Stützvektormaschine (SVM)

Autor und Herausgeber

Alexander Jung*, Department of Computer Science, Aalto University, Espoo, Finland (ORCID 0000-0001-7538-0990)
Editors: Konstantina Olioumtsevits (ORCID 0000-0003-2697-8659), Ekkehard Schnoor (ORCID 0000-0001-8252-1499), Aalto University, Espoo, Finland
*Corresponding author: alex.jung@aalto.fi

Zusammenfassung

Die Stützvektormaschine (SVM) ist ein Verfahren der binäre Klassifikation, das einen linearer Klassifikator mittels regularisierte empirische Risikominimierung (RERM) erlernt, indem es den Hinge-Verlust mit einem Strafterm aus einer quadrierten Norm kombiniert. Für eine linear trennbare Trainingsdatensatz und eine hinreichend schwache Regularisierung ist die Lösung die trennende Hyperebene mit maximalem Margin. Diese Hyperebene ist vollständig durch die ihr am nächsten liegenden Merkmalsvektoren bestimmt, die als Stütz-Vektoren bezeichnet werden. Die Kernel-SVM ergibt sich aus der Kombination der einfachen SVM mit einer aus einer Kernel-Funktion abgeleiteten Merkmalstransformation.

Definition

Eine handgeschriebene Ziffer wird anhand der Pixelwerte ihres Bildes erkannt, und ein Textdokument wird anhand der Häufigkeiten seiner Wörter nach Themen sortiert (Cristianini and Shawe-Taylor, 2000). Bei beiden Aufgaben muss ein numerischer Merkmalsvektor auf ein Label abgebildet werden, und die SVM wurde auf beide angewandt. Die SVM ist ein Verfahren der binäre Klassifikation für Datenpunkte mit dem Merkmalsraum $\mathcal{X} = \mathbb{R}^d$. In ihrer einfachsten Form erlernt die SVM einen linearer Klassifikator mit der Entscheidungsgrenze

$$\{\mathbf{x} \in \mathbb{R}^d : \mathbf{w}^\top \mathbf{x} + b = 0\}.$$

Der linearer Klassifikator wird durch einen von null verschiedenen Vektor $\mathbf{w} \in \mathbb{R}^d \setminus \{\mathbf{0}\}$ und einen Versatz $b \in \mathbb{R}$ parametrisiert. Der Vektor $\mathbf{w}$ ist der Normalenvektor zur Entscheidungsgrenze, und der Versatz b verschiebt die Entscheidungsgrenze vom Ursprung weg. Für einen Datenpunkt mit Merkmalsvektor $\mathbf{x}$ und Label $y \in \{-1, +1\}$ liefert die SVM die Vorhersage $\widehat{y} = \mathrm{sign}(\mathbf{w}^\top \mathbf{x} + b)$.

Die SVM lässt sich als eine Instanz der regularisierte empirische Risikominimierung (RERM) zum Erlernen eines linearer Klassifikator aus einer Trainingsdatensatz, die aus m Datenpunkte besteht, formulieren. Die Modellparameter $(\widehat{\mathbf{w}}, \widehat{b})$ des linearer Klassifikator ergeben sich als

$$(\widehat{\mathbf{w}}, \widehat{b}) = \underbrace{\operatorname*{arg\,min}_{\mathbf{w} \in \mathbb{R}^d,\, b \in \mathbb{R}} \frac{1}{m} \sum_{r=1}^{m} \max\{0,\, 1 - y^{(r)}(\mathbf{w}^\top \mathbf{x}^{(r)} + b)\} + \alpha \|\mathbf{w}\|_2^2}_{:= f(\mathbf{w}, b)}. \quad (1)$$

Die Zielfunktion der SVM besteht aus dem durchschnittlichen Hinge-Verlust auf der Trainingsdatensatz zuzüglich eines Strafterms $\alpha\|\mathbf{w}\|_2^2$ mit dem Regularisierungs-Parameter $\alpha > 0$ (Hastie et al., 2009, Sect. 12.3.2, p. 426; Shalev-Shwartz and Ben-David, 2014, Sect. 15.2, Eq. (15.5)). Der Strafterm im SVM-Problem (1) ist die skalierte quadrierte euklidische Norm $\alpha\|\mathbf{w}\|_2^2$, derselbe Strafterm wie bei der Ridge-Regression. Er hängt nicht vom Modellparameter b ab, der daher unbestraft bleibt.

Bei der lineare Regression schätzt der Strafterm, um wie viel höher der Verlust auf Datenpunkte außerhalb der Trainingsdatensatz ist als auf ihr (Hastie et al., 2009, Sect. 7.4, Eq. (7.24); siehe lineare Regression), eine Interpretation, die die quadratische Struktur des quadratischer Verlustes voraussetzt. Bei der SVM steuert der Strafterm stattdessen die Generalisierungsabstand: Der Hinge-Verlust ist in $\mathbf{w}$ Lipschitz-Stetigkeit-stetig, wobei die Konstante durch die euklidische Norm des Merkmalsvektors gegeben ist. Das Hinzufügen des Strafterms $\alpha\|\mathbf{w}\|_2^2$ zum durchschnittlichen Hinge-Verlust hat drei Effekte: Erstens macht es die gesamte Zielfunktion im SVM-Problem (1) in $\mathbf{w}$ stark konvex, sodass ihre Lösung $\widehat{\mathbf{w}}$ eindeutig ist (Shalev-Shwartz and Ben-David, 2014, Lemma 13.5). Zweitens steuert es, wie stark diese Lösung variiert, wenn sich die Datenpunkte in der Trainingsdatensatz ändern (Shalev-Shwartz and Ben-David, 2014, Cor. 13.6). Drittens hält es $\|\mathbf{w}\|_2$ klein und damit die Margins groß (Shalev-Shwartz and Ben-David, 2014, Sect. 15.2).

Wenn die Merkmalsvektoren in der Trainingsdatensatz eine beschränkte euklidische Norm aufweisen, nimmt der Erwartungswert der Differenz zwischen dem Risiko und dem empirischen Hinge-Verlust des erlernten linearer Klassifikator proportional zu $1/(\alpha m)$ ab (Shalev-Shwartz and Ben-David, 2014, Cor. 13.6, Cor. 15.7).

Die Lösung des SVM-Problems (1) hat eine geometrische Bedeutung, wenn die Trainingsdatensatz linear trennbar ist, d. h., wenn es eine Wahl für $(\mathbf{w}, b)$ gibt, sodass $\operatorname{sign}(\mathbf{w}^\top \mathbf{x}^{(r)} + b)y^{(r)} = 1$ für $r = 1, \ldots, m$ gilt. Für ein hinreichend kleines α bestimmt die Lösung des SVM-Problems (1) diejenige trennende Hyperebene, die am weitesten von den Merkmalsvektoren in der Trainingsdatensatz entfernt ist (siehe Abb. 1). Der Abstand zwischen dieser Hyperebene und den nächstgelegenen Merkmalsvektoren, bei denen es sich genau um die weiter unten besprochenen Stütz-Vektoren handelt, wird als Margin bezeichnet und ist durch $1/\|\widehat{\mathbf{w}}\|_2$ gegeben (Boser et al., 1992; Cortes and Vapnik, 1995; Cristianini and Shawe-Taylor, 2000; Hastie et al., 2009, Sect. 12.2, Eq. (12.4)).

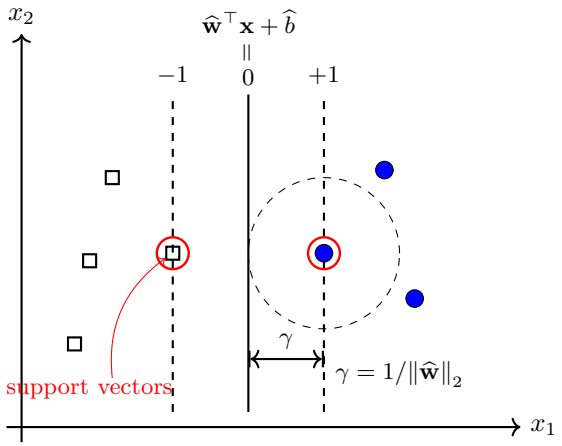

Abbildung 1: Trennung mit maximalem Margin für ein linear trennbares Trainingsdatensatz. Die Entscheidungsgrenze $\widehat{\mathbf{w}}^\top \mathbf{x} + \widehat{b} = 0$ (durchgezogene Linie) liegt genau in der Mitte zwischen zwei parallelen Hyperebenen $\widehat{\mathbf{w}}^\top \mathbf{x} + \widehat{b} = \pm 1$ (gestrichelte Linien). Die Stütz-Vektoren (offenes Quadrat und ausgefüllter Kreis in roten Ringen) sind die Merkmalsvektoren, die auf diesen Hyperebenen liegen. Der Margin $\gamma = 1/\|\widehat{\mathbf{w}}\|_2$ ist der Abstand von der Entscheidungsgrenze zu den Stütz-Vektoren. Datenpunkte mit dem Label $y = +1$ sind als ausgefüllte Kreise dargestellt, solche mit $y = -1$ als offene Quadrate. Der gestrichelte Kreis mit Radius γ zeigt, dass jeder Stütz-Vektor durch eine Störung mit einer euklidische Norm kleiner als γ verschoben werden kann, ohne die Entscheidungsgrenze zu überschreiten

Der Margin der erlernten Hyperebene misst die Robustheit gegenüber Störungen der Merkmale eines Datenpunkts. Wird ein Merkmalsvektor $\mathbf{x}$ um einen Vektor $\boldsymbol{\delta}$ gestört, ändert sich der Wert $\widehat{\mathbf{w}}^\top \mathbf{x} + \widehat{b}$ um $\widehat{\mathbf{w}}^\top \boldsymbol{\delta}$, dessen Betrag nach der Cauchy-Schwarz-Ungleichung höchstens $\|\widehat{\mathbf{w}}\|_2 \|\boldsymbol{\delta}\|_2$ beträgt. Ein Datenpunkt bleibt daher unter jeder Störung der Merkmale mit $\|\boldsymbol{\delta}\|_2 < \gamma$ korrekt klassifiziert, wie durch den gestrichelten Kreis um einen Stütz-Vektor in Abb. 1 veranschaulicht wird. Datenpunkte, deren Merkmalsvektoren weiter von der Entscheidungsgrenze entfernt liegen, tolerieren noch größere Störungen: Der tolerierte Radius entspricht dem Abstand des Merkmalsvektors von der Entscheidungsgrenze, der mindestens γ beträgt und genau für die Stütz-Vektoren gleich γ ist. Die Maximierung des Margins maximiert diesen im ungünstigsten Fall tolerierten Radius, sodass der Klassifikator mit maximalem Margin derjenige trennende Klassifikator ist, der gegenüber den größten Störungen der Merkmale robust ist. Dasselbe gilt über den trennbaren Fall hinaus: Das SVM-Problem (1) ist äquivalent zur Minimierung des Hinge-Verlustes im ungünstigsten Fall über beschränkte Störungen der Merkmalsvektoren (Xu et al., 2009, Thm. 3).

Jede Lösung des SVM-Problems (1) weist eine Struktur auf, die unabhängig

davon gilt, ob die Trainingsdatensatz linear trennbar ist oder nicht. Insbesondere gilt

$$\widehat{\mathbf{w}} = \sum_{r=1}^{m} \beta_r y^{(r)} \mathbf{x}^{(r)}$$

mit Entwicklungskoeffizienten $0 \leq \beta_r \leq 1/(2\alpha m)$, die nur für diejenigen Merkmalsvektoren von null verschieden sind, für die

$$y^{(r)}(\widehat{\mathbf{w}}^\top \mathbf{x}^{(r)} + \widehat{b}) \leq 1.$$

gilt. Diese Merkmalsvektoren werden als Stütz-Vektoren bezeichnet, da sie die SVM-Lösung $(\widehat{\mathbf{w}}, \widehat{b})$ vollständig bestimmen (Boser et al., 1992; Cortes and Vapnik, 1995, Sect. 2). Wird auf einen Merkmalsvektor, der kein Stütz-Vektor ist, eine Störung angewandt, bleiben $\widehat{\mathbf{w}}$, $\widehat{b}$ unverändert, sofern die Störung klein genug ist, um die strikte Ungleichung $y(\widehat{\mathbf{w}}^\top \mathbf{x} + \widehat{b}) > 1$ zu erhalten (siehe Abb. 2, deren Trainingsdatensatz einen Ausreißer enthält).

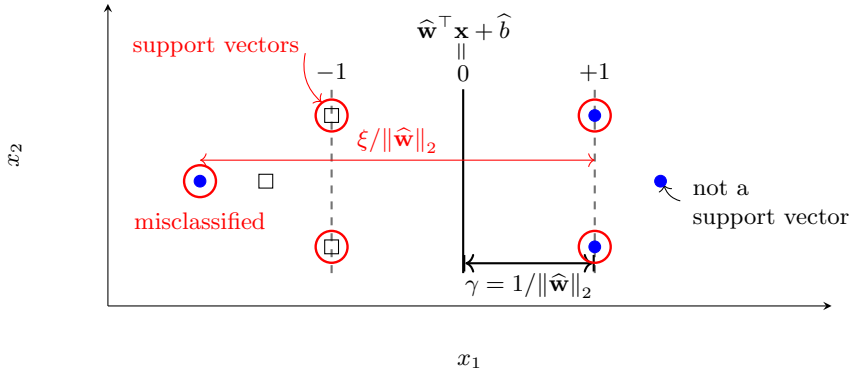

Abbildung 2: Nicht trennbares Trainingsdatensatz aus $m = 7$ Datenpunkten im Merkmalsraum $\mathbb{R}^2$. Die Lösung des SVM-Problems (1) platziert die Entscheidungsgrenze $\widehat{\mathbf{w}}^\top \mathbf{x} + \widehat{b} = 0$ (durchgezogene Linie) zwischen die beiden Hyperebenen $\widehat{\mathbf{w}}^\top \mathbf{x} + \widehat{b} = \pm 1$ (gestrichelte Linien) im Abstand $\gamma = 1/\|\widehat{\mathbf{w}}\|_2$. Diese beiden Hyperebenen enthalten vier der Stütz-Vektoren (rote Ringe), die durch die Datenpunkte mit einem Margin gleich γ gegeben sind. Der Ausreißer mit dem Label $+1$ ist ebenfalls ein Stütz-Vektor; er liegt falsch klassifiziert jenseits des gegenüberliegenden Margins: $y(\widehat{\mathbf{w}}^\top \mathbf{x} + \widehat{b}) = -2 < -1$, mit dem Hinge-Verlust-Wert ξ. Die beiden Datenpunkte (offenes Quadrat und ausgefüllter Kreis ohne rote Ringe), die weiter von der Entscheidungsgrenze entfernt sind als der Margin γ, sind keine Stütz-Vektoren. Daten erzeugt mit `pythondemos/svm.py`

Wie die logistische Regression und die lineare Regression erlernt die SVM die Modellparameter $(\mathbf{w}, b)$ eines lineares Modells. Im Gegensatz zur logistische Regression und zur lineare Regression ist die Zielfunktion $f(\mathbf{w}, b)$ der SVM aufgrund des Hinge-Verlustes nichtglatt. Da der Hinge-Verlust jedoch das Maximum zweier affiner Funktionen von $(\mathbf{w}, b)$ und somit konvex ist (Boyd and

Vandenberghe, 2004, Sect. 3.2.3), ist die Zielfunktion der SVM, also die Summe aus seinem Durchschnitt und dem konvexen Strafterm, ebenfalls konvex, und das Optimierungsproblem (1) kann mit Verfahren der konvexe Optimierung gelöst werden. Ein solches Verfahren ist der Subgradientenabstieg, der aus dem Gradientenabstieg hervorgeht, indem der Gradient der Zielfunktion durch einen Subgradienten ersetzt wird (Boyd and Vandenberghe, 2004; Polyak, 1987, Sect. 5.3).
Ausgehend von einer Anfangswahl $\mathbf{w}^{(0)}, b^{(0)}$ der Modellparameter wendet das Subgradientenabstieg-Verfahren wiederholt die Aktualisierung

$$\big(\mathbf{w}^{(t+1)}, b^{(t+1)}\big) = \big(\mathbf{w}^{(t)}, b^{(t)}\big) - \eta^{(t)}\mathbf{g}^{(t)}, \text{ für } t = 0, 1, \ldots,$$

mit einer Schrittweite $\eta^{(t)} > 0$ und einem Subgradienten $\mathbf{g}^{(t)} \in \partial f\big(\mathbf{w}^{(t)}, b^{(t)}\big)$ der SVM-Zielfunktion (1) an. Das Einsetzen der SVM-Zielfunktion (1) in die allgemeine Aktualisierung liefert die explizite Aktualisierung

$$\mathbf{w}^{(t+1)} = \big(1 - 2\eta^{(t)}\alpha\big)\mathbf{w}^{(t)} + \frac{\eta^{(t)}}{m}\sum_{r=1}^{m}\beta_r^{(t)} y^{(r)}\mathbf{x}^{(r)},$$

$$b^{(t+1)} = b^{(t)} + \frac{\eta^{(t)}}{m}\sum_{r=1}^{m}\beta_r^{(t)} y^{(r)},$$

mit Entwicklungskoeffizienten $\beta_r^{(t)} \in [0, 1]$, für $r = 1, \ldots, m$. Der Koeffizient $\beta_r^{(t)}$ ist nur für solche Datenpunkte von null verschieden, die $y^{(r)}\big(\big(\mathbf{w}^{(t)}\big)^\top \mathbf{x}^{(r)} + b^{(t)}\big) \leq 1$ erfüllen, d. h. für Datenpunkte, bei denen der Hinge-Verlust bei den aktuellen Modellparametern von null verschieden oder nicht differenzierbar ist. Jede Iteration skaliert $\mathbf{w}^{(t)}$ mit dem Faktor $1 - 2\eta^{(t)}\alpha$ (dem Effekt des Strafterms) und addiert Korrekturen $y^{(r)}\mathbf{x}^{(r)}$ nur von diesen Datenpunkten.

Insbesondere ist für die Initialisierung $\mathbf{w}^{(0)} = \mathbf{0}$ jede Iterierte $\mathbf{w}^{(t)}$ eine gewichtete Summe $\sum_{r=1}^{m}\beta_r y^{(r)}\mathbf{x}^{(r)}$ der Merkmalsvektoren im Trainingsdatensatz. Die gewichtete Summe hat die Form der oben angegebenen Entwicklung der Lösung $\widehat{\mathbf{w}}$ nach den Stütz-Vektoren, die Schritt für Schritt über jede Iteration aufgebaut wird. Ein Koeffizient der Iterierten $\mathbf{w}^{(t)}$ ist nur für einen Datenpunkt von null verschieden, der in einer früheren Iteration $t' < t$ die Bedingung $y^{(r)}\big(\big(\mathbf{w}^{(t')}\big)^\top \mathbf{x}^{(r)} + b^{(t')}\big) \leq 1$ erfüllt hat. Für Schrittweiten mit $2\eta^{(t')}\alpha < 1$ bleibt ein solcher Koeffizient danach positiv, da jede spätere Iteration ihn mit einem positiven Faktor umskaliert und einen nichtnegativen Betrag addiert. Die Iterierte kann daher auch Merkmalsvektoren enthalten, die keine Stütz-Vektoren der Lösung $\widehat{\mathbf{w}}$ sind. Für eine konstante Schrittweite $\eta = B/(\rho\sqrt{T})$ über T Iterationen, wobei B die Norm eines Minimierers von (1) und ρ die Norm der verwendeten Subgradienten beschränkt, übersteigt die SVM-Zielfunktion am Mittelwert der Iterierten ihr Minimum um höchstens $B\rho/\sqrt{T}$ (Shalev-Shwartz and Ben-David, 2014, Thm. 14.8).

Ein übermäßig großer Regularisierungs-Parameter α führt bei der SVM zu einer Unteranpassung: Die obige Entwicklung von $\widehat{\mathbf{w}}$ impliziert $\|\widehat{\mathbf{w}}\|_2 \leq \rho/(2\alpha)$, wobei ρ die größte euklidische Norm eines Merkmalsvektors im Trainingsdatensatz ist, sodass $\widehat{\mathbf{w}}$ gegen $\mathbf{0}$ schrumpft. Betrachtet sei ein Trainingsdatensatz

mit ungleichen Klassengrößen. Für hinreichend großes α nähern sich die Vorhersagen der konstanten Mehrheitsklassen-Vorhersage $\text{sign}(\widehat{b})$ an, und jeder Datenpunkt der Minderheitsklasse ist ein falsch klassifizierter Stütz-Vektor. Das Trainingsdatensatz aus Abb. 2 weist ebenfalls ungleiche Klassengrößen auf, vier Datenpunkte mit dem Label $y = +1$ und drei mit dem Label $y = -1$, doch mit dem dort verwendeten Wert $\alpha = 1/14$ klassifiziert die SVM einen Datenpunkt falsch, der als Ausreißer betrachtet werden kann. Eine Erhöhung des Wertes von α auf 50 für dieses Trainingsdatensatz verkleinert $\|\widehat{\mathbf{w}}\|_2$, setzt jede Vorhersage auf dem Trainingsdatensatz gleich $+1$ und klassifiziert alle drei Datenpunkte der Minderheitsklasse falsch.

Die grundlegende SVM (1) lernt einen linearer Klassifikator auf dem Merkmalsraum $\mathcal{X} = \mathbb{R}^d$. Sie lässt sich zu einer Klassifikationsmethode für Datenpunkte verallgemeinern, deren Merkmalsvektoren in einem beliebigen Merkmalsraum $\mathcal{X}$ liegen, indem zunächst eine Merkmalstransformation $\phi : \mathcal{X} \to \mathcal{X}'$ mit $\mathcal{X}' = \mathbb{R}^d$ angewendet und anschließend ein linearer Klassifikator auf den transformierten Merkmalsvektoren gelernt wird. Durch geeignete Wahl von ϕ (und d) kann jedes gegebene Trainingsdatensatz mit paarweise verschiedenen Merkmalsvektoren in $\mathcal{X}'$ linear trennbar gemacht werden, z. B. indem jeder verschiedene Merkmalsvektor auf seine eigene Koordinatenachse abgebildet wird (Cristianini and Shawe-Taylor, 2000). Eine allgemeine Konstruktion einer Merkmalstransformation ϕ erfolgt über eine Kernel $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$. Bei dieser Konstruktion ist der transformierte Merkmalsraum im Allgemeinen nicht $\mathbb{R}^d$, sondern ein Hilbertraum $\mathcal{H}$, der unendlichdimensional sein kann (siehe Kernel-Methode). Die resultierende Methode wird dann als Kernel-SVM bezeichnet (Schölkopf and Smola, 2002).

Synonyme: Maximum-Margin-Klassifikator.

Siehe auch: binäre Klassifikation, Hinge-Verlust, Margin, Hyperebene, Entscheidungsgrenze, Robustheit, Kernel-Methode, Subgradientenabstieg.

Querverweise

Literatur

1. Boser et al. (1992). A Training Algorithm for Optimal Margin Classifiers. In: Proc. 5th Annual Workshop on Computational Learning Theory (COLT), pp. 144–152. ACM.

2. Boyd and Vandenberghe (2004). Convex Optimization. Cambridge Univ. Press, Cambridge, U.K.

3. Cristianini and Shawe-Taylor (2000). An Introduction to Support Vector Machines and Other Kernel-based Learning Methods. Cambridge Univ. Press, New York, NY, USA.

4. Schölkopf and Smola (2002). Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. MIT Press, Cambridge, MA, USA.

5. Polyak (1987). Introduction to optimization. Optimization Software, Inc., Publications Division, New York.

6. Shalev-Shwartz and Ben-David (2014). Understanding Machine Learning: From Theory to Algorithms. Cambridge Univ. Press, New York, NY, USA.

7. Xu et al. (2009). Robustness and Regularization of Support Vector Machines. J. Mach. Learn. Res., 10(51), pp. 1485–1510.

8. Cortes and Vapnik (1995). Support-vector networks. Machine learning, 20(3), pp. 273–297.

9. Hastie et al. (2009). The Elements of Statistical Learning: Data Mining, Inference, and Prediction. 2nd ed., Springer Science+Business Media, New York, NY, USA.