



Vorlesung Wissensentdeckung

Stützvektormethode

Katharina Morik, Uwe Ligges

LS 8 Informatik
Computergestützte Statistik
Technische Universität Dortmund

10.6.2010



Gliederung

- 1 Hinführungen zur SVM
- 2 Maximum Margin Methode
 - Lagrange-Optimierung
- 3 Weich trennende SVM



Übersicht über die Stützvektormethode (SVM)

Eigenschaften der Stützvektormethode (SVM) (Support Vector Machine)

- Maximieren der Breite einer separierenden Hyperebene – maximum margin method – ergibt eindeutige, optimale trennende Hyperebene.
- Transformation des Datenraums durch Kernfunktion behandelt Nichtlinearität.
- Strukturelle Risikominimierung minimiert nicht nur den Fehler, sondern auch die Komplexität des Modells.



Einführende Literatur

- Vladimir Vapnik “The Nature of Statistical Learning Theory” Springer Vg. 1995
- W.N. Wapnik, A. Tscherwonenkis “Theorie der Zeichenerkennung” Akademie Vg. 1979
- Christopher Burges ”A Tutorial on Support Vector Machines for Pattern Recognition” in: Data Mining and Knowledge Discovery 2, 1998, 121-167

Vertiefung: Bernhard Schölkopf, Alexander Smola “Learning with Kernels”, MIT Press, 2002



Probleme der Empirischen Risikominimierung

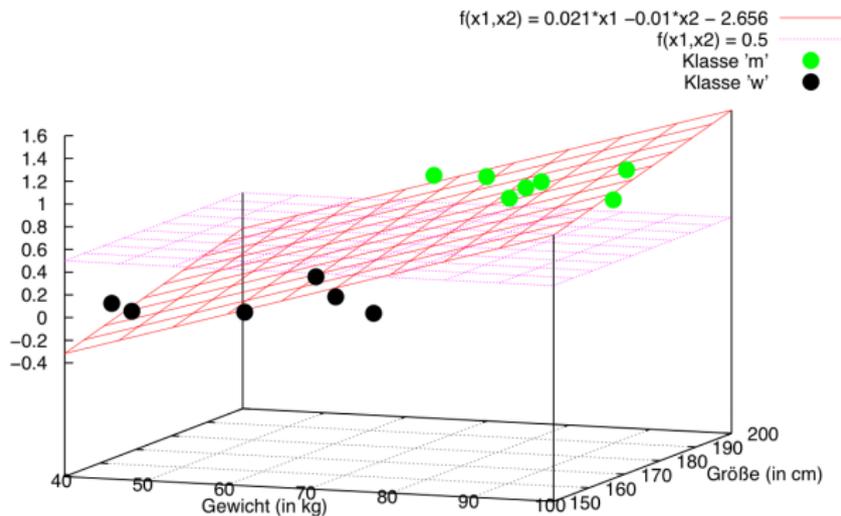
Empirische Risikominimierung: Bisher haben wir lineare Modelle

$$\hat{Y} = \hat{\beta}_0 + \sum_{j=1}^p X_j \hat{\beta}_j$$

auf die Fehlerminimierung hin optimiert:

$$RSS(\hat{\beta}) = \sum_{i=1}^N (y_i - \vec{x}_i^T \hat{\beta})^2$$

Wo trennen wir die Daten?



Problem: Mehrere Funktionen mit minimalem Fehler existieren.
 Welche wählen?

- 1. **Ausweg:** Verbessertes Kriterium: **maximum margin**.
- 2. **Ausweg:** Zusätzliches Kriterium: möglichst geringe Komplexität des Modells (**Strukturelle Risikominimierung**)

Klassifikationsproblem

Gegeben sei ein Klassifikationsproblem mit $Y = \{-1; +1\}$ und $\mathbf{X} \subseteq \mathbb{R}^p$.

Sei $\mathbf{X} = C_+ \dot{\cup} C_-$ die Menge der Trainingsbeispiele mit

$$C_+ = \{(\vec{x}, y) \mid y = +1\} \quad \text{und} \quad C_- = \{(\vec{x}, y) \mid y = -1\}$$

Zur Klassifikation ist nun eine Hyperebene

$$H = \left\{ \vec{x} \mid \beta_0 + \langle \vec{x}, \vec{\beta} \rangle = 0 \right\}$$

gesucht, die die Mengen C_+ und C_- *bestmöglichst* trennt

Für eine gegebene Hyperebene H erfolgt die Klassifikation dann durch

$$\hat{y} = \text{sign} \left(\beta_0 + \langle \vec{x}, \vec{\beta} \rangle \right)$$

Notationen...

Und warum jetzt $\langle \vec{x}, \vec{\beta} \rangle$ statt $\vec{x}^T \vec{\beta}$?

Wir bewegen uns derzeit in einem \mathbb{R} -Vektorraum der Beispiele mit dem Standardskalarprodukt

$$\langle \vec{x}, \vec{\beta} \rangle = \underbrace{\vec{x}^T \vec{\beta}}_{\text{Matrixmultiplikation}} = \underbrace{\vec{x} \vec{\beta}}_{\text{Implizites Skalarprodukt}}$$

Die Notation $\langle \vec{x}, \vec{\beta} \rangle$ sollte aus der linearen Algebra (Schule?) bekannt sein.

Klassifikation mit Hyperebenen

Ist eine Ebene \tilde{H} mit

$$\tilde{H} = \left\{ \vec{x} \mid \beta_0 + \langle \vec{x}, \vec{\beta} \rangle = 0 \right\}$$

gegeben, können wir diese in Hesse-Normalenform überführen

$$H = \left\{ \vec{x} \mid \beta_0^* + \langle \vec{x}, \vec{\beta}^* \rangle = 0 \right\} \quad \text{mit } \vec{\beta}^* := \frac{\vec{\beta}}{\|\vec{\beta}\|}, \beta_0^* := \frac{\beta_0}{\|\vec{\beta}\|}$$

und erhalten die vorzeichenbehaftete Distanz eines Punktes \vec{x} zu H durch

$$d(\vec{x}, H) = \langle \vec{x} - \vec{x}_0, \vec{\beta}^* \rangle = \frac{1}{\|\vec{\beta}\|} \left(\langle \vec{x}, \vec{\beta} \rangle + \beta_0 \right)$$

(Übungsaufgabe)

Klassifikation mit Hyperebenen

Die vorzeichenbehaftete Distanz $d(\vec{x}, H)$ drückt aus

- 1 den Abstand $|d(\vec{x}, H)|$ von \vec{x} zu Ebene H
- 2 die Lage von \vec{x} relativ zur Orientierung $(\vec{\beta})$ von H , d.h.

$$\text{sign}(d(\vec{x}, H)) = \begin{cases} +1 & , \text{ falls } \cos \angle(\vec{x}, \vec{\beta}) \geq 0 \\ -1 & , \text{ sonst} \end{cases}$$

Auf diese Weise lassen sich die Punkte klassifizieren mit

$$\hat{y} = \text{sign} \left(\beta_0 + \langle \vec{x}, \vec{\beta} \rangle \right)$$

Einfacher Ansatz nach Schölkopf/Smola

Ein einfacher Ansatz zu einer separierenden Hyperebene zu kommen, geht über die Zentroiden von C_+ und C_- :

Seien

$$\vec{c}_+ := \frac{1}{|C_+|} \sum_{(\vec{x}, y) \in C_+} \vec{x} \quad \text{und} \quad \vec{c}_- := \frac{1}{|C_-|} \sum_{(\vec{x}, y) \in C_-} \vec{x}$$

Wähle nun

$$\vec{x}_0 := \frac{\vec{c}_+ + \vec{c}_-}{2} \quad \text{und} \quad \vec{\beta} := \vec{c}_+ - \vec{c}_-$$

als Hyperebene mit Normalenvektor $\vec{\beta}$ durch den Punkt \vec{x}_0

Separierende Hyperebene über Zentroiden

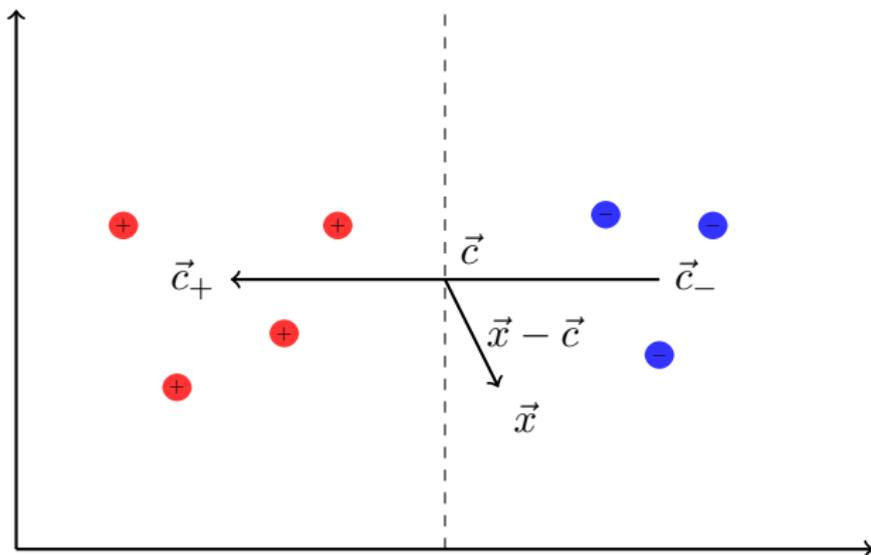
Durch $\vec{\beta}$ und \vec{x}_0 ist die Hyperebene gegeben als

$$\tilde{H} = \left\{ \vec{x} \mid \langle \vec{x} - \vec{x}_0, \vec{\beta} \rangle = 0 \right\} = \left\{ \vec{x} \mid \langle \vec{x}, \vec{\beta} \rangle - \underbrace{\langle \vec{x}_0, \vec{\beta} \rangle}_{=:-\beta_0} = 0 \right\}$$

Damit erfolgt die Klassifikation durch

$$\begin{aligned} \hat{y} &= \text{sign} \left(\langle \vec{x} - \vec{c}, \vec{\beta} \rangle \right) \\ &= \text{sign} \left(\langle \vec{x}, \vec{c}_+ \rangle - \langle \vec{x}, \vec{c}_- \rangle + \beta_0 \right) \quad (\text{Übung}) \end{aligned}$$

Lernalgorithmus im Bild





Fast...

... wäre das schon die Stützvektormethode. Aber:

- Einfach den Mittelpunkt der Beispiele einer Klasse zu berechnen ist zu einfach, um ein ordentliches $\vec{\beta}$ zu bekommen.
- Man erhält so nicht die optimale Hyperebene.



Die optimale Hyperebene

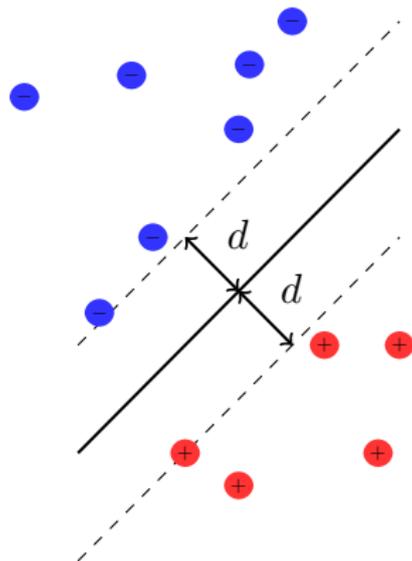
Eine Menge von Beispielen heißt **linear trennbar**, falls es eine Hyperebene H gibt, die die positiven und negativen Beispiele trennt.

5.1: Optimale Hyperebene

Eine separierende Hyperebene H heißt **optimal**, wenn ihr Abstand d zum nächsten positiven und nächsten negativen Beispiel maximal ist.

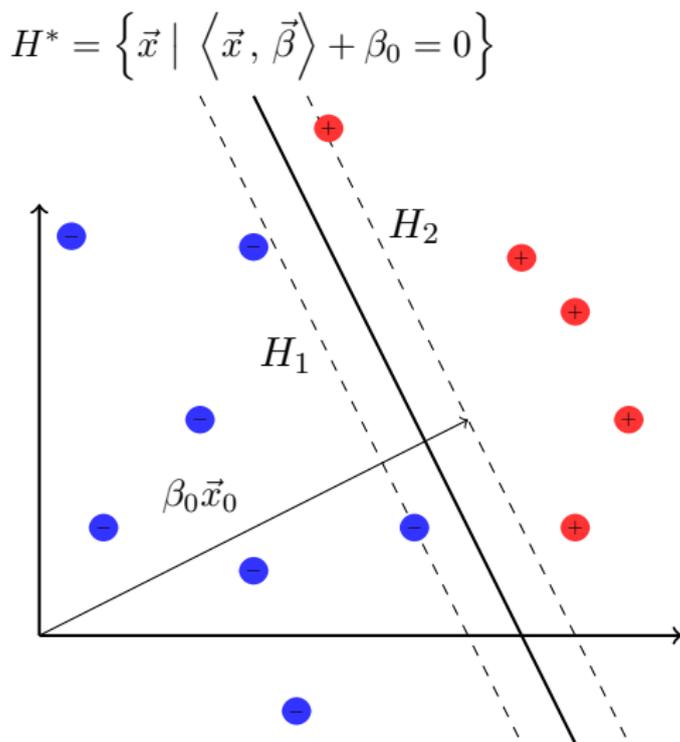
5.2: Satz (Eindeutigkeit)

Es existiert eine eindeutig bestimmte optimale Hyperebene.





Bild



Nach 5.1 wird die optimale Hyperebene durch die nächstliegenden Punkte aus C_+ und C_- bestimmt.

Skalierung von $\vec{\beta}$ und β_0 , so dass

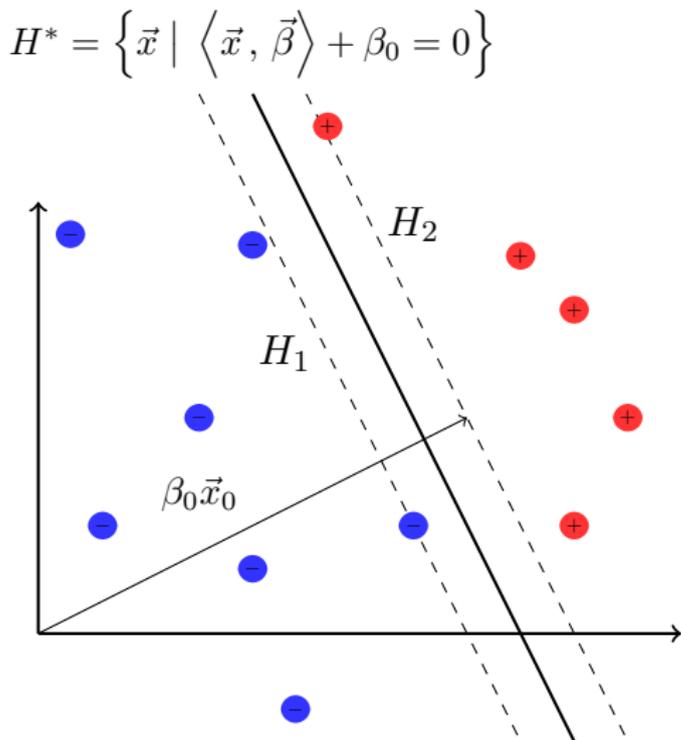
$$|\langle \vec{\beta}, \vec{x} \rangle + \beta_0| = 1$$

für alle Beispiele am nächsten zur Hyperebene liefert die Hyperebenen H_1 und H_2

$$H_j = \left\{ \vec{x} \mid \langle \vec{x}, \vec{\beta} \rangle + \beta_0 = (-1)^j \right\}$$



Abstand der Hyperebenen zum Ursprung



Der Abstand der mittleren Ebene H^* zum Ursprung beträgt

$$d(\vec{0}, H^*) = \frac{\beta_0}{\|\vec{\beta}\|}$$

Die Abstände der grauen Ebenen H_1 und H_2 sind

$$d(\vec{0}, H_j) = \frac{\beta_0 + (-1)^j}{\|\vec{\beta}\|}$$

$$\stackrel{H_1 \parallel H_2}{\Rightarrow} d(H_1, H_2) = \frac{2}{\|\vec{\beta}\|}$$

Margin

Nach Konstruktion liegt kein Beispiel zwischen H_1 und H_2 , d.h.

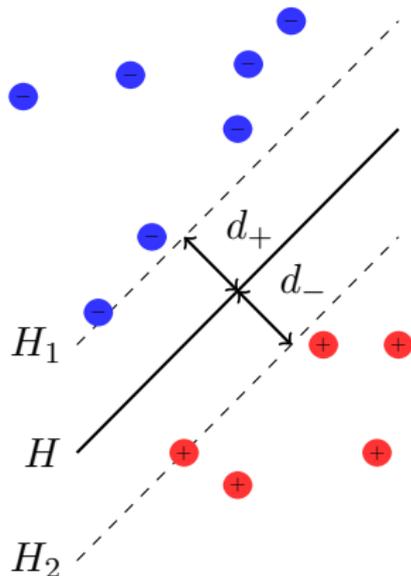
$$\langle \vec{x}, \vec{\beta} \rangle + \beta_0 \geq +1 \forall \vec{x} \in C_+ \quad (1)$$

$$\langle \vec{x}, \vec{\beta} \rangle + \beta_0 \leq -1 \forall \vec{x} \in C_- \quad (2)$$

Der Abstand

$$d(H_1, H_2) = \frac{2}{\|\vec{\beta}\|}$$

heißt **Margin** und soll maximiert werden!



Maximum Margin

Mit der Maximierung des Margin finden wir eine **optimale Hyperebene** innerhalb der Menge der möglichen trennenden Hyperebenen.

Durch die Minimierung von $\frac{1}{2} \|\vec{\beta}\|^2$ erhalten wir ein konvexes, quadratisches Optimierungsproblem, d.h.

- Es existiert eine eindeutig bestimmte, optimale Hyperebene

$$H^* = \left\{ \vec{x} \mid \langle \vec{x}, \vec{\beta} \rangle + \beta_0 = 0 \right\}$$

Das quadratische Optimierungsproblem läßt sich in Zeit $O(N^3)$ lösen.



Optimierungsaufgabe

Nach diesen Vorüberlegungen haben wir also (nur noch) die folgende Optimierungsaufgabe zu lösen:

Optimierungsaufgabe

Minimiere

$$\frac{1}{2} \|\vec{\beta}\|^2$$

unter den Nebenbedingungen

$$\langle \vec{x}, \vec{\beta} \rangle + \beta_0 \geq +1 \quad \forall \vec{x} \in C_+$$

$$\langle \vec{x}, \vec{\beta} \rangle + \beta_0 \leq -1 \quad \forall \vec{x} \in C_-$$

Die Nebenbedingungen lassen sich zusammenfassen zu

$$y(\langle \vec{x}, \vec{\beta} \rangle + \beta_0) - 1 \geq 0 \quad \forall (\vec{x}, y) \in \mathbf{X} \quad (3)$$

Optimierung mit Nebenbedingungen

Sei die optimierende Funktion $f : \mathbb{R} \rightarrow \mathbb{R}$ gegeben als

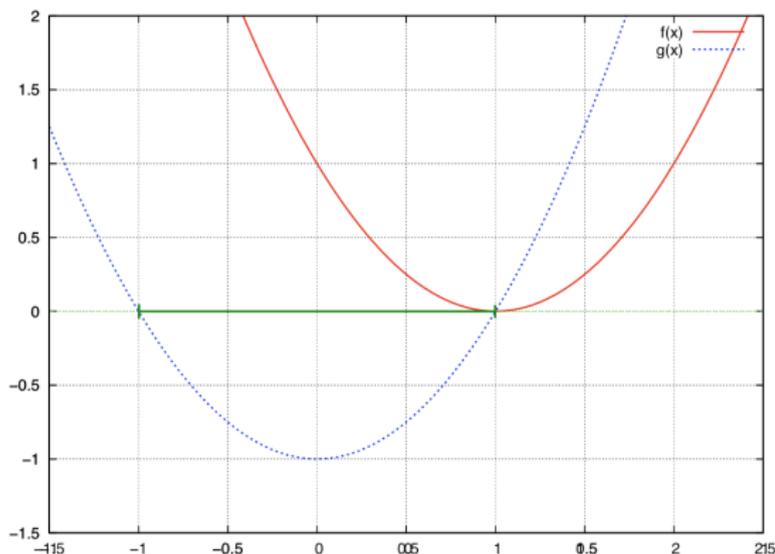
$$f(x) = (x - 1)^2$$

unter der einzigen Nebenbedingung

$$g(x) = x^2 - 1,$$

d.h. für die möglichen Lösungen \tilde{x} muss gelten

$$\tilde{x} \in \{x \in \mathbb{R} \mid g(x) \leq 0\}$$



Optimierung mit Lagrange

Die Optimierung nach Lagrange ermöglicht die Optimierung einer Funktion $f(x)$ unter Nebenbedingungen durch *Relaxation*.

Mit der Lagrange-Methode lassen sich Nebenbedingungen g_i und h_j der Art

$$g_i(x) \leq 0 \quad \text{und} \quad h_j(x) = 0$$

behandeln, indem diese zur zu optimierenden Funktion f hinzugefügt werden, im Falle eines Minimierungsproblems als

$$\min f(x) + \sum_i \alpha_i g_i(x) + \sum_j \mu_j h_j(x) \quad \text{mit} \quad \alpha_i, \mu_j \geq 0 \quad \forall i, j$$

Die α_i und μ_j heißen auch **Lagrange-Multiplikatoren**.

Lagrange-Funktion

Die Umformung der Nebenbedingungen (3) erlaubt nun die Anwendung von Lagrange (nur Ungleichheitsbedingungen):

Lagrange-Funktion

Sei das Optimierungsproblem gegeben, $f(\vec{\beta})$ zu minimieren unter den Nebenbedingungen $g_i(\vec{\beta}) \geq 0, i = 1, \dots, m$ dann ist die Lagrange-Funktion:

$$L(\vec{\beta}, \vec{\alpha}) = f(\vec{\beta}) - \sum_{i=1}^m \alpha_i g_i(\vec{\beta}) \quad (4)$$

Dabei muss gelten $\alpha_i \geq 0$, Gleichheitsbedingungen sind nicht gegeben.

Optimierungsfunktion als Lagrange

Die Nebenbedingungen g_i sind gegeben durch

$$g_i(\vec{\beta}, \beta_0) = y_i \left(\langle \vec{x}_i, \vec{\beta} \rangle + \beta_0 \right) - 1 \geq 0 \quad \forall \vec{x}_i \in \mathbf{X}$$

Die Formulierung des Optimierungsproblems nach Lagrange wird auch als **Primales Problem** bezeichnet:

Primales Problem

Die Funktion

$$L_P(\vec{\beta}, \beta_0, \vec{\alpha}) = \frac{1}{2} \|\vec{\beta}\|^2 - \sum_{i=1}^N \alpha_i \left(y_i \left(\langle \vec{x}_i, \vec{\beta} \rangle + \beta_0 \right) - 1 \right) \quad (5)$$

soll L_P bezüglich $\vec{\beta}$ und β_0 *minimiert* und bezüglich $\vec{\alpha}$ *maximiert* werden!

Karush-Kuhn-Tucker Bedingungen

Durch die partiellen Ableitung nach $\vec{\beta}$ und β_0 erhalten wir

$$\frac{\partial}{\partial \vec{\beta}} L_P(\vec{\beta}, \beta_0, \vec{\alpha}) = \vec{\beta} - \sum_i \alpha_i y_i \vec{x}_i \quad \text{und} \quad \frac{\partial}{\partial \beta_0} L_P(\vec{\beta}, \beta_0, \vec{\alpha}) = - \sum_i \alpha_i y_i$$

Nullsetzen der Ableitungen und die Berücksichtigung der Nebenbedingungen führt zu den KKT-Bedingungen für eine Lösung für L_P :

$$\vec{\beta} = \sum_{i=1}^N \alpha_i y_i \vec{x}_i \quad \text{und} \quad \sum_{i=1}^N \alpha_i y_i = 0 \quad (6)$$

$$\alpha_i \geq 0 \quad \forall i = 1, \dots, N \quad (7)$$

$$\alpha_i \left(y_i \left(\langle \vec{x}_i, \vec{\beta} \rangle + \beta_0 \right) - 1 \right) = 0 \quad \forall i = 1, \dots, N \quad (8)$$

Duales Problem

Das primale Problem soll bezüglich $\vec{\beta}$ und β_0 minimiert und bezüglich $\vec{\alpha}$ maximiert werden:

Mit den Bedingungen aus $\frac{\partial L_P}{\partial \vec{\beta}}$ und $\frac{\partial L_P}{\partial \beta_0}$ erhalten wir den *dualen Lagrange-Ausdruck* $L_D(\vec{\alpha})$

- Der duale Lagrange-Ausdruck $L(\vec{\alpha})$ soll maximiert werden.
- Das Minimum des ursprünglichen Optimierungsproblems tritt genau bei jenen Werten von $\vec{\beta}, \beta_0, \vec{\alpha}$ auf wie das Maximum des dualen Problems.

Umformung des primalen in das duale Problem

$$\begin{aligned}
 & \frac{1}{2} \|\vec{\beta}\|^2 - \sum_{i=1}^N \alpha_i \left[y_i \left(\langle \vec{x}_i, \vec{\beta} \rangle + \beta_0 \right) - 1 \right] \\
 &= \frac{1}{2} \|\vec{\beta}\|^2 - \sum_{i=1}^N \alpha_i y_i \left(\langle \vec{x}_i, \vec{\beta} \rangle + \beta_0 \right) + \sum_{i=1}^N \alpha_i \\
 &= \frac{1}{2} \|\vec{\beta}\|^2 - \sum_{i=1}^N \alpha_i y_i \langle \vec{x}_i, \vec{\beta} \rangle - \sum_{i=1}^N \alpha_i y_i \beta_0 + \sum_{i=1}^N \alpha_i \\
 &\stackrel{(6)}{=} \frac{1}{2} \|\vec{\beta}\|^2 - \sum_{i=1}^N \alpha_i y_i \langle \vec{x}_i, \vec{\beta} \rangle + \sum_{i=1}^N \alpha_i
 \end{aligned}$$

Umformung II

Einsetzen von $\vec{\beta} = \sum_{i=1}^N \alpha_i y_i \vec{x}_i$ führt zu

$$\begin{aligned}
 & \frac{1}{2} \|\vec{\beta}\|^2 && - \sum_{i=1}^N \alpha_i y_i \langle \vec{x}_i, \vec{\beta} \rangle && + \sum_{i=1}^N \alpha_i \\
 = & \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j \langle \vec{x}_i, \vec{x}_j \rangle && - \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j \langle \vec{x}_i, \vec{x}_j \rangle && + \sum_{i=1}^N \alpha_i \\
 = & + \sum_{i=1}^N \alpha_i && - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j \langle \vec{x}_i, \vec{x}_j \rangle
 \end{aligned}$$

unter den Nebenbedingungen $0 = \sum_{i=1}^N \alpha_i y_i$ und $\alpha_i \geq 0 \forall i$

SVM Optimierungsproblem (Duales Problem)

Die Umformungen führen nach Einsetzen der KKT-Bedingungen zum **dualen Problem**:

Duales Problem

Maximiere

$$L_D(\vec{\alpha}) = \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N y_i y_j \alpha_i \alpha_j \langle \vec{x}_i, \vec{x}_j \rangle \quad (9)$$

unter den Bedingungen

$$\alpha_i \geq 0 \quad \forall i = 1, \dots, N \quad \text{und} \quad \sum_{i=1}^N \alpha_i y_i = 0$$



Stützvektoren

Die Lösung $\vec{\alpha}^*$ des dualen Problems

$$L_D(\vec{\alpha}) = \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N y_i y_j \alpha_i \alpha_j \langle \vec{x}_i, \vec{x}_j \rangle$$

muss die KKT-Bedingungen erfüllen, d.h. es gilt unter anderem

$$\alpha_i \left(y_i \left(\langle \vec{x}_i, \vec{\beta} \rangle + \beta_0 \right) - 1 \right) = 0 \quad \forall i = 1, \dots, N$$

$\vec{\alpha}^*$ enthält für jedes Beispiel \vec{x}_i genau ein α_i mit

$\alpha_i = 0$, falls \vec{x}_i im richtigen Halbraum liegt

$\alpha_i > 0$, falls \vec{x}_i auf der Hyperebene H_1 oder H_2 liegt

Ein Beispiel \vec{x}_i mit $\alpha_i > 0$ heißt Stützvektor.

Optimale Hyperebene

Haben wir das optimale $\vec{\alpha}^*$ bestimmt, erhalten wir unsere optimale Hyperebene:

Nach (6) gilt

$$\vec{\beta} = \sum \alpha_i y_i \vec{x}_i$$

d.h. der optimale Normalenvektor $\vec{\beta}$ ist eine Linearkombination von Stützvektoren.

Um β_0 zu bestimmen können wir

$$\alpha_i \left(y_i \left(\langle \vec{x}_i, \vec{\beta} \rangle + \beta_0 \right) - 1 \right) = 0$$

für ein beliebiges i und unser berechnetes $\vec{\beta}$ nutzen.

Berechnung der α_i ?

Das prinzipielle Vorgehen ist bei der SVM wie bei anderen Lernverfahren auch:

- Parametrisierung der Modelle, hier über Umwege durch $\vec{\alpha}$
- Festlegung eines Optimalitätskriteriums, hier: **Maximum Margin**
- Formulierung als Optimierungsproblem

Das finale Optimierungsproblem läßt sich mit unterschiedlichen Ansätzen lösen

- Numerische Verfahren (*quadratic problem solver*)
- *Sequential Minimal Optimization* (SMO, [J. C. Platt, 1998])
- Evolutionäre Algorithmen (EvoSVM, [I. Mierswa, 2006])

Zusammenfassung der Lagrange-Optimierung für SVM

Das Lagrange-Optimierungs-Problem (5) ist definiert als:

$$L_P = \frac{1}{2} \|\vec{\beta}\|^2 - \sum_{i=1}^N \alpha_i \left[y_i (\langle \vec{x}_i, \vec{\beta} \rangle + \beta_0) - 1 \right]$$

mit den *Lagrange-Multiplikatoren* $\vec{\alpha}_i \geq 0$.

Notwendige Bedingung für ein Minimum liefern die Ableitungen nach $\vec{\beta}$ und β_0

$$\frac{\partial L_P}{\partial \vec{\beta}} = \vec{\beta} - \sum_{i=1}^N \alpha_i y_i \vec{x}_i \quad \text{und} \quad \frac{\partial L_P}{\partial \beta_0} = \sum_{i=1}^N \alpha_i y_i$$

Diese führen zum *dualen Problem* (9)

$$L_D = \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{i'=1}^N \alpha_i \alpha_{i'} y_i y_{i'} \langle \vec{x}_i, \vec{x}_{i'} \rangle$$

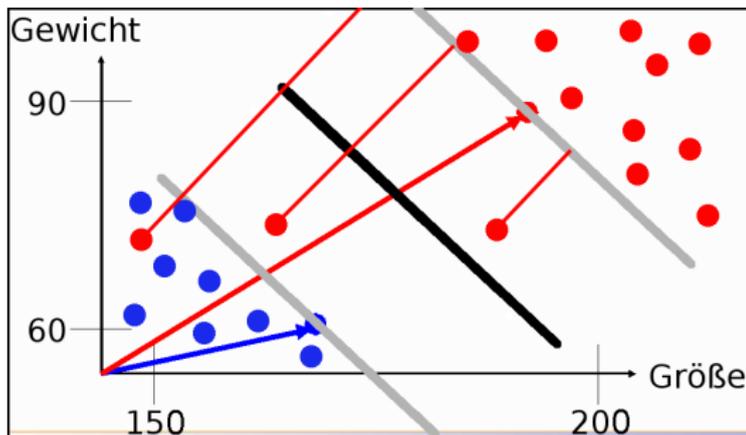


Was wissen wir jetzt?

- Maximieren des Margins einer Hyperebene ergibt eine eindeutige Festlegung der optimalen trennenden Hyperebene.
- Dazu minimieren wir die Länge des Normalenvektors $\vec{\beta}$
 - Formulierung als Lagrange-Funktion
 - Formulierung als duales Optimierungsproblem
- Das Lernergebnis ist eine Linearkombination von Stützvektoren.
- Mit den Beispielen müssen wir nur noch das Skalarprodukt rechnen.

SVM mit Ausnahmen

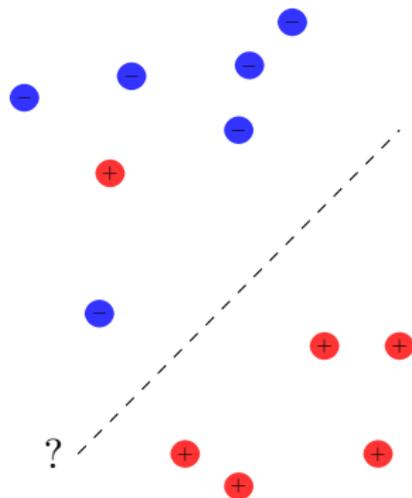
- Was passiert, wenn die Beispiele nicht komplett trennbar sind?



Nicht linear trennbare Daten

In der Praxis sind linear trennbare Daten selten:

- 1. Ansatz: Entferne eine minimale Menge von Datenpunkten, so dass die Daten linear trennbar werden (minimale Fehlklassifikation).
- Problem: Algorithmus wird exponentiell.



SVM mit Ausnahmen

Ein anderer Ansatz basiert wieder auf einer Relaxation:

- Punkte, die nicht am Rand oder auf der richtigen Seite der Ebene liegen, bekommen einen Strafterm $\xi_j > 0$.
- Korrekt klassifizierte Punkte erhalten eine Variable $\xi_j = 0$.

Dies führt zu folgenden Minimierungsproblem

$$\frac{1}{2} \|\vec{\beta}\|^2 + C \sum_{j=1}^N \xi_j \quad \text{für ein festes } C \in \mathbb{R}_{>0} \quad (10)$$

Daraus folgt insbesondere

$$0 \leq \alpha_i \leq C$$

Weich trennende Hyperebene

Relaxiertes Optimierungsproblem

Sei $C \in \mathbb{R}$ mit $C > 0$ fest. Minimiere

$$\|\vec{\beta}\|^2 + C \sum_{i=1}^N \xi_i$$

unter den Nebenbedingungen

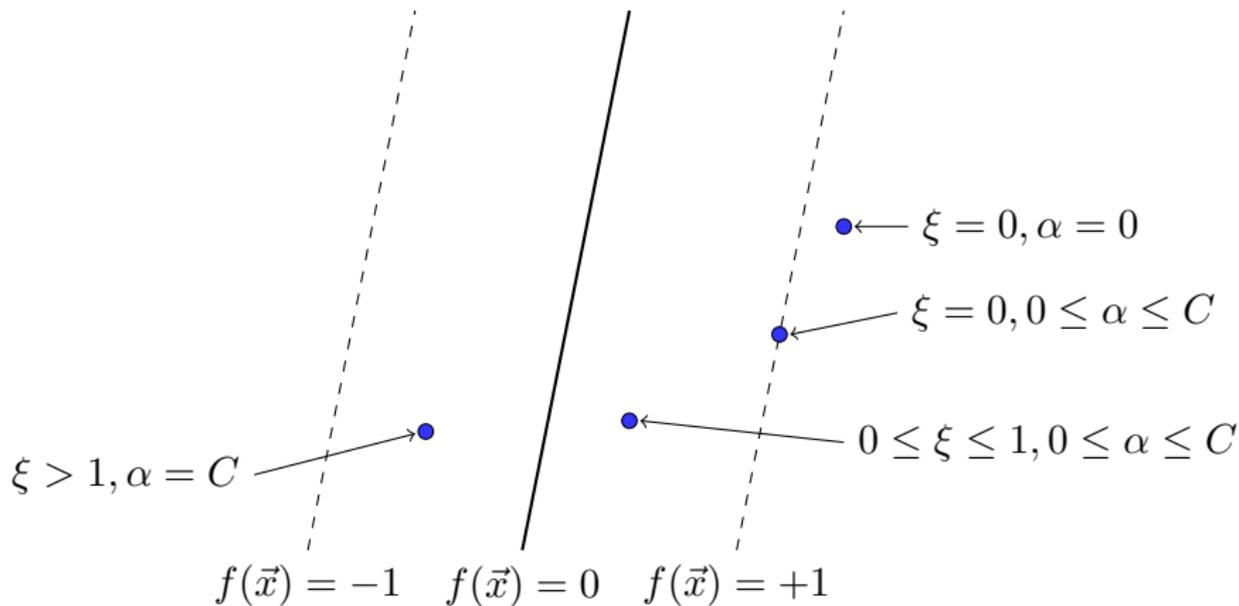
$$\langle \vec{x}_i, \vec{\beta} \rangle + \beta_0 \geq +1 - \xi_i \quad \text{für } \vec{y}_i = +1$$

$$\langle \vec{x}_i, \vec{\beta} \rangle + \beta_0 \leq -1 + \xi_i \quad \text{für } \vec{y}_i = -1$$

Durch Umformung erhalten wir wieder Bedingungen für die Lagrange-Optimierung:

$$y_i (\langle \vec{x}_i, \vec{\beta} \rangle + \beta_0) \geq 1 - \xi_i \quad \forall i = 1, \dots, N$$

Bedeutung von ξ und $\vec{\alpha}$



Beispiele \vec{x}_i mit $\alpha_i > 0$ sind Stützvektoren.

Wo sind wir?

- Maximieren der Breite einer separierenden Hyperebene (**maximum margin method**) ergibt eindeutige, optimale trennende Hyperebene.
 - Das haben wir heute in der Theorie für linear separierbare Beispielmengen und mit weicher Trennung gesehen – wie es praktisch geht, sehen wir nächstes Mal.
 - Die Grundlagen waren die selben wie bei den **linearen Modellen**.
- Transformation des Datenraums durch Kernfunktion behandelt Nichtlinearität.
 - Das kam nur einmal am Rande vor. Wir sehen es nächstes Mal genauer.
 - Es baut auf die Behandlung der Nichtlinearität durch die **Basisexpansion** auf.
- Strukturelle Risikominimierung minimiert nicht nur den Fehler, sondern auch die Komplexität des Modells. Später!