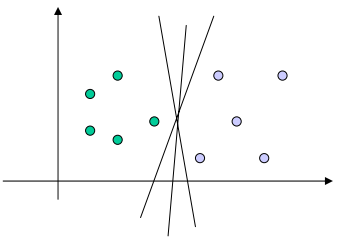
Linear separierbare Klassen

- Daten in $\mathbb{R}^2$



  - Allgemeine Hyperebene $\langle \mathbf{w}, \mathbf{x} \rangle + b = 0$
  - Klassifikation via $f(x) = \mathrm{sgn}(\langle \mathbf{w}, \mathbf{x} \rangle + b)$
  - z.B. Rosenblatt´s Perceptron (1956)

    Iteratives Lernen, Korrektur nach jeder Fehlklassifikation

    ->keine Eindeutigkeit der Lösung

Optimale Hyperebene: Maximierung des Randes

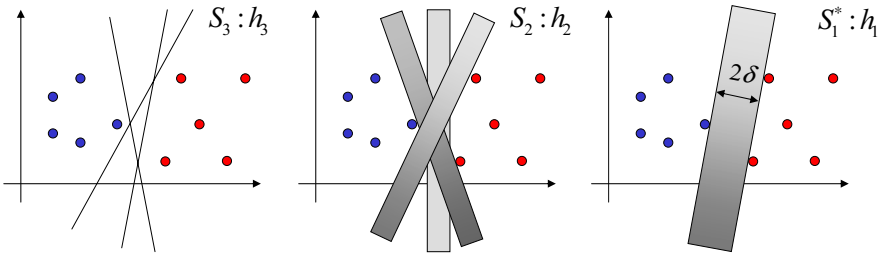
- Geht man von einem linear separierbarem Zweiklassenproblem aus, so erreichen alle Geraden, welche beide Klassen trennen einen empirischen Fehler Null
- Die Konfidenz wird minimiert durch ein Polynom minimaler VC-Dimension, nämlich eine Hyperebene
- Die VC-Dimension kann weiter abgesenkt werden durch „breite Hyperebenen“ (large margin hyperplanes)

• Trennende Hyperebene mit maximalem Rand	⟺	• Trennende Hyperebene mit minimaler VC-Dimension

- Dieses Ergebis ist plausibel. Bei konstanter Intraklassenstreuung, wächst die Klassifikationssicherheit mit wachsendem Interklassenabstand.
- oder: bei konstant gehaltenem Interklassenabstand (z.Bsp. 0) muss die Intraklassenstreung maxmial zulässig werden

“Large-margin“-Klassifikator

– Hyperebene mit größtem Rand [Vap63]



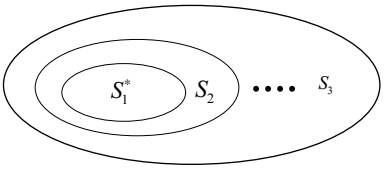
hohe VC-Dimension

in $\mathbb{R}^N : N+1$

mittlere VC-Dimension
Variabilität wird kleiner!

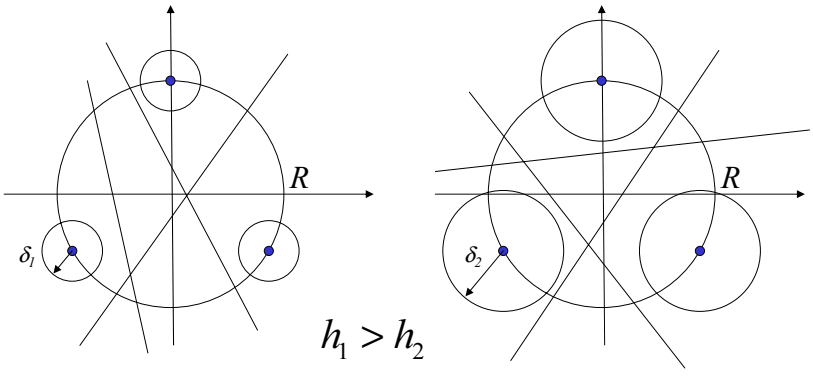
kleinste VC-Dimension
mit maximaler Breite
Variabilität gleich Null

- Anschaulich sinnvoll
- theoretisch begründet
- Lösung abhängig von wenigen Daten:
  =>“Support-Vektoren“



$h_1 < h_2 < \ldots < h_3$

VC-Dimension von „breiten“ Hyperebenen



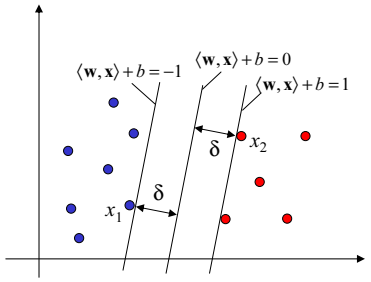
$h_1 > h_2$

Die VC-Dimension h von Hyperebenen mit einem Mindestabstand δ von den zu trennenden Punkten ist begrenzt durch:

$$VC\dim \le \frac{R^2}{\delta^2} + 1$$

=> grosser Margin, kleine VCdim

Formalisierung



Die Daten werden korrekt klassifiziert, falls:

$$y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) > 0$$

Dieser Ausdruck ist invariant gegenüber einer positiven Reskalierung:

$$y_i(\langle a\mathbf{w}, \mathbf{x}_i \rangle + ab) > 0$$

Einführung von kanonischen Hyperebenen:

$$\begin{cases} \langle \mathbf{w}, \mathbf{x}_1 \rangle + b = -1 & \text{für die blaue Klasse} \\ \langle \mathbf{w}, \mathbf{x}_2 \rangle + b = +1 & \text{für die rote Klasse} \end{cases}$$

Der Abstand zwischen den kanonischen Hyperebenen ergibt sich durch Projektion von $\mathbf{x}_1$-$\mathbf{x}_2$ auf den Normalenvektor w/||w||:

$$\frac{+(\langle \mathbf{w}, \mathbf{x}_1 \rangle + b = +1) \quad -(\langle \mathbf{w}, \mathbf{x}_2 \rangle + b = -1)}{\langle \mathbf{w}, (\mathbf{x}_1 - \mathbf{x}_2) \rangle = 2} \Rightarrow \underbrace{\langle \frac{\mathbf{w}}{\|\mathbf{w}\|}, (\mathbf{x}_1 - \mathbf{x}_2) \rangle}_{2\delta} = \frac{2}{\|\mathbf{w}\|}$$

$$\Rightarrow \boxed{\delta = 1/\|\mathbf{w}\|}$$

Die Maximierung von δ ist gleichwertig mit einer Minierung von $\|\mathbf{w}\|^2$ =>

Optimierungsproblem

Primales OP:

$$\text{minimiere} \quad J(\mathbf{w}, b) = \tfrac{1}{2}\|\mathbf{w}\|^2 = \tfrac{1}{2}\langle \mathbf{w}, \mathbf{w} \rangle$$
$$\text{unter der N.B.:} \quad \forall i \left[y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) \geq 1 \right]$$

Einführung einer Lagrangefunktion:

$$L(\mathbf{w}, b, \alpha) = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{l} \alpha_i [y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) - 1]$$

mit: $\alpha_i \geq 0$

Die partiellen Ableitungen nach w_i, b und den α_i führen nach Einsetzen in das primale OP auf das äquivalente:

Wolf-duale OP:

$$\text{maximiere} \quad L'(\mathbf{w}, b, \boldsymbol{\alpha}) = \sum_{i=1}^{l} \alpha_i - \tfrac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} y_i y_j \alpha_i \alpha_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle$$
$$\text{unter der N.B.:} \quad \alpha_i \geq 0 \ \text{ und } \ \sum_{i=1}^{l} y_i \alpha_i = 0$$

Dies ist ein positiv semidefinites Problem, welches mit Hilfe der konvexen quadratischen Programmierung numerisch iterativ gelöst werden kann!

Lösung:

– Lösung des dualen Problems liefert eindeutig gewünschte Hyperebene

$$\mathbf{w}^* = \sum_{i=1}^{l} \alpha_i y_i \mathbf{x}_i = \sum_{\mathbf{x}_i \in SV} \alpha_i y_i \mathbf{x}_i$$

$$b^* = -\frac{\max_{y_i=-1}(\langle \mathbf{w}, \mathbf{x}_i \rangle) + \min_{y_i=1}(\langle \mathbf{w}, \mathbf{x}_i \rangle)}{2}$$

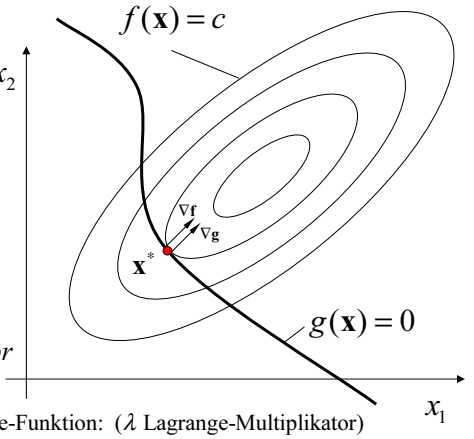
Klassifikation: $f(\mathbf{x}) = \text{sgn}(\langle \mathbf{w}^*, \mathbf{x} \rangle + b^*) = \text{sgn}(\sum_{\mathbf{x}_i \in SV} \alpha_i y_i \mathbf{x}_i) + b^*)$

Lösung nur abhängig von den Supportvektoren !!

Beobachtungen:

- Für die Support-Vektoren gilt: $0<\alpha_i<\infty$
- Für alle Beispiele ausserhalb des Randes ist $\alpha_i=0$
  -> Support-Vektoren, "sparse"-Darstellung der Lösung
- Eindeutigkeit der Ebene, *globales Optimum*!!

Optimierung von $f(\mathbf{x})$ unter gegebenen Nebenbedingungen $g(\mathbf{x})=0$ mit Hilfe des Lagrange-Ansatzes



$$\nabla \mathbf{f} = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \end{bmatrix}$$

Gradient = *Normalenvektor*

$$f(\mathbf{x}) \overset{!}{=} \min_{\mathbf{x}}$$

NB: $g(\mathbf{x}) = 0$

Eine Lösung für das Optimierungsproblem zu finden ist gleichbedeutend mit der Aufgabe, stationäre Punkte x* zu finden, in denen gilt:

$$\boxed{\nabla \mathbf{f} \parallel \nabla \mathbf{g}}$$

Aus der Lagrange-Funktion: (λ Lagrange-Multiplikator)

$$\boxed{L(\mathbf{x}, \lambda) = f(\mathbf{x}) + \lambda g(\mathbf{x})}$$

folgt mit der notwendigen Bedingung für ein Extremum:

$$\nabla \mathbf{L} = \frac{\partial L}{\partial \mathbf{x}} = \nabla \mathbf{f} + \lambda \nabla \mathbf{g} = \mathbf{0} \quad \Rightarrow \nabla \mathbf{f} = -\lambda \nabla \mathbf{g}$$

$\Rightarrow \boxed{\nabla \mathbf{f} \parallel \nabla \mathbf{g}}$ dies ist aber genau die Bedingung für einen stationären Punkt!

Dies ist nur eine notwendige Bedingung für ein lokales Optimum; es kann mehrere Lösungen geben!

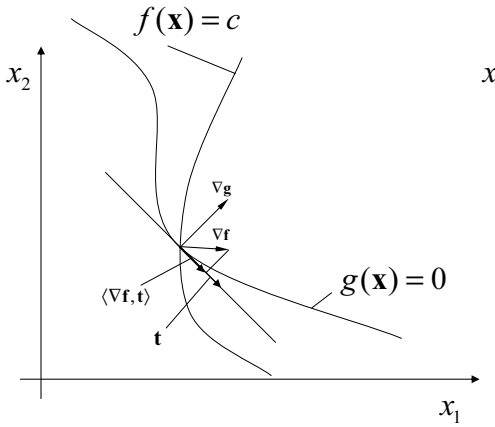
Der Gradient ist ein Normalenvektor an die Kurve $g(\mathbf{x})=0$

Betrachten wir die Taylorentwicklung von g bzgl. einer kleinen vektoriellen Störung $\boldsymbol{\varepsilon}$ an einer Stelle x auf der Kurve $g(\mathbf{x})=0$, so erhält man:

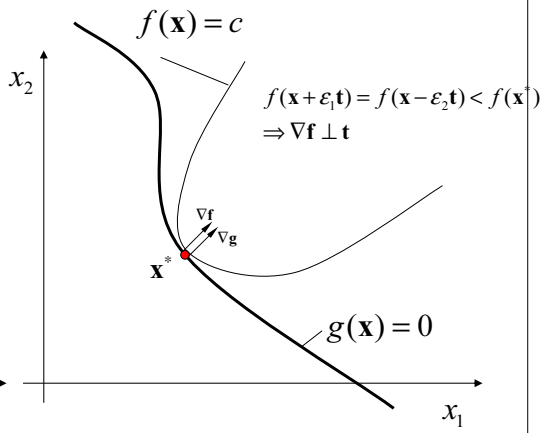
$$g(\mathbf{x}+\boldsymbol{\varepsilon}) = g(\mathbf{x}) + \boldsymbol{\varepsilon}^T \nabla \mathbf{g}$$

Bewegt sich die Störung $\boldsymbol{\varepsilon}$ entlang der Kurve, so gilt $g(\mathbf{x}+\boldsymbol{\varepsilon}) = g(\mathbf{x})$ und somit auch $\boldsymbol{\varepsilon}^T\nabla\mathbf{g}(\mathbf{x}) = 0$. Daraus erkennen wir, dass der Gradient senkrecht steht auf die Oberfläche $g(\mathbf{x})=0$ (Normalenvektor).

Bedingung für einen stationären Punkt $\mathbf{x}^*$



a) Verhältnisse an einem nichtstationären Punkt



b) Verhältnisse an einem stationären Punkt

a) Enthält die Projektion von $\nabla\mathbf{f}$ auf die Tangentenrichtung $\mathbf{t}$ einen von Null verschiedenen Beitrag, so kann das Optimierungskriterium durch Bewegung in diese Richtung entlang der Kurve der NB verbessert werden! => kein stationärer Punkt!

b) An einem stationären Punkt wird das Gütekriterium f(x) in beiden tangentialen Richtungen verschlechtert. $\nabla\mathbf{f}$ steht senkrecht auf $\mathbf{t}$.

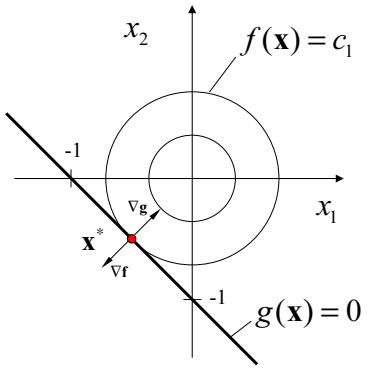
Beispiel:

1) $f(\mathbf{x}) = x_1^2 + x_2^2$

2) NB: $g(\mathbf{x}) = x_1 + x_2 + 1 = 0$

$$\nabla \mathbf{f} = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \end{bmatrix} = 2 \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$\nabla \mathbf{g} = \begin{bmatrix} \frac{\partial g}{\partial x_1} \\ \frac{\partial g}{\partial x_2} \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$



$\nabla \mathbf{f}^* \parallel \nabla \mathbf{g}^*$

$L(\mathbf{x}, \lambda) = f(\mathbf{x}) + \lambda g(\mathbf{x})$

notw. Bed. für ein Optimum:

1) $\nabla \mathbf{L} = \frac{\partial L}{\partial \mathbf{x}} = \nabla \mathbf{f} + \lambda \nabla \mathbf{g} = \mathbf{0}$

2) $\frac{\partial L}{\partial \lambda} = g(\mathbf{x}) = 0$

$\Rightarrow$

1) $2x_1 + \lambda = 0$

$2x_2 + \lambda = 0$

2) $x_1 + x_2 + 1 = 0$

$\Rightarrow \lambda^* = 1$ und $\mathbf{x}^* = -\begin{bmatrix} ½ \\ ½ \end{bmatrix}$

$$\Rightarrow \nabla \mathbf{f}(\mathbf{x}^*) = \begin{bmatrix} -1 \\ -1 \end{bmatrix} = -\lambda^* \nabla \mathbf{g}(\mathbf{x}^*) = -1 \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

Demos mit MATLAB

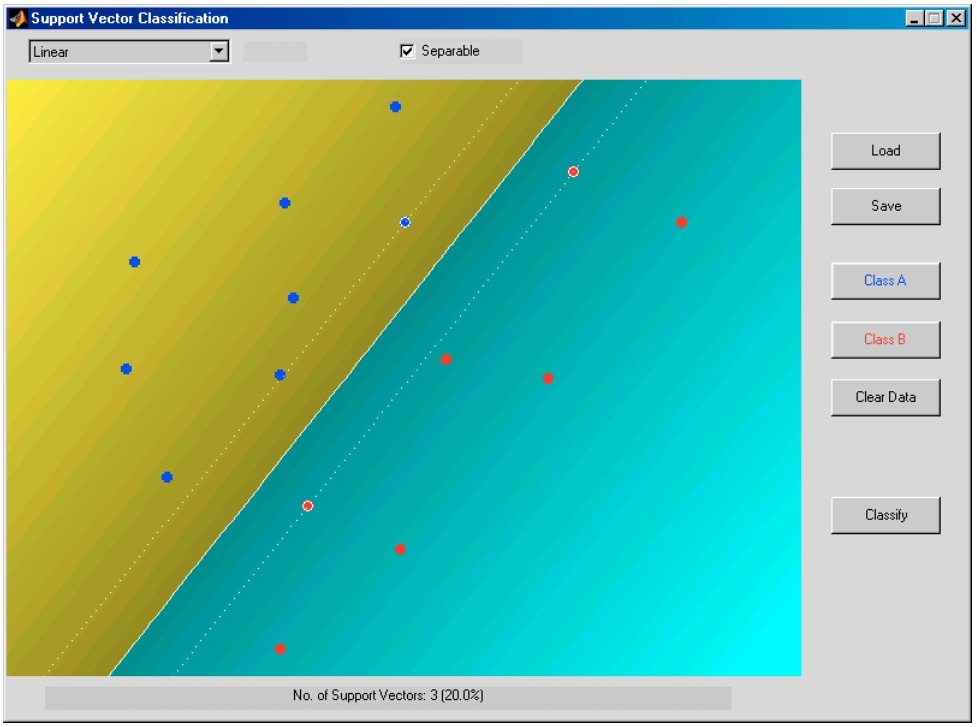
(svmmatlab, uiclass.m)

- Linearer Klassifikator
  - Harter Rand, Problem separierbar
  - Harter kleiner Rand wegen Ausreißer, Problem separierbar, aber schlechte Generalisierung => emp. Fehler vergrößern mit Gewinn bei Generalisierung => weichen Rand einführen
- Nichtlinearer Klassifikator
  - Für nichtlineare Probleme muss VC-Dimension der trennenden Hyperflächen und damit ihre Kapazität vergrößert werden!
  - Lineare Separierung eines quadratischen Problems mit weichem Rand
  - Polynomiale Separierung (p=2), harter Rand
  - Polynomiale Separierung (p=4), harter Rand
  - Bananenshape mit Polynomkernen
  - Bananenshape mit Gauss-Radialbasisfunktionen

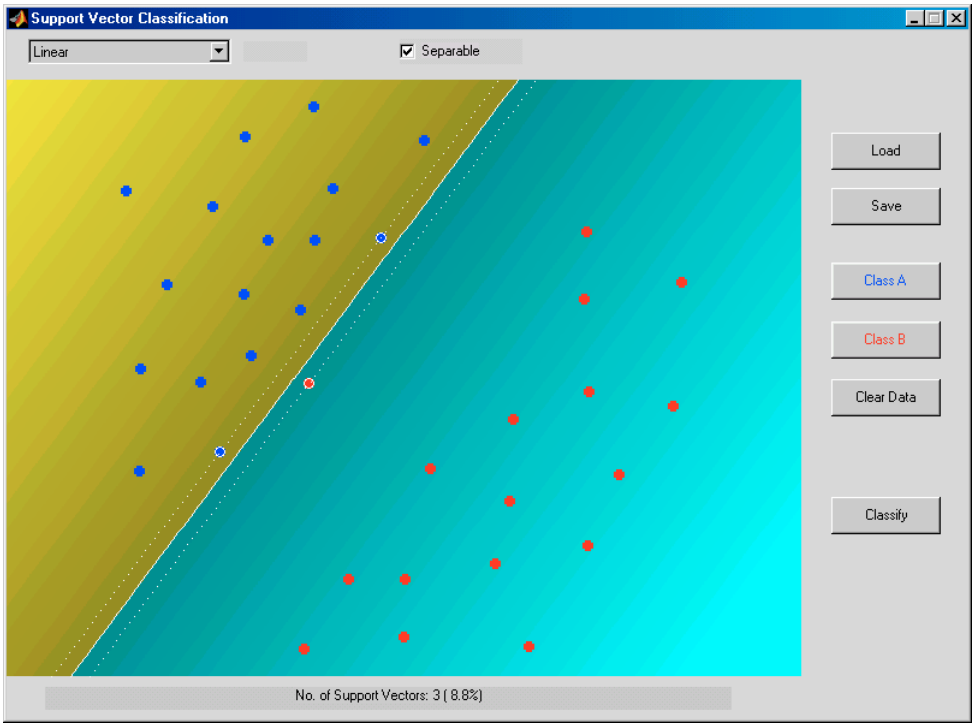
Start der Matlab-Demo

matlab-SVM.bat

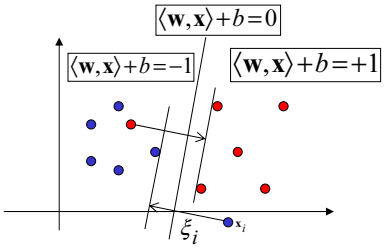
Linear separierbare Klassen; harter Rand



Linear separiebare Klassen; harter, kleiner Rand, schlechte Generalisierung



Nichtseparabler Fall



Bestrafen von Randverletzungen via "slack"-Variablen [Smith68] ->"Soft-Margin" SVM

minimieren von: $\|\mathbf{w}\|^2 + C\sum_{i=1}^{l}\xi_i$

mit: $y_i(\langle \mathbf{w}, \mathbf{x}_i\rangle + b) \geq 1 - \xi_i$ und $\xi_i \geq 0$

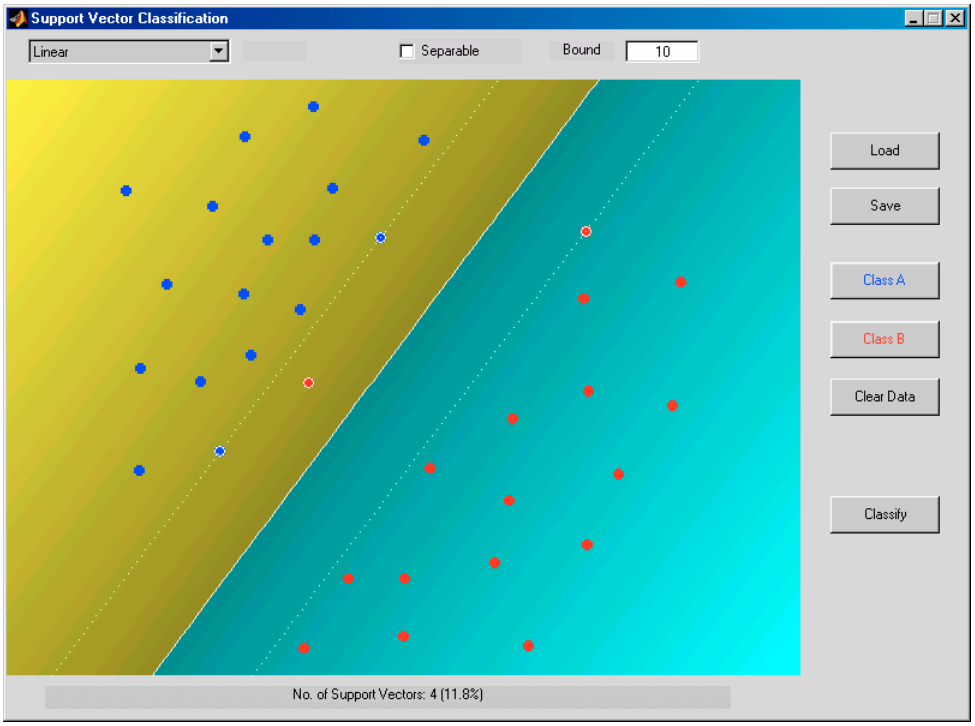
Gründe für diese Verallgemeinerung:

- Lösung nicht existent mit bisherigem Ansatz mit hartem Rand
- Verbesserung der Generalisierung bei Ausreißern in der Randzone

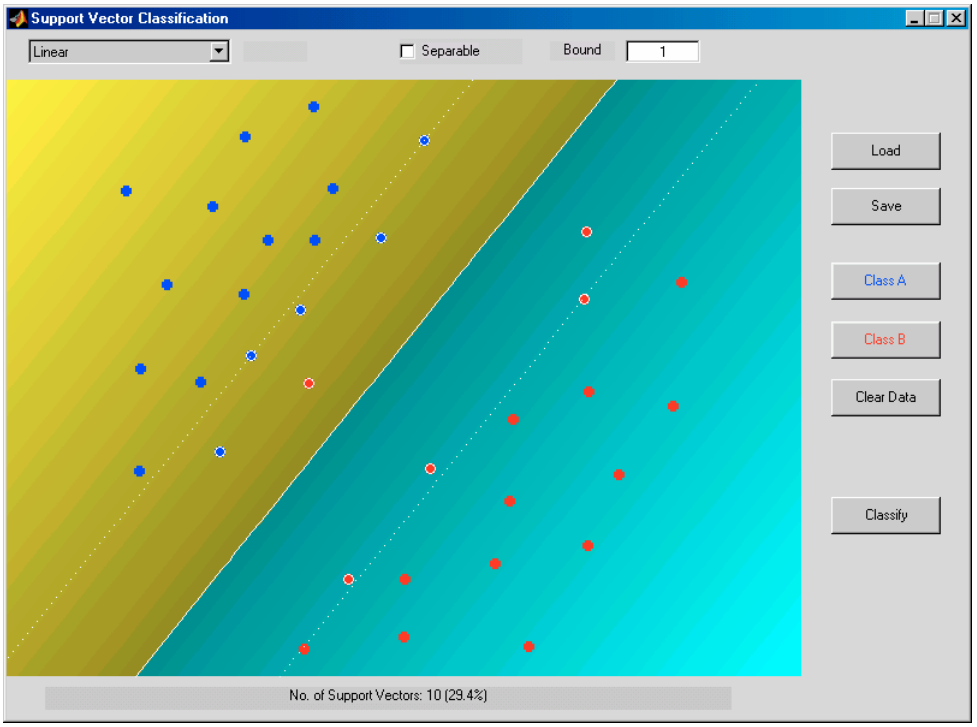
Fallunterscheidung:

- $0<\alpha_i<C$ ⇔ SV mit $\xi_i=0$
- $\alpha_i=C$ ⇔ SV mit $\xi_i>0$
- $\alpha_i=0$ ⇔ für die restlichen Vektoren $\mathbf{x}_i$

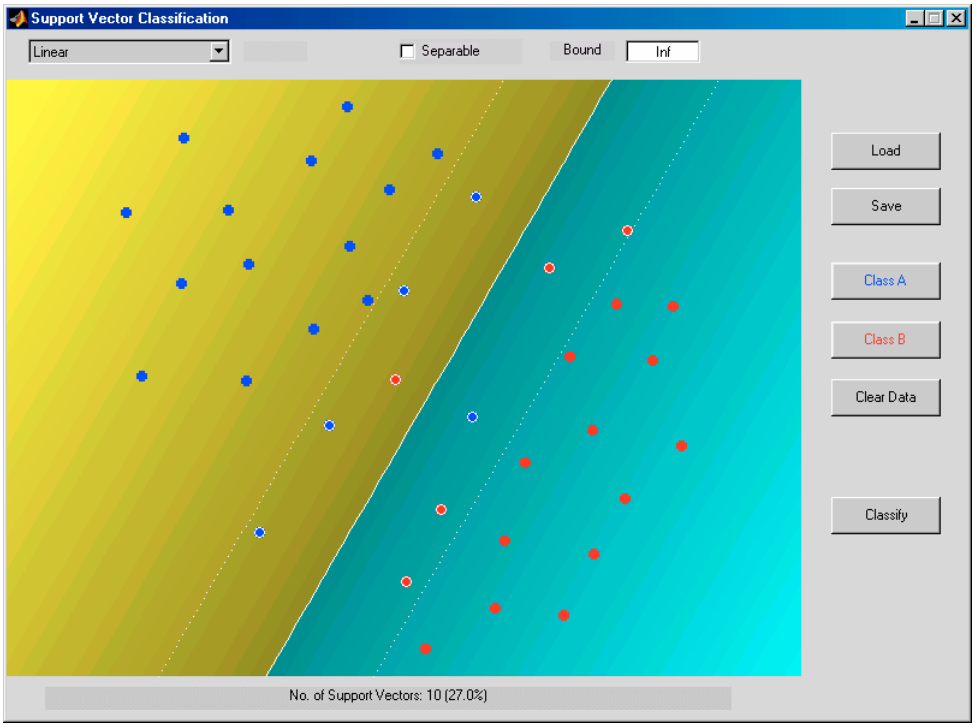
Linear sep. Klassen; weicher, breiter Rand => R_{emp} größer, gute Generalisierung



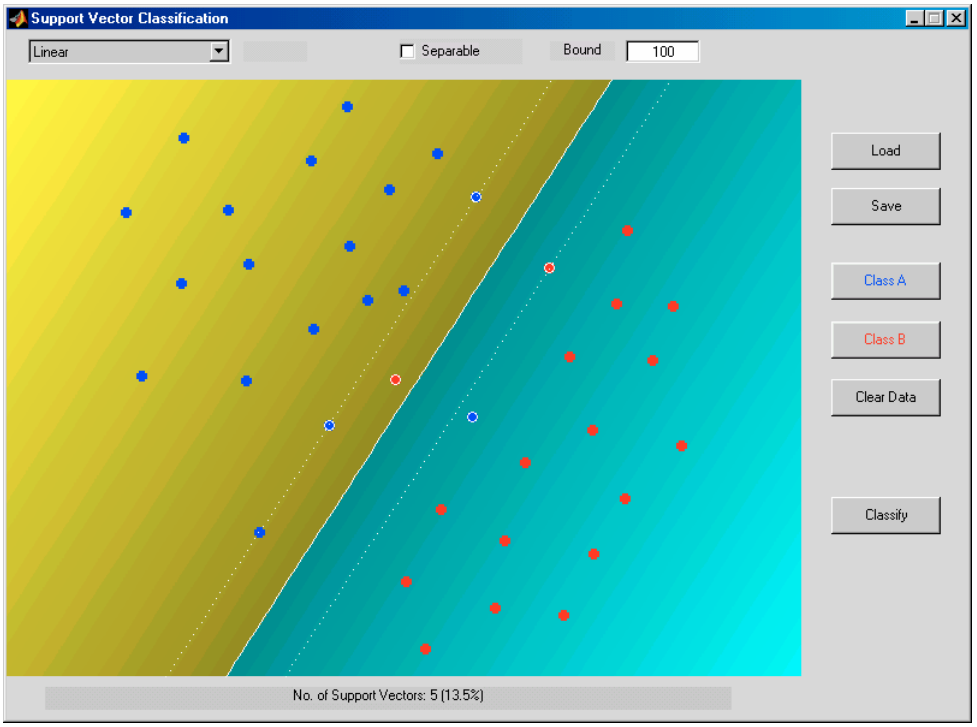
Linear sep. Klassen; weicher, breiter Rand => R_{emp} größer, gute Generalisierung



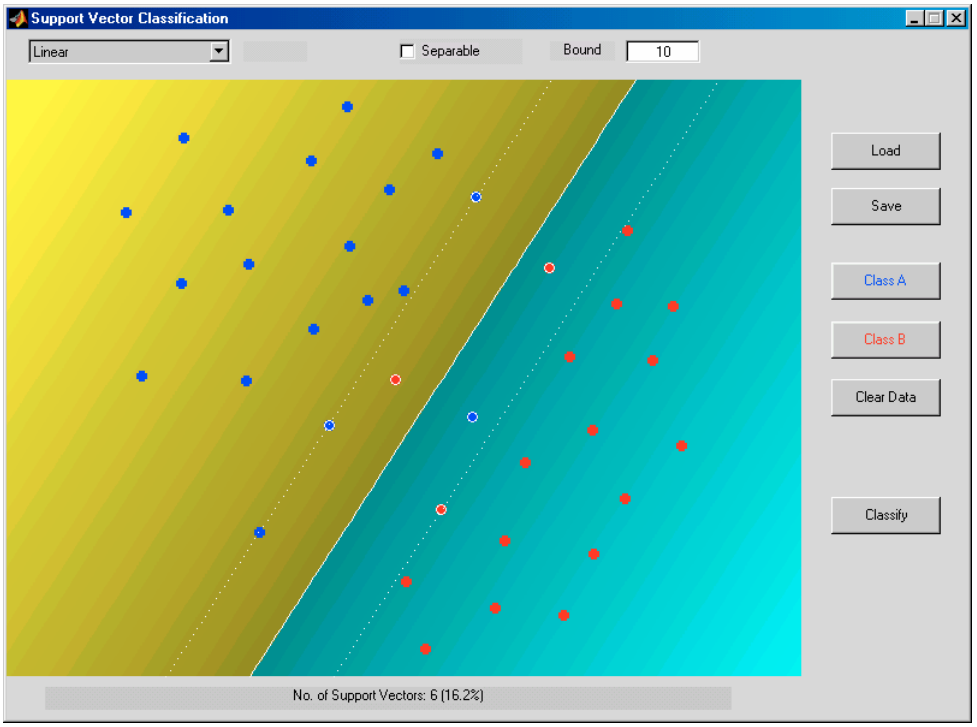
Linear separiebare Klassen; harter Rand



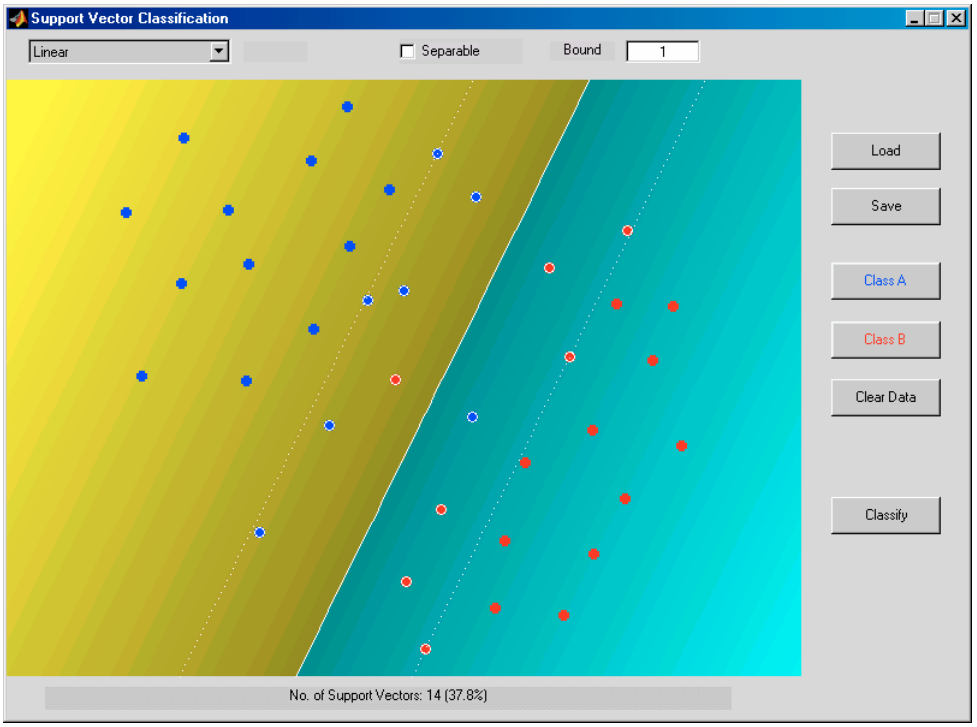
Linear separiebare Klassen; harter Rand



Linear separiebare Klassen; harter Rand

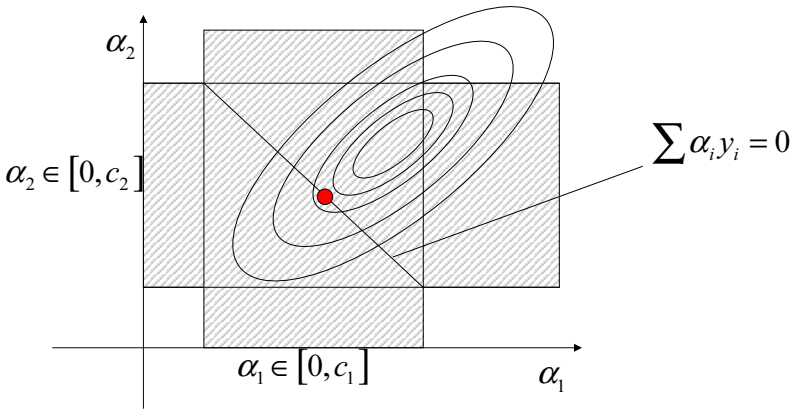


Linear separiebare Klassen; harter Rand



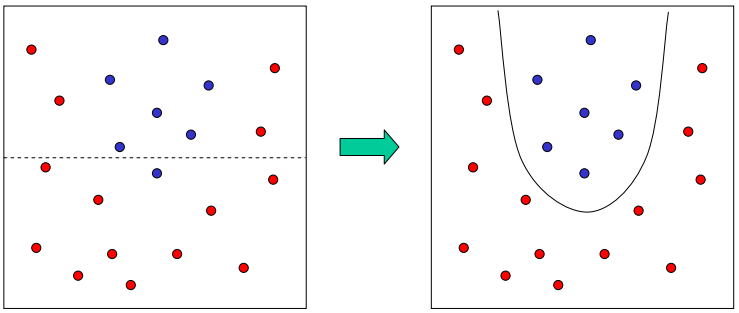
Globales, eindeutiges Optimum

- Optimierung eines quadratischen Problems in **w** (konvex)
- unter linearen Nebenbedingungen
- Lineare Nebenbedingungen miteinander geschnitten ergeben konvexes Gebiet; dieses geschnitten mit quadratischer Form => ergibt wiederum konvexes Gebiet => eindeutiges Minimum!



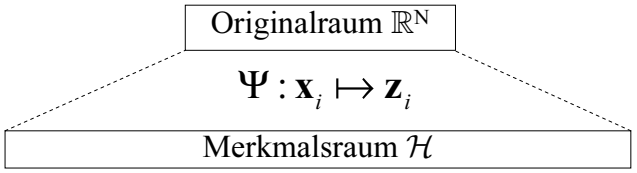
Nichtlineare Probleme

- manche Probleme haben nichtlineare Klassengrenzen
- Hyperebenen erreichen keine zufriedenstellende Genauigkeit



Erweiterung des Hypothesenraumes

Idee: Finde Hyperebene im höherdimensionalen Merkmalsraum

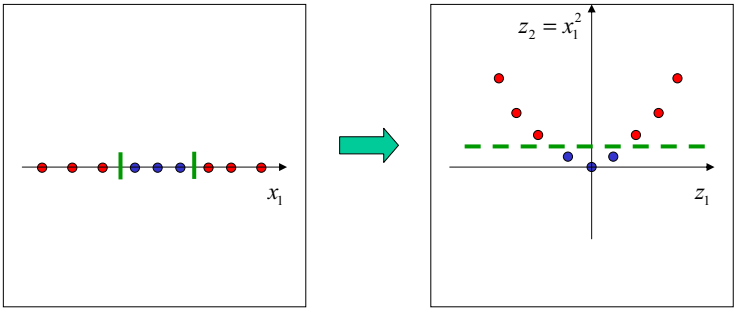


Die trennende Hyperebene im Merkmalsraum ist eine nichtlineare Trennfläche im Originalraum (siehe XOR-Problem mit Polynomklassifikator)

Nichtlineare Probleme

- Eindimensionaler Originalraum: x_1
- Zweidimensionaler Merkmalsraum:

$$\Psi(x_1) = \mathbf{z}^T = \left[z_1 = x_1, z_2 = x_1^2 \right]^T$$



linear nicht separierbar

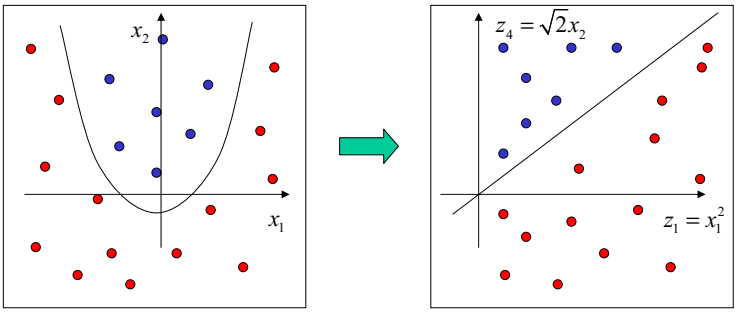
lineare Separation

Nichtlineare Probleme

- Originalraum:

  $\mathbf{x}=(x_1,x_2)$ (zweidimensional)
- Merkmalsraum:

$$\Psi(\mathbf{x}) = \mathbf{z}^T = \left[z_1 = x_1^2, z_2 = x_2^2, z_3 = \sqrt{2}x_1, z_4 = \sqrt{2}x_2, z_5 = \sqrt{2}x_1x_2, z_6 = 1 \right]^T$$



- $x_2 > x_1^2$
- $x_2 < x_1^2$

nichtlineare Separation

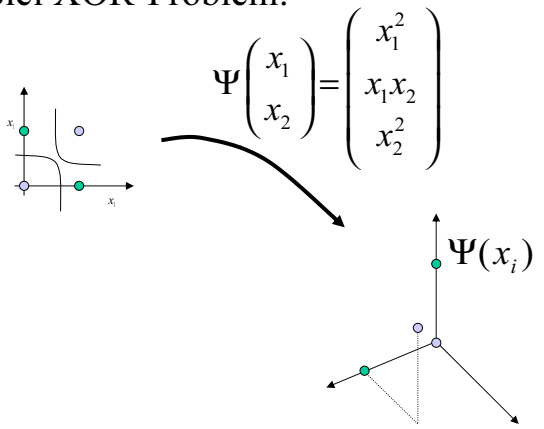
- $z_4 > \sqrt{2}z_1$
- $z_4 < \sqrt{2}z_1$

lineare Separation

Nichtlineare Erweiterung

Nichtlineare Abbildung vorschalten $\Psi(x): \mathbb{R}^N \to \mathcal{H}$

- Beispiel XOR-Problem:



– Effekt:
  Steigerung der Separabilität
  Trennfläche im Ursprungsraum nichtlinear
– Fragen:
  1. Optimalität der Hyperebene?
  2. Hoher Rechenaufwand in hochdimensionalen Räumen?
– Zu 1: Optimalität bleibt erhalten, erneut positiv semidefinite Form, da in der zu optimierenden Funktion die gleichen Skalarprodukte auftauchen, nur in einem neuen Raum $\mathcal{H}$

Der Trick mit Kernfunktionen

Problem: Sehr hohe Dimension des Merkmalraumes! Polynome p-ten Grades über der Dimension N des Originalraums führen zu $O(N^p)$ Dimensionen im Merkmalsraum!

Lösung: Im dualen OP tauchen nur Skalarprodukte $<\mathbf{x}_i,\mathbf{x}_j>$ auf. Im korrespondierenden Problem im Merkmalsraum tauchen dann ebenfalls nur Skalarprodukte in $<\Psi(\mathbf{x}_i), \Psi(\mathbf{x}_j)>$ auf. Diese müssen nicht explizit ausgerechnet werden, sondern können mit reduzierter Komplexität mit Kernfunktionen ausgedrückt werden:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \langle \Psi(\mathbf{x}_i), \Psi(\mathbf{x}_j) \rangle$$

Beispiel:

Für $\Psi(\mathbf{x}) = \mathbf{z}^T = \left[z_1 = x_1^2, z_2 = x_2^2, z_3 = \sqrt{2}x_1, z_4 = \sqrt{2}x_2, z_5 = \sqrt{2}x_1x_2, z_6 = 1 \right]^T$

berechnet $K(\mathbf{x}_i, \mathbf{x}_j) = (\langle \mathbf{x}_i, \mathbf{x}_j \rangle + 1)^2 = \langle \Psi(\mathbf{x}_i), \Psi(\mathbf{x}_j) \rangle$

das Skalarprodukt im Merkmalsraum.

Häufig verwendete Kernfunktionen

Polynom-Kerne $K(\mathbf{x}_i, \mathbf{x}_j) = (\langle \mathbf{x}_i, \mathbf{x}_j \rangle + 1)^2$

Gauss-Kerne $K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\left\| \mathbf{x}_i - \mathbf{x}_j \right\|^2 / (2\sigma^2) \right)$

Sigmoid-Kerne $K(\mathbf{x}_i, \mathbf{x}_j) = \tanh(\kappa \langle \mathbf{x}_i, \mathbf{x}_j \rangle + \theta)$

Die resultierenden Klassifikatoren sind vergleichbar mit Polynomklassifikatoren, radialen Basisfunkten und mit Neuronalen Netzen (sie werden allerdings anders motiviert).

Allgemeine Anforderung: Mercer´s Bedingung. Sie garantiert, dass eine bestimmte Kernfunktion tatsächlich auch ein Skalarprodukt in irgendeinem Raum ist, aber sie erklärt nicht wie das dazugehörige Abbildung Φ aussieht und wie der Raum $\mathcal{H}$ beschaffen ist.

Ausserdem: Linearkombinationen von gültigen Kernen liefern neue Kerne

Das Theorem von Mercer

Es existiert eine Abbildung Φ und eine Entwicklung .

$$K(\mathbf{x}_i, \mathbf{x}_j) = \langle \Psi(\mathbf{x}_i), \Psi(\mathbf{x}_j) \rangle$$

genau dann, wenn für ein beliebiges $g(\mathbf{x})$ mit

$$\int g(\mathbf{x})^2 \, d\mathbf{x} < \infty$$

gilt:

$$\int K(\mathbf{x}_i, \mathbf{x}_j) g(\mathbf{x}_i) g(\mathbf{x}_j) d\mathbf{x}_i d\mathbf{x}_j \geq 0$$

Es gibt allerdings auch Fälle, wo Kernfunktionen die Mercer-Bedingung nicht erfüllen, aber für einen bestimmten Trainingsdatensatz zu einer positiv semidefiniten Hesse-Matrix führen und damit zu einem globalen Optimum konvergieren.