(n)} = w^T x^{(n)} / ||w||^2$.
- $G_w \cap E_{[w,b]} = \{\alpha_2 w\}$ mit $\alpha_2 = -b/(w^T w) = -b/||w||^2$.
- $||\alpha_1 w - \alpha_2 w|| = ||w|| \cdot |\alpha_1 - \alpha_2| = ||w|| \cdot |w^T x^{(n)}/||w||^2 - (-b/||w||^2)| = (1/||w||) \cdot |w^T x^{(n)} + b| = (1/||w||) \cdot y^{(n)}(w^T x^{(n)} + b)$.

□

Bemerkung 1 *Der Abstand (1) ist gleich* $(1/||\beta w||) \cdot y^{(n)}((\beta w)^T x^{(n)} + \beta b)$*, unabhängig von* $\beta > 0$*. Wir können uns o.B.d.A.* $[w, b]$ *so skaliert denken, dass* $\min_{1 \le n \le N} y^{(n)}(w^T x^{(n)} + b) = 1$ *ist.*

Support Vector Machines bestimmen die Trennebene mit dem größten Abstand zu den Punkten $x^{(n)}$ der gegebenen Datenmenge *(fat margins)*.

Betrachten wir verschiedene Versionen dieser Optimierungsaufgabe.

- *Version 1* "Maximiere $1/||w||$ für $[w, b]$ mit $\min_{1 \le n \le N} y^{(n)}(w^T x^{(n)} + b) = 1$".

- *Version 2* "Minimiere $\frac{1}{2}w^T w$ für $[w,b]$ mit $y^{(n)}(w^T x^{(n)} + b) \geq 1$ für alle n".

  Im Unterschied zur Version 1 ist die Nebenbedingung mit $\geq$ formuliert und scheinbar weniger streng. In Wahrheit ist Version 2 aber genauso streng, denn das Gleichheitszeichen wird automatisch für mindestens ein n realisiert (sonst hätte man beim Minimieren von $\frac{1}{2}w^T w$ noch "Luft nach unten".)

- *Version 3 (Lagrange-Formulierung)*

  Für die Größe

$$L(w,b,\alpha) = \tfrac{1}{2}w^T w - \sum_{1\leq n\leq N} \alpha_n (y^{(n)}(w^T x^{(n)} + b) - 1)$$

$$\text{finde} \quad \min_{[w,b]\in\mathbf{R}^{d+1}} \left(\max_{\alpha\in\mathbf{R}^N, \alpha\geq 0} L(w,b,\alpha) \right).$$

  Der Ansatz kommt aus der konvexen Optimierung. Mehr darüber findet man in [2]. Die Lösung $\{w,b,\alpha\}$ hat nach [2] folgende Eigenschaften:

  - $\nabla_{[w,b]} L(w,b,\alpha) = 0$ *(Lagrangian stationarity)*,
  - $y^{(n)}(w^T x^{(n)} + b) - 1 \geq 0$ für alle n *(primal feasibility)*,
  - $\alpha_n (y^{(n)}(w^T x^{(n)} + b) - 1) = 0$ für alle n *(complementary slackness)*,
  - $\alpha_n > 0 \Rightarrow y^{(n)}(w^T x^{(n)} + b) = 1$ *(active constraints, support vectors $x^{(n)}$)*.

  Die Lagrange-Stationarität hat folgende Konsequenzen:

  - $\nabla_w L(w,b,\alpha) = 0 \Rightarrow w = \sum_{1\leq n\leq N} \alpha_n y^{(n)} x^{(n)}$,
  - $\frac{\partial}{\partial b} L(w,b,\alpha) = 0 \Rightarrow \sum_{1\leq n\leq N} \alpha_n y^{(n)} = 0$, d.h. $\alpha^T y = 0$.
  - Die Größe $L(w,b,\alpha)$ vereinfacht sich zur Größe $L(\alpha)$, für die gilt

$$L(\alpha) = -\tfrac{1}{2} \sum_{1\leq m\leq N} \sum_{1\leq n\leq N} y^{(m)} y^{(n)} \alpha_m \alpha_n {x^{(m)}}^T x^{(n)} + \sum_{1\leq n\leq N} \alpha_n$$

    und die Optimierungsaufgabe lautet nun:

$$\text{finde} \quad \max_{\alpha\geq 0, \alpha^T y = 0} L(\alpha).$$

- *Version 4 (QP-Formulierung)*

  Mit der (positiv semidefiniten) Matrix $Q = [y^{(i)} y^{(j)} {x^{(i)}}^T x^{(j)}]_{1\leq i,j\leq N}$

$$\text{finde} \quad \min_{\alpha\geq 0, \alpha^T y = 0} \tfrac{1}{2}\alpha^T Q \alpha + \begin{bmatrix} -1 & \dots & -1 \end{bmatrix} \alpha.$$

  Man wendet einen QP-Löser an...

  Auswertung der Lösung α:

- Für $\alpha_n > 0$ hat $x^{(n)}$ minimalen Abstand zur Trennebene *(support vector)*.
- $w = \sum_{x^{(n)} \text{ support vector}} \alpha_n y^{(n)} x^{(n)}$,
- Wählt man einen *support vector* $x^{(n^*)}$ und löst $y^{(n^*)}(w^T x^{(n^*)} + b) = 1$ nach b auf, erhält man $b = (1/y^{(n^*)}) - w^T x^{(n^*)}$.
- die finale Hypothese ist $u \mapsto \text{sgn}(\sum_{x^{(n)} \text{ support vector}} \alpha_n y^{(n)} (x^{(n)} - x^{(n^*)})^T u + 1/y^{(n^*)})$.

Bemerkung 2 (*kernel*)

- *Der SVM-Lösungsprozess hängt von den Daten nur über einen kernel ab:*

$$Q = [y^{(i)} y^{(j)} k(x^{(i)}, x^{(j)})]_{1 \le i,j \le N} \quad \textit{mit} \quad k(x^{(i)}, x^{(j)}) = \left\langle x^{(i)}, x^{(j)} \right\rangle_{\mathbf{R}^d}.$$

- *Auch die finale Hypothese arbeitet nur mit dem kernel:*

$$u \mapsto \text{sgn}\Big(\sum_{x^{(n)} \textit{ support vector}} \alpha_n y^{(n)} (k(x^{(n)}, u) - k(x^{(n)}, x^{(n^*)})) + (1/y^{(n^*)}) \Big).$$

- *"Kernel-Trick" (einfachste Form): verwende* $\tilde{k}(x, x') = \langle \Phi(x), \Phi(x') \rangle_{\tilde{X} \times \tilde{X}}$ *anstelle von* $k(x, x') = \langle x, x' \rangle_{X \times X}$ *mit* $\Phi : (X, \langle ., . \rangle_{X \times X}) \to (\tilde{X}, \langle ., . \rangle_{\tilde{X} \times \tilde{X}})$

  (Wechsel des feature space).

Bemerkung 3 (Hard vs. soft margins) *Das hier vorgestellte Verfahren geht von linearer Trennbarkeit der Daten aus. Die Datenpunkte werden ohne Ausnahme sauber getrennt (hard margins). Liegt lineare Trennbarkeit nicht vor, versagt das Verfahren.*

Es gibt ein erweitertes Verfahren, das auch bei Nichtvorliegen linearer Trennbarkeit funktioniert. Es lässt zu, dass sich einige Datenpunkte der linearen Trennung entziehen (soft margins). Siehe hierzu [1], Lecture 15, pp.15.

Literatur

[1] Y. S. Abu-Mostafa. Learning from Data. Caltech CS 156.

[2] C. Rudin. MIT 15.097, Prediction: Machine Learning and Statistics, Convex Optimization Overview, 2012.
http://ocw.mit.edu/courses/sloan-school-of-management/15-097-prediction-machine-learning-and-statistics-