

Support Vector Machines

Michael In der Au

8 Juni 2018

Contents

1 Einleitung	2
2 Grundlagen	2
3 SVM Algorithmus	4
3.1 Besonderheiten der SVM	5
4 Nichtlineare Klassifikation	5
4.1 Grundlagen	5
5 Anwendungsbeispiele	6
6 Quellen	7

1 Einleitung

Support Vector Machines (SVM) stellen Algorithmen des überwachten Lernens (supervised learning) dar, mit denen es möglich ist sowohl Regressions- als auch Klassifikationsprobleme zu behandeln. Typischerweise werden allerdings Klassifikationsprobleme mittels SVM bearbeitet.¹

Im Falle einer Klassifikation kann jeder Datenpunkt im n-dimensionalen Raum durch ein Tupel dargestellt werden. Somit ergeben sich N Tupel von Trainingsdaten zu:

$$(x_1, y_1), (x_2, y_2) \dots (x_N, y_N) \text{ mit } x_i \in \mathbb{R}^n \text{ und } y_i \in \{\pm 1\}$$

Hierbei stellt die erste Komponente, x_i die Eingangsdaten und die zweite Komponente, y_i , die Klassen, in die unterschieden werden sollen dar.

Die Trennung der Klassen erfolgt nun durch eine Hyperebene. Hierzu wird eine Funktion gesucht, durch die die Trainingsmenge korrekt klassifiziert wird.

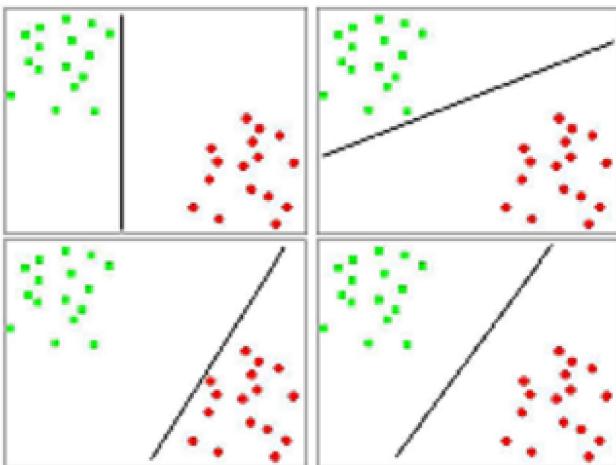
$$f : \mathbb{R}^n \rightarrow \{\pm 1\} \text{ sodass } f(x_i) = y_i$$

Neue Tupel werden somit durch folgende Funktion einer Klasse zugeordnet.

$$f(x_k) = y_k$$

2 Grundlagen

Im simpelsten Anwendungsfall der SVM ist es möglich, die Daten linear zu separieren. Hierzu werden zwei beliebige Klassen von Daten betrachtet, die durch zwei Einflussgrößen beschrieben sind. In diesem einfachen Fall existieren bereits viele Möglichkeiten, die Klassen zu trennen. Grafisch zeigt sich, dass neben einer horizontalen und vertikalen Trennung diverse diagonale Trennungen erfolgen können. Einige mögliche Hyperebenen sind im folgenden dargestellt.

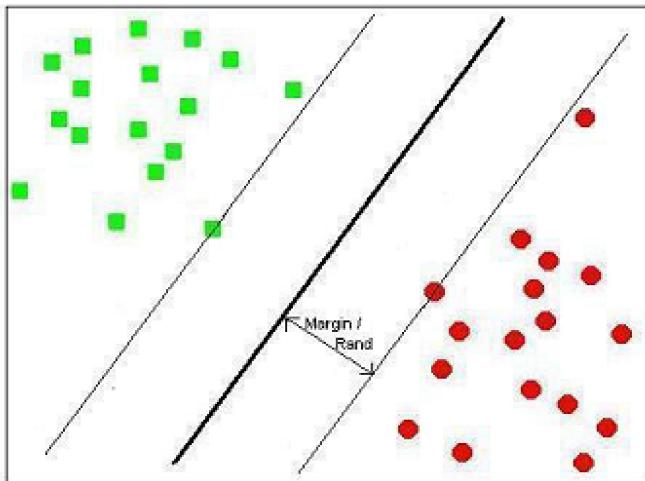


Durch Abbildung 1 wird das Problem deutlich, welche lineare Trennung die optimale ist. Deshalb gilt es nun aus diesen Möglichkeiten jene auszuwählen, welche den breitesten Rand (margin) aufweist. Aus dieser Voraussetzung ergibt sich die Bezeichnung der SVM als “large margin classifier”. In diesem Fall wäre dies die

¹<https://blogs.sas.com/content/subconsciousmusings/files/2017/04/machine-learning-cheat-sheet.png>

Variante unten rechts in Abbildung 2. Neue Punkte können somit mit einer maximalen Wahrscheinlichkeit der korrekten Klasse zugeordnet werden.

Um den größten Abstand der Klassen zueinander zu ermitteln werden nur jene Vektoren aus den Klassen betrachtet, die am nächsten zueinander liegen. Die Bezeichnung als Vektoren bezieht sich darauf, dass die Tupel in diesem zwei-dimensionlen Beispiel zwar als Punkte dargestellt werden können, bei mehr als drei Dimensionen sind diese allerdings mehrdimensionale Vektoren.



Durch die Abbildung zeigt sich, dass zur Bestimmung der trennenden Hyperebene in diesem Fall nur die Stützvektoren einen Einfluss nehmen. Die zentrale Gerade zwischen den Stützvektoren wird als “seperating hyperplane” bezeichnet und stellt die Hyperebene zur optimalen Trennung der Klassen dar.

3 SVM Algorithmus

Zur Bestimmung der Hyperebene bzw. der Stützvektoren wird der im Folgenden dargestellte Algorithmus verwendet.

Definition der trennenden Hyperebene:

$$\mathcal{H} := \{x \in \mathbb{R}^n \mid \langle w, x \rangle + b = 0\}$$

mit $w \in \mathbb{R}^n$ als zu \mathcal{H} orthogonal Vektor und $b \in \mathbb{R}$ als Verschiebung

Dies ist noch keine eindeutige Definition der Hyperebene, denn

$$\mathcal{H} = \{x \in \mathbb{R}^n \mid \langle aw, x \rangle + ab = 0\} \quad \forall a \in \mathbb{R} \setminus \{0\}$$

Diese muss noch normiert werden, durch

$$\min_{i=1, \dots, N} |\langle aw, x_i \rangle + ab| = 1$$

Der Abstand eines Punktes x_i zu einer Hyperebene lässt sich nun berechnen zu

$$y_i(\langle \frac{w}{\|w\|}, x_i \rangle + \frac{b}{\|w\|})$$

mit $\|w\|$ als Länge (Norm) des Vektors

Als Ergebnis ergibt sich durch Umformungen

$$\langle \frac{w}{\|w\|}, (x_1 - x_2) \rangle = \frac{2}{\|w\|}$$

Um eine tatsächliche Trennung der Trainingsdaten durch die Hyperebene zu erreichen, wird die folgende Nebenbedingung eingeführt

$$y_i(\langle w, x_i \rangle + b) \geq 1 \quad \forall i = 1, \dots, N$$

Somit entsteht ein Problem der quadratischen Programmierung, bei dem Lagrange-Multiplikatoren $\alpha_i \geq 0$ eingeführt. Die Lagrange-Funktion wird für w und b minimiert und für α_i maximiert.

$$L(w, b, \alpha) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^N \alpha_i (y_i(\langle w, x_i \rangle + b) - 1)$$

Werden die partiellen Ableitungen von L nach w und b gleich 0 gesetzt erhält man:

$$\sum_{i=1}^N \alpha_i y_i = 0 \text{ und } \sum_{i=1}^N \alpha_i y_i x_i$$

Für das Optimum gilt nun entweder

$$\alpha_i = 0 \text{ oder } y_i(\langle x_i, w \rangle) + b = 1$$

Somit haben nur die Punkte mit $\alpha_i \geq 0$ einen Einfluss auf die optimale Lösung. Diese Punkte werden als "support vectors" bzw. "Stützvektoren" bezeichnet.

Durch das Vorzeichen der Funktion kann nun über die Zuordnung zu einer Klasse entschieden werden.

$$f(x_{neu}) = \operatorname{sgn}\left(\sum_{i=1}^N \alpha_i y_i \langle x, x_i \rangle + b\right)$$

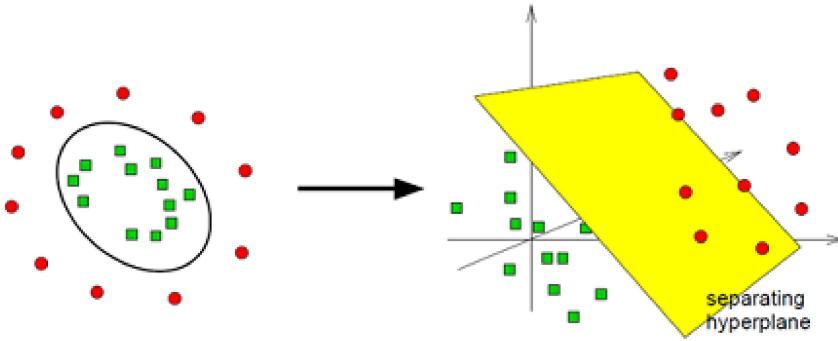
3.1 Besonderheiten der SVM

Im Gegensatz zu anderen Klassifikationsalgorithmen betrachtet eine SVM als Grundlage zur Einteilung in eine Klasse nicht die “typischen” Eigenschaften dieser Klassen. Stattdessen werden die am weitesten von Zentrum einer Klasse entfernten Vektoren miteinander verglichen. Aus dem Vergleich dieser kann die “separating hyperplane” ermittelt werden. Somit zeigt sich, dass die Anzahl an Werten irrelevant ist. Diese Eigenschaft unterscheidet eine SVM von vielen anderen Klassifikationsalgorithmen.

4 Nichtlineare Klassifikation

4.1 Grundlagen

Da in der Realität nicht alle Klassifikationsprobleme von Grund auf linear separierbar sind, können diese Fälle durch die Verwendung des Kern-Tricks dennoch linear separierbar gemacht werden. Betrachtet wird in Abbildung 3 der Fall, dass die vorliegenden Daten im Ursprungssraum nicht linear separierbar sind. Um diesen Fall nun dennoch durch eine Hyperebene linear separieren zu können werden der Vektorraum und die Trainingsdaten durch eine Funktion Φ in einen höherdimensionalen Raum überführt. Wird die Hyperebene anschließend in den Ursprungssraum übertragen wird die Hyperebene nichtlinear.



Durch eine Kernel-Funktion, die im Ursprungssraum definiert ist, ist es möglich, die diese im featur space wie ein Skalarprodukt zu behandeln.

$$K(p, q) = \langle \Phi(p), \Phi(q) \rangle$$

Das Skalarprodukt kann hierbei berechnet werden ohne die Funktion Φ zu berechnen. Das Vorgehen des Kernel-Tricks könnte als Blackbox angesehen werden, da der feature space und die Abbildung in diesen nicht bekannt sein muss. Übliche Kernel sind z.B.:

Linearer Kernel

$$K(p, q) = \langle p, q \rangle$$

Polynomialer Kernel

$$K(p, q) = (\gamma \langle p, q \rangle + c_0)^d$$

Radiale Basisfunktion (Gauss Kernel)

$$K(p, q) = \exp(-\gamma \|p - q\|)^2$$

5 Anwendungsbeispiele

Der SVM Algorithmus findet in diversen Bereichen Anwendung, z.B. in bei der Erkennung von Fingerabdrücken² oder Erkennung von Emotionen einer Testperson anhand des Pulses und des Alters.³ Außerdem werden SVM häufig für die Klassifikation von Handschrift verwendet.⁴ Neben diesen Beispielen findet sich in der Praxis auch der Anwendungsfall, Schäden an Rotorblättern von Windkraftanlagen zu klassifizieren.⁵

²<http://www.iaescore.com/journals/index.php/IJEECS/article/view/10018>

³<https://github.com/akasantony/pulse-classification-svm>

⁴https://github.com/ksopyla/svm_mnist_digit_classification

⁵<http://www.futureblades.com/>

6 Quellen

- T. Hastie, R. Tibshirani, J. Friedman. The elements of statistical learning
- F. Markowetz - Classification by SVM
- J. Fischer - Support Vector Machines (SVM)
- Schölkopf - Statistical Learning an Kernel Methods
- S. Raval- Support Vector Machines