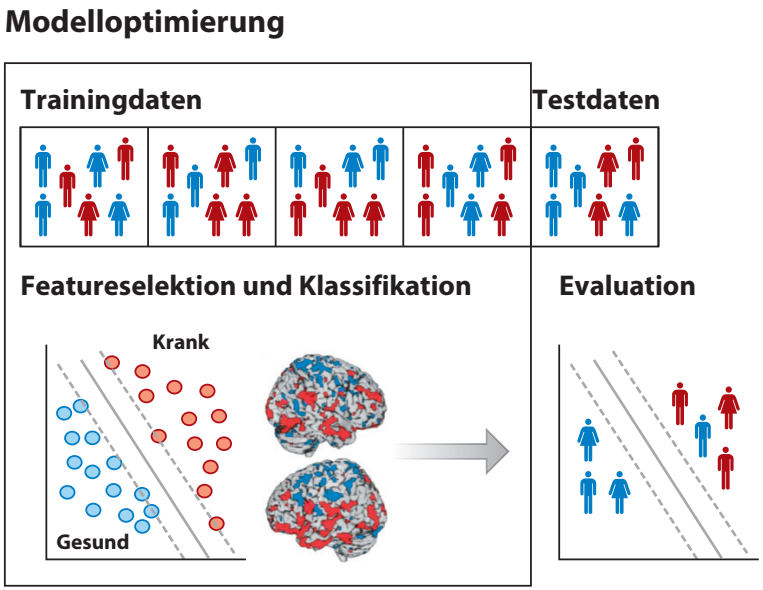
Multivariate Datenanalyse

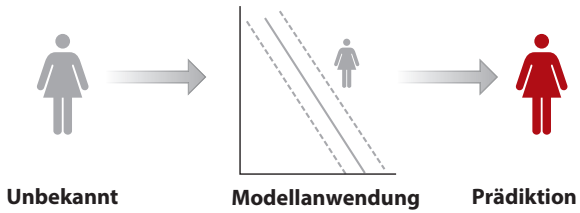
MSc Psychologie WiSe 2021/22

Prof. Dr. Dirk Ostwald

(8) Support Vektor Maschinen

Struktur der Prädiktiven Modellierung





nach Dwyer, Falkai, and Koutsouleris (2018)

Rhethorik der Prädiktiven Modellierung

Daten	Trainingsdaten und Testdaten
Statistisches Modell	Modell, Machine Learning Algorithmus
Schätzen von Parametern	Trainieren des Modells, Lernen von Parametern, Supervised Learning

Definition (Binärer Klassifikationstrainingdatensatz)

Ein *binärer Klassifikationsdatensatz*

$$\mathcal{D} := \{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), ..., (x^{(n)}, y^{(n)})\} \tag{1}$$

ist eine Menge von n *Trainingsdatenpunkten*

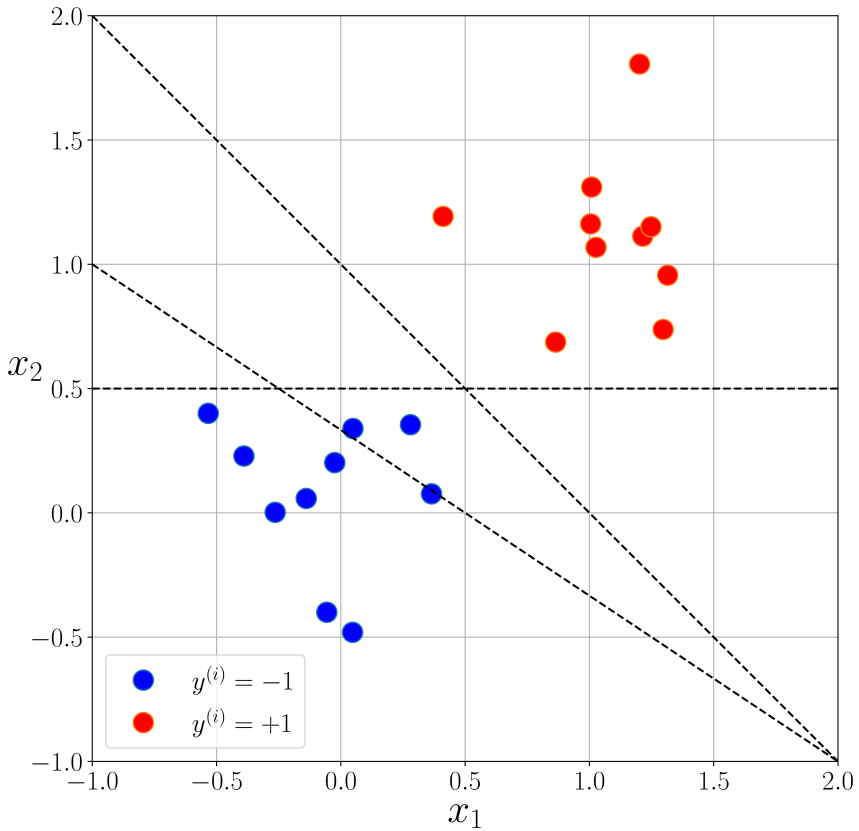
$$(x^{(i)}, y^{(i)}) \text{ mit } x^{(i)} \in \mathbb{R}^m \text{ und } y^{(i)} \in \{-1, 1\} \text{ für } i = 1, ..., n, \tag{2}$$

wobei $x^{(i)}$ m*-dimensional Featurevektor* und $y^{(i)}$ *Label* genannt wird

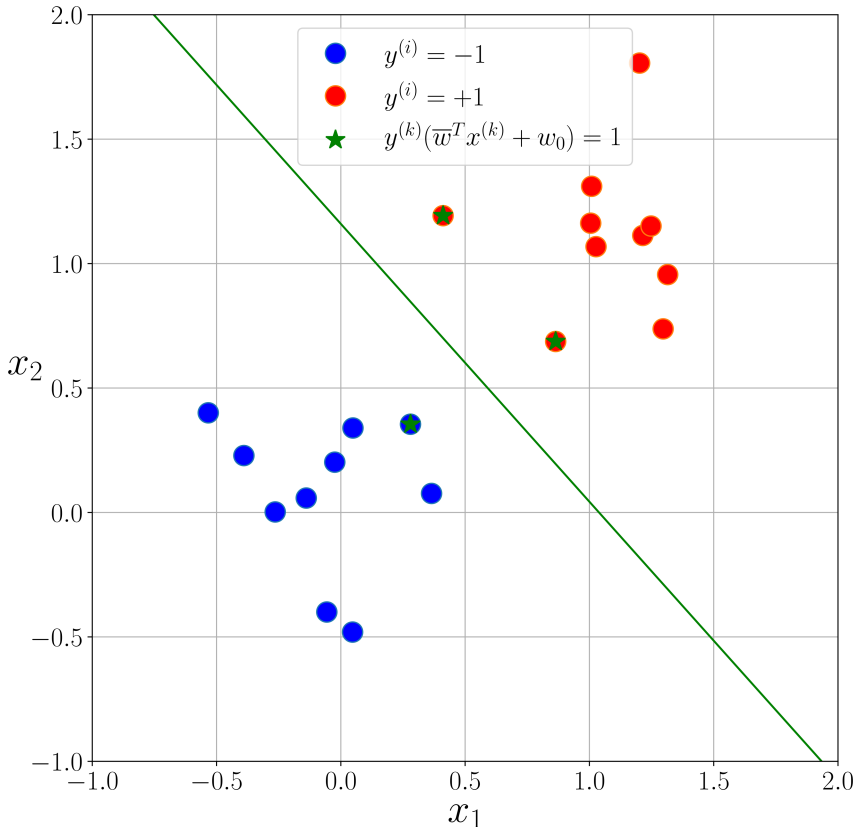
Bemerkung

- $y^{(i)} \in \{-1, 1\}$ bezeichnet die Klassenzugehörigkeit des Featurevektors $x^{(i)} \in \mathbb{R}^m$.
- Man beachte, dass hier $y^{(i)} \in \{-1, 1\}$, wohingegen bei LDA/LR $y^{(i)} \in \{0, 1\}$.

Welche lineare Diskriminanzfunktion (Hyperebene) soll hier man wählen ?



Nach der Theorie der Maximum Margin Support Vektor Maschinen diese:



Geometrie linearer Diskriminanzfunktionen

Support Vektor Maschinen Training

SVM Training als quadratisches Optimierungsproblem

Kernelisierung der Maximum Margin SVM

Selbstkontrollfragen

Geometrie linearer Diskriminanzfunktionen

Support Vektor Maschinen Training

SVM Training als quadratisches Optimierungsproblem

Kernelisierung der Maximum Margin SVM

Selbstkontrollfragen

Definition (Lineare Diskriminanzfunktion)

Eine *Lineare Diskriminanzfunktion* ist eine multivariate reellwertige Funktion der Form

$$h : \mathbb{R}^m \to \{-1, +1\}, x \mapsto h(x) := g(f(x)), \tag{3}$$

wobei

- f eine multivariate reellwertige, parameterabhängige linear-affine Function der Form

  $$f : \mathbb{R}^m \to \mathbb{R}, x \mapsto f(x) := w^T x + w_0, \tag{4}$$

  mit *Parametervektor* $w \in \mathbb{R}^m$ und *Biasparameter* $w_0 \in \mathbb{R}$ ist und
- g eine univariate reellwertige, parameterunabhängige *Klassifikationsfunktion* der Form

  $$g : \mathbb{R} \to \{-1, 1\}, f(x) \mapsto g(f(x)) := \begin{cases} -1, & f(x) < 0 \\ +1, & f(x) \geq 0 \end{cases} \tag{5}$$

ist. Eine LDF induziert im Featurevektorenraum $\mathbb{R}^m$

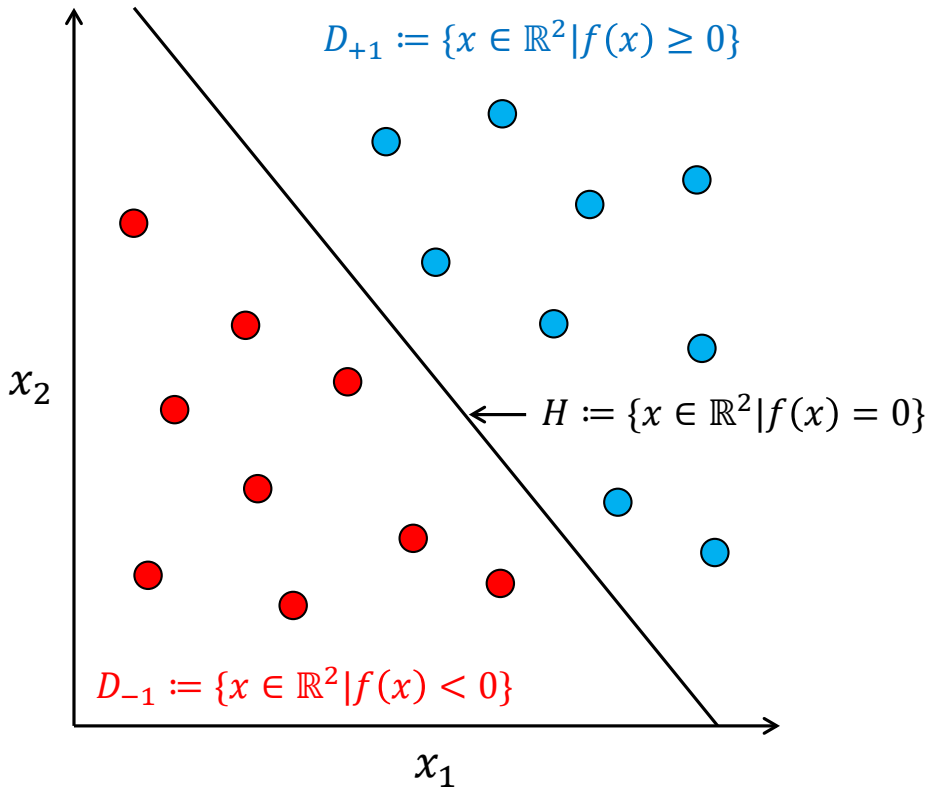
- eine *Entscheidungsgrenze* $H := \{x \in \mathbb{R}^m | f(x) = 0\}$, genannt *Hyperebene*,
- eine *Entscheidungsregion* $D_{-1} := \{x \in \mathbb{R}^m | f(x) < 0\}$, und
- eine *Entscheidungsregion* $D_{+1} := \{x \in \mathbb{R}^m | f(x) \geq 0\}$.

Geometrie linearer Diskriminanzfunktionen

$$m := 2$$

$$\Rightarrow x^{(i)} = \left(x_1^{(i)}, x_2^{(i)}\right)^T \in \mathbb{R}^2$$

- $x^{(i)}$ mit $y^{(i)} = -1$
- $x^{(i)}$ mit $y^{(i)} = +1$



Graphgleichungen für Hyperebenen in $\mathbb{R}^2$

$$f(x) = 0 \Leftrightarrow w^T x + w_0 = 0 \Leftrightarrow w_1 x_1 + w_2 x_2 + w_0 = 0 \Leftrightarrow x_2 = -\frac{w_1}{w_2} x_1 - \frac{w_0}{w_2} \qquad (6)$$

Theorem (Geometrie linearer Diskriminanzfunktionen)

$$f : \mathbb{R}^m \to \mathbb{R}, x \mapsto f(x) := w^T x + w_0 \tag{7}$$

sei eine multivariate reellwertige, parameterabhängige linear-affine Funktion und

$$H := \{x \in \mathbb{R}^m | f(x) = 0\} \subset \mathbb{R}^m \tag{8}$$

sei die zugehörige Hyperebene. Weiterhin sei

$$||v||_2 := \sqrt{v^T v} \tag{9}$$

die Euklidische Länge von $v \in \mathbb{R}^m$. Dann gelten die folgenden geometrischen Beziehungen:

(1) w ist zu jedem Vektor, der in der Richtung von H orientiert ist, orthogonal.

(2) Der minimale Euklidische Abstand d zwischen $x \in \mathbb{R}^m$ und einem Punkt auf H ist

$$d = \frac{1}{||w||_2} f(x). \tag{10}$$

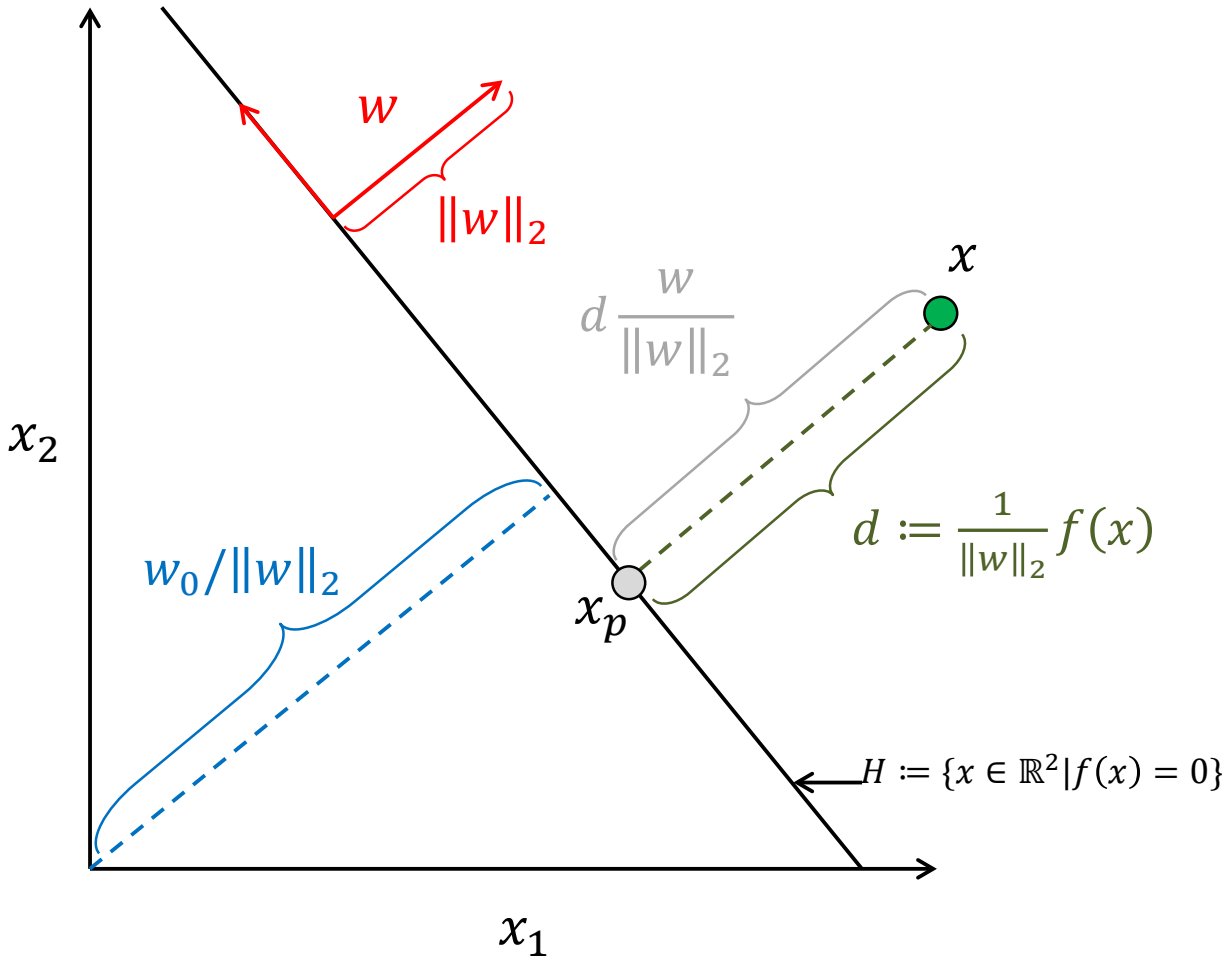
(3) Der minimale Euklidische Abstand d_0 zwischen dem Nullpunkt und einem Punkt auf H ist

$$d_0 = \frac{w_0}{||w||_2}. \tag{11}$$

Bemerkung

- w bestimmt die Orientierung der Hyperebene im Featurevektorenraum.
- w_0 bestimmt die Lage der Hyperebene im Featurevektorenraum.

Geometrie linearer Diskriminanzfunktionen



Geometrie linearer Diskriminanzfunktionen

Beweis von (1)

$x_a \in H_w$ und $x_b \in H_w$ seien zwei beliebige Punkte auf der Hyperebene. Dann gilt folgendes lineares Gleichungssystem:

$$w^T x_a + w_0 = 0 \tag{12}$$

$$w^T x_b + w_0 = 0. \tag{13}$$

Subtraktion von (13) von (12) ergibt

$$w^T x_a - w^T x_b = 0 \Leftrightarrow w^T (x_a - x_b) = 0. \tag{14}$$

Also ist der Parametervektor orthogonal zu dem Vektor $y := (x_a - x_b)$, welcher in Richtung der Hyperebene orientiert ist.

Beweis von (2)

Wir betrachten die Zerlegung eines Punktes $x \in \mathbb{R}^m$ in seine orthogonale Projektion auf eine Hyperebene $x_p \in \mathbb{R}^m$ und seinen Abstand von der Hyperebene $d \frac{w}{\|w\|_2}$

$$x = x_p + d \frac{w}{\|w\|_2}. \tag{15}$$

Diese Zerlegung ist möglich, weil w orthogonal zu jedem in Richtung der Hyperebene orientiertem Vektor ist und $\| \frac{w}{\|w\|_2} \|_2 = 1$ gilt.

Als nächstes betrachten wir die Transformation dieses so zerlegten x durch die lineare Diskriminanzfunktion:

$$f(x) = w^T x + w_0 = w^T \left(x_p + d\frac{w}{\|w\|_2} \right) + w_0 = w^T x_p + w_0 + d\frac{w^T w}{\|w\|_2}. \tag{16}$$

Dann gilt, weil $x_p \in H_w$ und somit $w^T x_p + w_0 = 0$, dass

$$f(x) = d\frac{w^T w}{\|w\|_2} = d\frac{\|w\|_2^2}{\|w\|_2} = d\|w\|_2. \tag{17}$$

Also folgt

$$d = \frac{1}{\|w\|_2} f(x). \tag{18}$$

Beweis von (3)

Für den minimalen Abstand des Nullpunktes $x_0 = (0, \ldots, 0)^T \in \mathbb{R}^m$ zu Punkten auf der Hyperebene gilt

$$d_0 = \frac{1}{\|w\|_2} f(x_0) = \frac{1}{\|w\|_2} (w^T x_0 + w_0) = \frac{1}{\|w\|_2} w^T \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix} + \frac{w_0}{\|w\|_2} = \frac{w_0}{\|w\|_2}. \tag{19}$$

□

Definition (Hyperebenenmargin und Support Vektoren)

$\mathcal{D}$ sei ein Trainingsdatensatz, f sei eine multivariate reellwertige linear-affine Funktion und H sei die durch f induzierte Hyperebene. Weiterhin sei

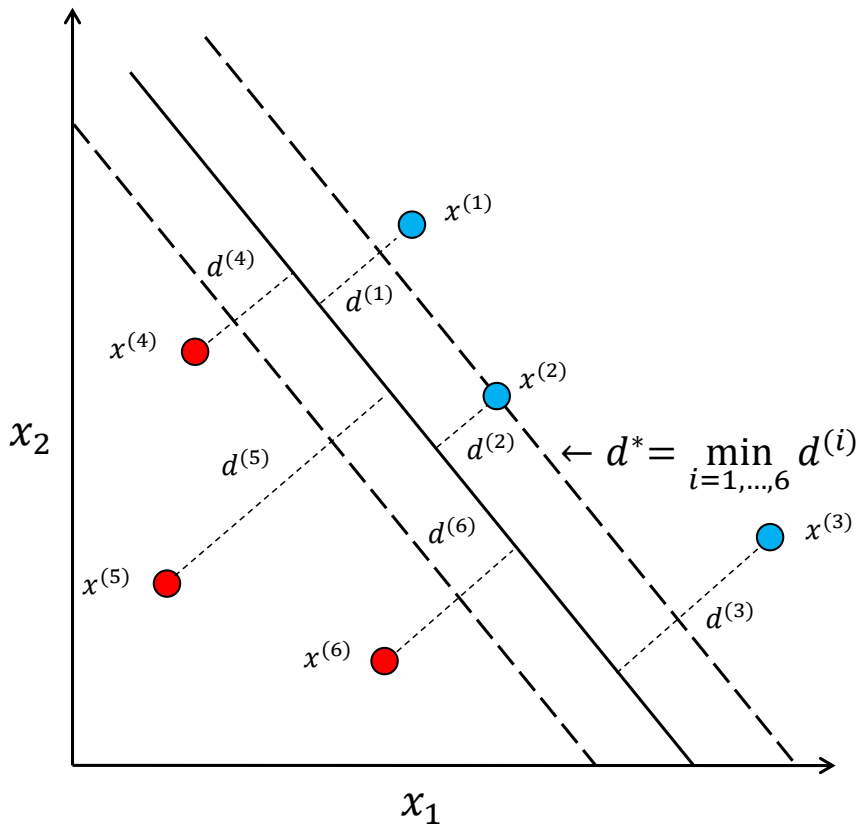
$$\left|d^{(i)}\right| := \left|\frac{1}{\|w\|_2} f\left(x^{(i)}\right)\right| = \frac{y^{(i)}}{\|w\|_2} f\left(x^{(i)}\right) = \frac{y^{(i)}(w^T x^{(i)} + w_0)}{\|w\|_2} \geq 0 \tag{20}$$

der absolute Wert des minimalen Euklidischen Abstands eines Featurevektors $x^{(i)}, i = 1, \ldots, n$ von H. Dann ist der *Margin* d^* von H hinsichtlich $\mathcal{D}$ definiert als das Minimum der absoluten minimalen Euklidischen Abstände von Featurevektoren zur Hyperebene,

$$d^* := \min_{i=1,\ldots,n} \left\{\left|d^{(i)}\right|\right\} = \min_{i=1,\ldots,n} \left\{\frac{y^{(i)}(w^T x^{(i)} + w_0)}{\|w\|_2}\right\}. \tag{21}$$

Ein Featurevektor $x^{(i)}$ wird *Support Vektor* genannt, wenn $|d^{(i)}| = d^*$, d.h., wenn $x^{(i)}$ auf dem Margin der Hyperebene liegt.

Hyperebenenmargin und Support Vektoren



Definition (Äquivalente Hyperebenen und kanonische Hyperebene)

f sei eine multivariate reellwertige linear-affine Funktion und

$$H := \{x \in \mathbb{R}^m | f(x) = 0\} \tag{22}$$

sei die durch f induzierte Hyperebene. Dann induzieren alle skalaren Vielfachen von f (und damit von w und w_0) die identische Hyperebene, denn aus $f(x) = 0$ folgt, dass $af(x) = 0$ für jedes $a \in \mathbb{R} \setminus \{0\}$. Die Hyperebenen

$$H_a := \{x \in \mathbb{R}^m | af(x) = 0, a \in \mathbb{R} \setminus \{0\}\} \tag{23}$$

heißen die zu H äquivalenten Hyperebenen. Zu einem Support Vektor x^* und einer Menge äquivalenter Hyperebenen (und somit einer Menge Parametervektoren und Biasparametern, welche die äquivalenten Hyperebenen induzieren) ist die *kanonische Hyperebene* definiert als die Hyperebene (und somit der spezifische Parametervektor w und Biasparameter w_0), für die gilt

$$|f(x^*)| = y^*(w^T x^* + w_0) = 1. \tag{24}$$

Aus der Definition der kanonischen Hyperebene folgt dann sofort, dass der Margin der kanonischen Hyperebene durch

$$d^* = \frac{1}{\|w\|_2}. \tag{25}$$

gegeben ist.

Geometrie linearer Diskriminanzfunktionen

Support Vektor Maschinen Training

SVM Training als quadratisches Optimierungsproblem

Kernelisierung der Maximum Margin SVM

Definition (Linear separierbarer Trainingsdatensatz)

Ein Trainingsdatensatz heißt *linear separierbarer Trainingsdatensatz*, wenn eine lineare Diskriminanzfunktion existiert, so dass alle Trainingsdatenpunkte korrekt klassifiziert werden können. Ein Trainingsdatensatz heißt nicht-linear separierbarer Trainingsdatensatz, wenn keine solche lineare Diskriminanzfunktion existiert.

Definition (Maximum Margin-Klassifikation)

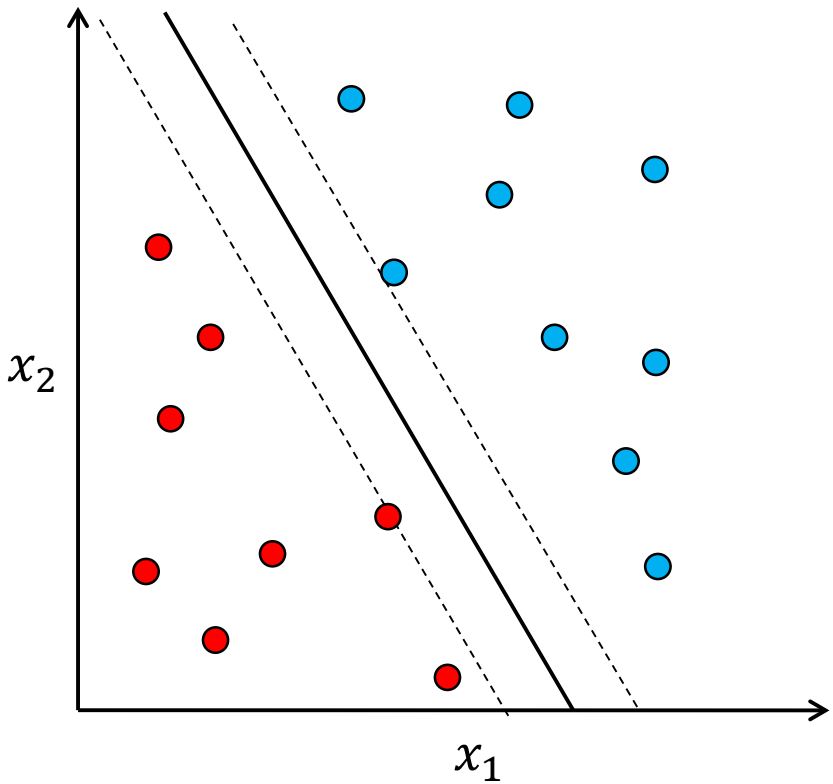
$\mathcal{D}$ sei ein linear separierbarer Trainingsdatensatz. Dann ist das Training einer Support Vektor Maschine für *Maximum Margin-Klassifikation* gegeben durch das Optimierungsproblem

$$\min_w \frac{1}{2}\|w\|_2^2 \text{ mit den Nebenbedingungen } y^{(i)}(w^T x^{(i)} + w_0) \geq 1 \text{ für } i = 1, \ldots, n. \tag{26}$$

Speziell entsprechen hierbei

- das Ziel
$$\min_w \frac{1}{2}\|w\|_2^2 \Leftrightarrow \max_w \frac{1}{\|w\|_2} \tag{27}$$
der Maximierung des Margins der von der SVM induzierten Hyperebene,
- die Nebenbedingungen
$$y^{(i)}(w^T x^{(i)} + w_0) \geq 1 \text{ für } i = 1, \ldots, n \tag{28}$$
dem Ziel, dass alle Featurevektoren auf der korrekten Seite der Hyperebene liegen oder Support Vektoren sind.

Maximum Margin-Klassifikation



$$\min \ \frac{1}{2}\|w\|_2^2 \text{ u. d. Nbg. } y^{(i)}\left(w^T x^{(i)} + w_0\right) \geq 1$$

Definition (Soft Margin-Klassifikation)

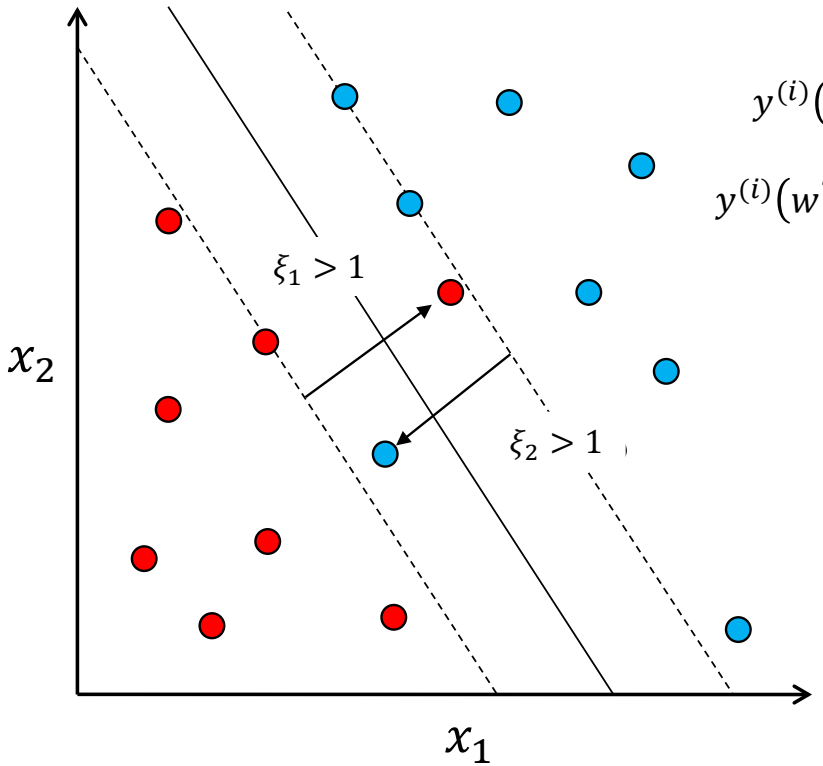
$\mathcal{D}$ sei ein nicht notwendigerweise linear separierbarer Trainingsdatensatz. Dann ist das Training einer Support Vektor Maschine für *Soft Margin-Klassifikation* gegeben durch das Optimierungsproblem

$$\min_{w,\xi} \frac{1}{2}\|w\|_2^2 + C\sum_{i=1}^{n} \xi_i^k \text{ unter den Nebenbedingungen } y^{(i)}(w^T x^{(i)} + w_0) \geq 1 - \xi_i, \xi \geq 0 \tag{29}$$

wobei $\xi := (\xi_1, \ldots, \xi_n)$ ein Vektor sogenannter *Schlupfvariablen (slack variables)* $\xi_i, i = 1, \ldots, n$ ist, der term $\sum_{i=1}^{n} \xi_i^k$ *Loss* genannt wird, $k \in \mathbb{N}$ eine Konstante ist, welche die genaue Form des Losses bestimmt (z.B. *hinge loss* für $k = 1$, *quadratic loss* für $k = 2$), und $C \in \mathbb{R}$ eine empirisch gewählte Konstante ist. Speziell entsprechen hierbei

- das Optimierungsziel dem Ziel, den Margin der durch die SVM induzierten Hyperebene zu maximieren und gleichzeitig den Loss zu minimieren, wobei die relative Gewichtung dieser beiden Ziele durch C gegeben ist,
- die Nebenbedingungen den Zielen
  (1) der korrekten Trainingsdatenpunktklassifikation und der Maximierung des Margins für $\xi_i = 0$„
  (2) der korrekten Trainingsdatenpunktklassifikation für $0 < \xi \leq 1$, und
  (3) inkorrekter Trainingsdatenpunktklassifikation für $\xi > 1$.

Soft Margin-Klassifikation



$$y^{(i)}\left(w^T x^{(i)} + w_0\right) \geq 1$$
$$\downarrow$$
$$y^{(i)}\left(w^T x^{(i)} + w_0\right) \geq 1 - \xi_i$$

$$\min_{w,w_0,\xi} \frac{1}{2}\|w\|_2^2 + C\sum_{i=1}^{n}\xi_i^k \text{ u. d. Nbg. } y^{(i)}\left(w^T x^{(i)} + w_0\right) \geq 1 - \xi_i, \xi_i \geq 0$$

Soft Margin SVM Training mit R Paket e1071

```
# Einlesen der Studienerfolgdaten und Fokus auf binäre Klassifikation
library(foreign)
D         = read.spss(file.path(getwd(), "8_Daten", "studienerfolg.sav"),
                      to.data.frame = T)
D         = D[D[,1] != "befriedigend",]

# SVM Training und Trainingsdatenprädiktion
library(e1071)                                                  # SVM Paket
acc = rep(NaN,4)                                                # Accuracyarray
for(i in 1:4){                                                  # Featuresetiteration
    x         = D[,2:(2+i-1)]                                   # Featurevektoren
    y         = D[,1]                                           # Labelvariable
    svm.train = svm(x,y, kernel = "linear")                     # SVM Training
    svm.pred  = predict(svm.train, x, kernel = "linear")        # Vorhersage der Trainingsdatenlabel
    acc[i]    = mean(svm.pred == y)                             # Trainingsdatenvorhersageaccuracy
}
print(acc)
```

```
> [1] 0.767 0.800 0.767 0.900
```

Soft Margin SVM Leave-One-Out Cross-Validation, $m = 2$

```
# Einlesen der Studienerfolgdaten und Fokus auf binäre Klassifikation
library(foreign)
D         = read.spss(file.path(getwd(), "8_Daten", "studienerfolg.sav"),
                      to.data.frame = T)
D         = D[D[,1] != "befriedigend",]
K         = nrow(D)
x         = D[,2:3]
y         = D[,1]

# K-fache Leave-One-Out Cross-Validation
library(e1071)
correct   = rep(NaN,K)
h         = rep(NaN,K)
for(k in 1:K){

    # Datensatzpartition
    x_train       = x[-k,]
    y_train       = y[-k]
    x_test        = x[k,]
    y_test        = y[k]

    # Trainingsdatensatz-basiertes Parameterlernen
    svm.train     = svm(x_train,y_train, kernel = "linear")

    # Testdatensatz-basierte Prädiktion
    svm.pred      = predict(svm.train, x_test, kernel = "linear")
    h[k]          = as.numeric(svm.pred)
    correct[k]    = svm.pred == y_test
}
cat("Accuracy: ", mean(correct))
```

Soft Margin SVM Leave-One-Out Cross-Validation, $m = 2$

	x_1	x_2	y	h(x)
1	54	44	-1	-1
2	60	20	-1	-1
3	67	36	-1	1
4	41	39	-1	-1
5	66	57	-1	1
6	51	28	-1	-1
7	51	46	-1	-1
8	37	46	-1	-1
9	57	54	-1	-1
10	47	12	-1	-1
11	50	67	-1	-1
12	42	63	-1	-1
13	60	64	-1	1
14	36	64	-1	-1
15	60	71	-1	1
31	71	41	1	1
32	65	28	1	-1
33	67	76	1	1
34	68	54	1	1
35	75	33	1	1
36	71	82	1	1
37	68	64	1	1
38	63	72	1	1
39	48	54	1	-1
40	53	86	1	-1
41	62	71	1	1
42	69	25	1	1
43	67	72	1	1
44	74	92	1	1
45	76	75	1	1

Prediction Accuracy = 0.77.

Leave-One-Out Cross-Validation, $m = 3$

	x_1	x_2	x_3	y	h(x)
1	54	44	31	-1	-1
2	60	20	33	-1	-1
3	67	36	26	-1	1
4	41	39	31	-1	-1
5	66	57	34	-1	1
6	51	28	42	-1	-1
7	51	46	34	-1	-1
8	37	46	36	-1	-1
9	57	54	28	-1	-1
10	47	12	34	-1	-1
11	50	67	27	-1	-1
12	42	63	26	-1	-1
13	60	64	29	-1	1
14	36	64	36	-1	-1
15	60	71	42	-1	1
31	71	41	37	1	1
32	65	28	33	1	-1
33	67	76	38	1	1
34	68	54	34	1	1
35	75	33	25	1	1
36	71	82	32	1	1
37	68	64	34	1	1
38	63	72	32	1	1
39	48	54	41	1	-1
40	53	86	27	1	-1
41	62	71	31	1	1
42	69	25	32	1	1
43	67	72	31	1	1
44	74	92	35	1	1
45	76	75	37	1	1

Prediction Accuracy = 0.77.

Soft Margin SVM Leave-One-Out Cross-Validation, $m = 4$

	x_1	x_2	x_3	x_4	y	h(x)
1	54	44	31	60	-1	-1
2	60	20	33	31	-1	-1
3	67	36	26	54	-1	1
4	41	39	31	26	-1	-1
5	66	57	34	56	-1	1
6	51	28	42	23	-1	-1
7	51	46	34	40	-1	-1
8	37	46	36	31	-1	-1
9	57	54	28	49	-1	-1
10	47	12	34	41	-1	-1
11	50	67	27	53	-1	-1
12	42	63	26	36	-1	-1
13	60	64	29	40	-1	1
14	36	64	36	21	-1	-1
15	60	71	42	40	-1	1
31	71	41	37	30	1	1
32	65	28	33	38	1	-1
33	67	76	38	28	1	1
34	68	54	34	53	1	-1
35	75	33	25	54	1	-1
36	71	82	32	49	1	1
37	68	64	34	33	1	1
38	63	72	32	36	1	1
39	48	54	41	34	1	-1
40	53	86	27	35	1	-1
41	62	71	31	53	1	-1
42	69	25	32	25	1	1
43	67	72	31	28	1	1
44	74	92	35	50	1	1
45	76	75	37	12	1	1

Prediction Accuracy = 0.60.

Geometrie linearer Diskriminanzfunktionen

Support Vektor Maschinen Training

SVM Training als quadratisches Optimierungsproblem

Kernelisierung der Maximum Margin SVM

Selbstkontrollfragen

Kernelisierung der Maximum Margin SVM

Kernelmethoden basieren auf dem dualen Problem des Maximum Margin SVM Trainingproblems.

Die zentrale Einsicht ist dabei, dass die Zielfunktion des dualen SVM Trainingproblems

$$q(\lambda) := \sum_{i=1}^{n} \lambda_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_i \lambda_j y^{(i)} y^{(j)} {x^{(i)}}^T x^{(j)} \tag{30}$$

lediglich von den Skalarprodukten der Featurevektoren

$${x^{(i)}}^T x^{(j)} \text{ für } i, j = 1, ..., n. \tag{31}$$

abhängt.

Die Projektion der Featurevektoren in einen "hochdimensionalen Featureraum", in welchem auf lineare Separabilität gehofft wird, benötigt also nur die Auswertung von Skalarprodukten.

Skalarprodukte in den Projektionsräumen werden *Kernel* genannt.

Geometrie linearer Diskriminanzfunktionen

Support Vektor Maschinen Training

SVM Training als quadratisches Optimierungsproblem

Kernelisierung der Maximum Margin SVM

Selbstkontrollfragen

Beginn Exkurs

Grundlagen der

Optimierung mit Nebenbedingungen

Definition (Optimierungsproblem mit Nebenbedingungen)

Ein *Optimierungsproblem mit Nebenbedingungen* hat die allgemeine Form

$$\min_{x \in \mathbb{R}^n} f(x) \text{ u.d.N. } c_i(x) = 0, i \in E, c_i(x) \geq 0, i \in I, \tag{32}$$

wobei $f : \mathbb{R}^n \to \mathbb{R}$ und $c_i : \mathbb{R}^n \to \mathbb{R}, i \in E \cup I$ glatte multivariate reellwertige Funktions und E, I endliche Indexmengen sind. f heißt *Zielfunktion*, die $c_i, i \in E$ heißen *Gleichungsnebenbedingungen* und die $c_i, i \in I$ heißen *Ungleichungsnebenbedingungen*. Die Menge

$$\mathcal{X} := \{x \in \mathbb{R}^n | c_i(x) = 0, i \in E \text{ und } c_i(x) \geq 0, i \in I\} \tag{33}$$

heißt *feasible set*.

Bemerkung

- Die notwendigen Bedingungen für Minimalstellen bei Optimierungsproblem ohne Nebenbedingungen sind für $n = 1$: $f'(x^*) = 0$ und für $n > 1$: $\nabla f(x^*) = 0_n$. Im Folgenden führen wir analoge notwendige Bedingungen erster Ordnung für Minimalstellen bei Optimierungsproblemen mit Nebenbedingungen ein.

Beispiel

Definition (Quadratisches Programm)

Ein *Quadratisches Programm* ist das konvexe Optimierungsproblem mit den Nebenbedingungen

$$\min_{x \in \mathbb{R}^n} \frac{1}{2} x^T P x + q^T x \text{ u.d.N. } Ax = b \text{ und } -Gx + h \geq 0, \tag{34}$$

wobei

- $P \in \mathbb{R}^{n \times n}$ eine positiv definite Matrix ist,
- $q \in \mathbb{R}^n$, $A \in \mathbb{R}^{p \times n}$, $b \in \mathbb{R}^p$ sind und
- $G \in \mathbb{R}^{m \times n}$, und $h \in \mathbb{R}^m$ sind.

Bemerkungen

- Quadratische Programme sind Optimierungsprobleme mit Nebenbedingungen.
- Parameterlernen bei Support Vektor Maschinen führt auf ein Quadratisches Programm.
- Optimierungstoolboxen enhalten Funktionen zur Lösung Quadratischer Programme.
- In R bietet sich das Paket `quadprog` an.

Definition (Lagrange Funktion, Lagrange Multiplikatoren)

Es sei

$$\min_{x \in \mathbb{R}^n} f(x) \text{ u.d.N. } c_i(x) = 0, i \in E, c_i(x) \geq 0, i \in I, \tag{35}$$

ein Optimierungsproblem mit Nebenbedingungen. Dann ist die *Lagrange Funktion* dieses Problems definiert als

$$L : \mathbb{R}^n \times \mathbb{R}^{|E \cup I|} \to \mathbb{R}, (x, \lambda) \mapsto L(x, \lambda) := f(x) - \sum_{i \in E \cup I} \lambda_i c_i(x). \tag{36}$$

Hierbei wird $\lambda \in \mathbb{R}^{|E \cup I|}$ *Lagrange-Multiplikatoren Vektor* genannt und die einzelnen $\lambda_i \in \mathbb{R}$ mit $i \in E \cup I$ werden *Lagrange Multiplikatoren*genannt.

Bemerkung

- Die Lagrange Funktion und die Lagrange Multiplikatoren nehmen in den notwendigen Bedingungen der Optimierung mit Nebenbedingungen eine zentrale Rolle ein.

Definition (Notwendige Bedingungen erster Ordnung)

x^* sei eine lokale Lösung des Optimierungsproblems

$$\min_{x \in \mathbb{R}^n} f(x) \text{ u.d.N. } c_i(x) = 0, i \in E, c_i(x) \geq 0, i \in I. \tag{37}$$

Dann gibt es einen Lagrange-Multiplikatoren Vektor $\lambda^* \in \mathbb{R}^{|E \cup I|}$ mit den Komponenten $\lambda_i^*, i \in E \cup I$, so dass die folgenden Bedingungen an der Stelle $(x^*, \lambda^*) \in \mathbb{R}^{n+|E \cup I|}$ gelten

$$\begin{aligned}
\nabla_x L(x^*, \lambda^*) &= 0 \\
c_i(x^*) &= 0 \text{ für alle } i \in E \\
c_i(x^*) &\geq 0 \text{ für alle } i \in I \\
\lambda_i^* &\geq 0 \text{ für alle } i \in I \\
\lambda_i^* c_i(x^*) &= 0 \text{ für alle } i \in E \cup I
\end{aligned}$$

Bemerkungen

- Die Bedingungen werden auch *Karush-Kuhn-Tucker (KKT)* Bedingungen genannt.
- Für einen Beweis und Regularitätsbedingungen, siehe Nocedal and Wright (2006) Section 12.4.
- Die letzte Bedingung impliziert $\lambda_i^* > 0 \Rightarrow c_i(x^*) = 0$.

Definition (Duales Problem)

Es sei

$$\min_{x \in \mathbb{R}^n} f(x) \text{ u.d.N. } c(x) \geq 0, \tag{38}$$

ein Optimierungsproblem ohne Gleichungsnebenbedingungen, $c(x) := (c_1(x), c_2(x), \ldots, c_m(x))^T$ sei die multivariate vektorwertige Funktion der Ungleichungsnebenbedingungen und die zugehörige Lagrange Funktion und der Lagrange Multiplikatoren Vektoren $\lambda \in \mathbb{R}^m$ seien durch

$$L : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}, (x, \lambda) \mapsto L(x, \lambda) := f(x) - \lambda^T c(x). \tag{39}$$

gegeben. Dann ist die *duale Zielfunktion* (auch *duale Lagrange Funktion genannt*) definiert als

$$q : \mathbb{R}^m \to \mathbb{R}, \lambda \mapsto q(\lambda) := \min_x L(x, \lambda), \tag{40}$$

und das *duale Problem* ist definiert als

$$\max_{\lambda \in \mathbb{R}^m} q(\lambda) \text{ u.d.N. } \lambda \geq 0. \tag{41}$$

Bemerkung

- Duale Probleme sind manchmal einfacher zu lösen als die (primären) Ausgangsprobleme.
- Duale Probleme sind für das Parameterlernen von Support Vektor Maschinen zentral.

Theorem (Schwache Dualität)

Für jede Lösung $\bar{x}$ von

$$\min_{x \in \mathbb{R}^n} f(x) \text{ u.d.N. } c(x) \geq 0, \tag{42}$$

und jedes $\bar{\lambda} \geq 0$ gilt, dass

$$q(\bar{\lambda}) \leq f(\bar{x}). \tag{43}$$

Beweis

Mit den Definitionen von q, $\bar{\lambda} \geq 0$, und $c(\bar{x}) \geq 0$, gilt, dass

$$q(\bar{\lambda}) = \min_x f(x) - \bar{\lambda}^T c(x) \leq f(\bar{x}) - \bar{\lambda}^T c(\bar{x}) \leq f(\bar{x}). \tag{44}$$

□

Bemerkung

- Das Theorem besagt, dass der optimierte Wert des dualen Problems eine untere Grenze für den optimalen Wert der Zielfunktion des Ausgangsproblems ist.

Theorem (Starke Dualität)

Gegeben seien das Optimierungsproblem

$$\min_{x \in \mathbb{R}^n} f(x) \text{ u.d.N. } c(x) \geq 0 \tag{45}$$

und seine zugehörigen notwendigen Bedingungen erster Ordnung

$$\begin{aligned} \nabla f(\bar{x}) - \nabla c(\bar{x})\bar{\lambda} &= 0, \\ c(\bar{x}) &\geq 0, \\ \bar{\lambda} &\geq 0, \\ \bar{\lambda}_i c_i(\bar{x}) &= 0, i = 1, 2, \ldots, m, \end{aligned} \tag{46}$$

mit $\nabla c(x) = (\nabla c_1(x), \nabla c_2(x), \ldots, \nabla c_m(x)) \in \mathbb{R}^{n \times m}$. $\bar{x}$ sei eine Lösung des Ausgangsproblems und f sowie $-c_i, i = 1, 2, \ldots, m$ konvexe Funktionen auf $\mathbb{R}^n$, die in $\bar{x}$ differenzierbar sind. Dann ist jedes $\bar{\lambda}$, für das $(\bar{x}, \bar{\lambda})$ die notwendigen Bedingungen des Ausgangsproblem erfüllt, eine Lösung des dualen Problems

Bemerkungen

- Die optimalen Lagrange Multiplikatoren des Ausgangsproblems sind Lösungen des dualen Problems.
- SVM Training als Quadratisches Programm benötigt das Konzept der starken Dualität.

Beweis

Wir nehmen an, dass $(\bar{x}, \bar{\lambda})$ die notwendigen Bedingungen erster Ordnung für ein Minimum des Ausgangsproblem erfüllen und dass $L(\cdot, \bar{\lambda})$ konvex und differenzierbar ist. Dann gilt für jedes $x \in \mathbb{R}^n$, dass

$$L(x, \bar{\lambda}) \geq L(\bar{x}, \bar{\lambda}) + \nabla_x L(\bar{x}, \bar{\lambda})(x - \bar{x}) = L(\bar{x}, \bar{\lambda}), \tag{47}$$

weil $\nabla_x L(\bar{x}, \bar{\lambda}) = 0$. Also gilt für die duale Zielfunktion

$$q(\bar{\lambda}) = \inf_x L(x, \bar{\lambda}) = L(\bar{x}, \bar{\lambda}). \tag{48}$$

Mit der letzten der notwendigen Bedingungen erster Ordnung folgt weiterhin

$$q(\bar{\lambda}) = L(\bar{x}, \bar{\lambda}) = f(\bar{x}) - \bar{\lambda}^T c(\bar{x}) = f(\bar{x}) \tag{49}$$

Schließlich gilt mit dem Theorem zur Schwachen Dualität, dass $q(\lambda) \leq f(\bar{x})$ für alle $\lambda \geq 0$. Also folgt mit $q(\bar{\lambda}) = f(\bar{x})$, dass $\bar{\lambda}$ eine Lösung des dualen Problems ist. □

Ende Exkurs

Grundlagen der

Optimierung mit Nebenbedingungen

Theorem (SVM Training als quadratisches Programmierungsproblem I)

Das duale Problem des Maximum Margin SVM Trainingproblems

$$\min_{w} \frac{1}{2}\|w\|_2^2 \text{ mit den Nebenbedingungen } y^{(i)}(w^T x^{(i)} + w_0) \geq 1 \text{ für } i = 1, \ldots, n \tag{50}$$

ist gegeben als

$$\max_{\lambda \in \mathbb{R}^n} q(\lambda) := \sum_{i=1}^{n} \lambda_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_i \lambda_j y^{(i)} y^{(j)} {x^{(i)}}^T x^{(j)} \tag{51}$$

unter den Nebenbedingungen

$$\lambda \geq 0 \text{ and } \sum_{i=1}^{n} \lambda_i y^{(i)} = 0. \tag{52}$$

Basierend auf einer Lösung $\bar{\lambda}$ des dualen Problems sind alle $x^{(i)}$ mit $\bar{\lambda}_i > 0, i = 1, \ldots, n$ Support Vektoren und die Lösungen für den Parametervektor und den Biasparameter des primären Problems sind

$$\bar{w} = \sum_{i=1}^{n} \bar{\lambda_i} y^{(i)} x^{(i)} \text{ and } \bar{w}_0 = \frac{1}{n} \sum_{i=1}^{n} \left(y^{(i)} - \bar{w}^T x^{(i)} \right), \tag{53}$$

respective.

Theorem (SVM Training als quadratisches Programmierungsproblem II)

Weiterhin gilt, dass bei Definition von

$$y := \left(y^{(i)}\right)_{i=1,\ldots,n} \in \mathbb{R}^n \text{ and } K := \left(x^{(i)^T} x^{(j)}\right)_{i,j=1,\ldots,n} \in \mathbb{R}^{n\times n}, \tag{54}$$

as well as

$$P := yy^T K \in \mathbb{R}^{n\times n}, q := -1_n, G := -I_n, h := 0_n, A := y^T, \text{ and } b := 0 \tag{55}$$

das duale Problem des Maximum Marging SVM Trainingproblems als quadratisches Programmierproblem der Form

$$\min_{\lambda\in\mathbb{R}^n} \frac{1}{2}\lambda^T P\lambda + q^T\lambda \text{ mit den Nebenbedingungen } -G\lambda + h \geq 0_n \text{ and } A\lambda = b \tag{56}$$

geschrieben werden kann, und somit mit allen Standardalgorithmen der quadratischen Programmierung gelöst werden kann.

Bemerkungen

- Einerseits führt die QP Formulierung von Maximum Margin SVM Problem auf ein Standardproblem.
- Andererseits motiviert die QP Formulierung des Maximum Margin SVM Problems auch Kernelmethoden

Beweis

(1) Lagrangefunktion des primären Problems

Per definition ist die Lagrangefunktion des primären Problems

$$\min_{w,w_0} \frac{1}{2}\|w\|_2^2 \text{ mit den Nebenbedingungen } y^{(i)}(w^T x^{(i)} + w_0) \geq 1 \text{ für } i = 1, \ldots, n \tag{57}$$

gegeben durch

$$L(w, w_0, \lambda) := \frac{1}{2} w^T w - \sum_{i=1}^{n} \lambda_i (y^{(i)}(w^T x^{(i)} + w_0) - 1). \tag{58}$$

(2) Bestimmung der Zielfunktion des dualen Problems

Per definition ist die Zielfunktion des dualen Problems gegeben durch

$$q : \mathbb{R}^n \to \mathbb{R}, \lambda \mapsto q(\lambda) := \min_{w,w_0} L(w, w_0, \lambda). \tag{59}$$

Die analytische Bestimmung des Minimums der Lagrangefunktion L hinsichtlich w und w_0 entspricht der Bestimmung der partiellen Ableitungen von L hinsichtlich w und w_0, Nullsetzen, und lösen. Wir definieren

$$\bar{w} := \arg\min_{w \in \mathbb{R}^m} L(w, w_0, \lambda) \text{ and } \bar{w_0} := \arg\min_{w_0 \in \mathbb{R}} L(w, w_0, \lambda). \tag{60}$$

Beweis (fortgeführt)

Für die Minimierung von L hinsichtlich w ergibt sich

$$\begin{aligned}\nabla_w L(w, w_0, \lambda) &= \nabla_w \left(\frac{1}{2} w^T w - \sum_{i=1}^{n} \lambda_i (y^{(i)} (w^T x^{(i)} + w_0) - 1) \right) \\ &= w - \nabla_w \left(\sum_{i=1}^{n} \lambda_i y^{(i)} w^T x^{(i)} + \lambda_i y^{(i)} w_0 - \lambda_i \right) \qquad (61) \\ &= w - \sum_{i=1}^{n} \lambda_i y^{(i)} x^{(i)}\end{aligned}$$

und somit

$$\bar{w} = \sum_{i=1}^{n} \lambda_i y^{(i)} x^{(i)}. \qquad (62)$$

Beweis (fortgeführt)

In ähnlicher Weise ergibt sich für die Minimierung von L bezüglich w_0

$$\begin{aligned}\nabla_{w_0} L(w, w_0, \lambda) &= \nabla_{w_0} \left(\frac{1}{2} w^T w - \sum_{i=1}^{n} \lambda_i (y^{(i)} (w^T x^{(i)} + w_0) - 1) \right) \\ &= \nabla_{w_0} \left(\sum_{i=1}^{n} \lambda_i y^{(i)} w^T x^{(i)} + \lambda_i y^{(i)} w_0 - \lambda_i \right) \\ &= -\sum_{i=1}^{n} \lambda_i y^{(i)}.\end{aligned} \tag{63}$$

An der Minimalstelle von L hinsichtlich w_0 ergibt sich also

$$-\sum_{i=1}^{n} \lambda_i y^{(i)} = 0. \tag{64}$$

Man beachte, dass wir hier lediglich die Bedingung $-\sum_{i=1}^{n} \lambda_i y^{(i)} = 0$ an der Minimalstelle von L hinsichtlich w_0 erhalten, nicht aber die Minimalstelle $\bar{w}_0$ selbst.

SVM Training als quadratisches Programmierungsproblem

Beweis (fortgeführt)

Für die Zielfunktion des dualen Problems ergibt sich also

$$
\begin{aligned}
& q(\lambda) \\
& = \min_{w, w_0} L(w, w_0, \lambda) \\
& = L(\bar{w}, \bar{w}_0, \lambda) \\
& = \frac{1}{2} \bar{w}^T \bar{w} - \sum_{i=1}^{n} \lambda_i \left(y^{(i)} (\bar{w}^T x^{(i)} + \bar{w}_0) - 1 \right) \\
& = \frac{1}{2} \left(\sum_{i=1}^{n} \lambda_i y^{(i)} x^{(i)} \right)^T \left(\sum_{j=1}^{n} \lambda_i y^{(j)} x^{(j)} \right) - \sum_{i=1}^{n} \lambda_i \left(y^{(i)} \left(\left(\sum_{j=1}^{n} \lambda_j y^{(j)} x^{(j)} \right)^T x^{(i)} + \bar{w}_0 \right) - 1 \right. \\
& = \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_i \lambda_j y^{(i)} y^{(j)} x^{(i)^T} x^{(j)} - \sum_{i=1}^{n} \lambda_i y^{(i)} \left(\left(\sum_{j=1}^{n} \lambda_j y^{(j)} x^{(j)} \right)^T x^{(i)} + \bar{w}_0 \right) + \sum_{i=1}^{n} \lambda_i
\end{aligned}
$$

Beweis (fortgeführt)

und weiterhin

$$
\begin{aligned}
q(\lambda) &= \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\lambda_i\lambda_j y^{(i)}y^{(j)}x^{(i)^T}x^{(j)} - \sum_{i=1}^{n}\lambda_i\lambda_j y^{(i)}y^{(j)}x^{(i)^T}x^{(j)} - \bar{w}_0\sum_{i=1}^{n}\lambda_i y^{(i)} + \sum_{i=1}^{n}\lambda_i \\
&= \sum_{i=1}^{n}\lambda_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\lambda_i\lambda_j y^{(i)}y^{(j)}x^{(i)^T}x^{(j)} - \bar{w}_0\sum_{i=1}^{n}\lambda_i y^{(i)} \\
&= \sum_{i=1}^{n}\lambda_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\lambda_i\lambda_j y^{(i)}y^{(j)}x^{(i)^T}x^{(j)}.
\end{aligned}
$$

Hierbei folgt die letzte Gleichung mit der Tatsache, dass an der Stelle $\bar{w}_0$ gilt, dass $\sum_{i=1}^{n}\lambda_i y^{(i)} = 0$.

Beweis (fortgeführt)

Wir haben also gezeigt, dass die Zielfunktion des dualen Problems des Maximum Margin SVM Trainingsproblems von der Form

$$q : \mathbb{R}^n \to \mathbb{R}, \lambda \to q(\lambda) = \sum_{i=1}^{n} \lambda_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_i \lambda_j y^{(i)} y^{(j)} {x^{(i)}}^T x^{(j)}. \tag{65}$$

ist.

(3) Formulierung des dualen Problems

Das duale Problem zum Maximum Margin SVM Trainingsproblem ergibt sich also zu

$$\max_{\lambda \in \mathbb{R}} q(\lambda) = \sum_{i=1}^{n} \lambda_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_i \lambda_j y^{(i)} y^{(j)} {x^{(i)}}^T x^{(j)} \tag{66}$$

unter den Nebenbedingunge

$$\lambda_i \geq 0, i = 1, \ldots, n \text{ and } \sum_{i=1}^{n} \lambda_i y^{(i)} = 0, \tag{67}$$

wobei die letzte Nebenbedingung das Minimum der Lagrangefunktion hinsichtlich w_0 sicherstellt.

Beweis (fortgeführt)

Lösen des dualen Problems mithilfe eines Standardalgorithmus ergibt einen Vektor optimaler Lagrangemultiplikatoren

$$\bar{\lambda} = \underset{\lambda \in \mathbb{R}^n}{\arg\max}\, q(\lambda) = \underset{\lambda \in \mathbb{R}^n}{\arg\max}\, L(\bar{w}, \bar{w}_0, \lambda). \tag{68}$$

Basierend auf der Minimierung von L hinsichtlich von w ergibt sich also

$$\bar{w} = \sum_{i=1}^{n} \bar{\lambda}_i y^{(i)} x^{(i)}. \tag{69}$$

Schließlich ergibt sich für den optimalen Biasparameter $\bar{w}_0$ zunächst mit den KKT Bedingungen, dass für alle $\hat{\lambda}_i > 0, i = 1, \ldots, n$ gilt, dass

$$\begin{aligned} y^{(i)}(\bar{w}^T x^{(i)} + w_0) - 1 &= 0 \\ \Leftrightarrow y^{(i)}(\bar{w}^T x^{(i)} + w_0) &= 1 \\ \Leftrightarrow y^{(i)} y^{(i)}(\bar{w}^T x^{(i)} + w_0) &= y^{(i)} \\ \Leftrightarrow \bar{w}^T x^{(i)} + w_0 &= y^{(i)}. \end{aligned} \tag{70}$$

SVM Training als quadratisches Programmierungsproblem

Beweis (fortgeführt)

Dies impliziert, erstens, dass alle $x^{(i)}$ mit $\bar{\lambda}_i > 0$ Support Vektoren sind, weil ihre Distanz zur optimalen Hyperebene gleich 1 ist, und zweitens, dass

$$\sum_{i=1}^{n} \bar{w}^T x^{(i)} + n w_0 = \sum_{i=1}^{n} y^{(i)} \Leftrightarrow w_0 = \frac{1}{n} \sum_{i=1}^{n} \left(y^{(i)} - \bar{w}^T x^{(i)} \right). \tag{71}$$

(4) Standardform eines QP Problems

Die Äquivalenzen

$$\sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_i \lambda_j y^{(i)} y^{(j)} {x^{(i)}}^T x^{(j)} \Leftrightarrow \lambda^T y y^T K \lambda \Leftrightarrow \lambda^T P \lambda$$

$$\sum_{i=1}^{n} \lambda_i \Leftrightarrow 1_n^T \lambda \Leftrightarrow q^T \lambda \tag{72}$$

$$\lambda \geq 0 \Leftrightarrow I_n \lambda + 0_n \geq 0_n \Leftrightarrow -G\lambda + h \leq 0_n$$

$$\sum_{i=1}^{n} \lambda_i y^{(i)} = 0 \Leftrightarrow y^T \lambda = 0 \Leftrightarrow A\lambda = b$$

ergeben sich direkt mit den Regeln der Matrixmultiplikation.

Geometrie linearer Diskriminanzfunktionen

Support Vektor Maschinen Training

SVM Training als quadratisches Optimierungsproblem

Kernelisierung der Maximum Margin SVM

Selbstkontrollfragen

Kernelisierung der Maximum Margin SVM

Kernelmethoden basieren auf dem dualen Problem des Maximum Margin SVM Trainingproblems.

Die zentrale Einsicht ist dabei, dass die Zielfunktion des dualen SVM Trainingproblems

$$q(\lambda) := \sum_{i=1}^{n} \lambda_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_i \lambda_j y^{(i)} y^{(j)} {x^{(i)}}^T x^{(j)} \tag{73}$$

lediglich von den Skalarprodukten der Featurevektoren

$${x^{(i)}}^T x^{(j)} \text{ für } i, j = 1, ..., n. \tag{74}$$

abhängt.

Die Projektion der Featurevektoren in einen "hochdimensionalen Featureraum", in welchem auf lineare Separabilität gehofft wird, benötigt also nur die Auswertung von Skalarprodukten.

Skalarprodukte in den Projektionsräumen werden *Kernel* genannt.

Geometrie linearer Diskriminanzfunktionen

Support Vektor Maschinen Training

SVM Training als quadratisches Optimierungsproblem

Kernelisierung der Maximum Margin SVM

Selbstkontrollfragen

Selbstkontrollfragen

1. Wie unterscheiden sich binäre Klassifikationsdatensätze für Support Vektor Maschinen von binären Klassifikationsdatensätzen für Lineare Diskriminanzanalyse und Logistische Regression?
2. Definieren Sie den Begriff der linearen Diskriminanzfunktion.
3. Definieren Sie in Bezug zum Begriff der linearen Diskriminanzfunktion die Begriffe der Entscheidungsgrenze, der Hyperebene und der Entscheidungsregion.
4. Geben Sie das Theorem zur Geometrie linearer Diskriminanzfunktionen wieder.
5. Erläutern Sie die Bedeutung des Theorems zur Geometrie linearer Diskriminanzfunktionen bei der Bestimmung von Hyperebenen.
6. Definieren Sie den Begriff des Hyperebenenmargins.
7. Definieren Sie den Begriff des Support Vektors.
8. Definieren Sie den Begriff der kanonischen Hyperebene.
9. Definieren Sie den Begriff des linear separierbaren Trainingsdatensatzes.
10. Definieren und erläutern Sie das Optimierungsproblem zur Maximum Margin Klassifikation bei SVMs.
11. Definieren und erläutern Sie das Optimierungsproblem zur Soft Margin Klassifikation bei SVMs.
12. Lesen Sie den Datensatz studienerfolg.sav mit R ein und bestimmen Sie den Trainingsdatenprädiktionsfehler nach Trainieren einer Support Vektor Maschine mithilfe des R Pakets e1071 zur prädiktiven Modellierung des Studienerfolgs (gut, ungenügend) basierend auf (1) den Intelligenztestdaten, (2) den Intelligenz- und Mathematiktestdaten und (3) den Intelligenztest-, Mathematiktest-, und Gewissenhaftigkeitsdaten.

References

Dwyer, Dominic B., Peter Falkai, and Nikolaos Koutsouleris. 2018. "Machine Learning Approaches for Clinical Psychology and Psychiatry." *Annual Review of Clinical Psychology* 14 (1): 91–118. https://doi.org/10.1146/annurev-clinpsy-032816-045037.

Nocedal, Jorge, and Stephen J. Wright. 2006. *Numerical Optimization*. 2nd ed. Springer Series in Operations Research. New York: Springer.