

# Zusammenfassung - Mustererkennung und Kontextanalyse

Andreas Ruschinski, Marc Meier

14. Januar 2016

Korrektheit und Vollständigkeit der Informationen sind nicht gewährleistet. Macht euch eigene Notizen oder ergänzt/korrigiert meine Ausführungen!

## Inhaltsverzeichnis

<b>1</b>	<b>Überblick Klassifikation</b>	<b>2</b>
1.1	Einführendes Beispiel . . . . .	2
1.2	Entscheidungsgrenzen . . . . .	2
1.3	Mustererkennungssysteme . . . . .	2
1.4	Entwurf von Mustererkennungssystemem . . . . .	3
<b>2</b>	<b>Grundlagen Signalverarbeitung</b>	<b>3</b>
2.1	Digitalisierung . . . . .	3
2.1.1	Dithering . . . . .	3
2.1.2	Abtastung . . . . .	3
2.2	Lineare Systeme . . . . .	4
2.2.1	Überlagerung . . . . .	4
2.2.2	Faltung . . . . .	4
2.2.3	Korrelation . . . . .	4
2.3	Fourier-Transformation . . . . .	4
2.3.1	Allgemein . . . . .	4
2.3.2	Fourier-Transformation Grundideen . . . . .	5
2.3.3	Diskrete Fourier-Transformation . . . . .	5
2.4	TODO . . . . .	6
<b>3</b>	<b>Merkmale</b>	<b>6</b>
3.1	Statistische Merkmale . . . . .	6
3.1.1	Erwartungswert . . . . .	6
3.1.2	Momente . . . . .	6
3.1.3	Emprische Werte . . . . .	6
3.1.4	Lageparameter . . . . .	6
3.1.5	Streuungsmaße . . . . .	6
3.1.6	Standardisierung . . . . .	7
3.1.7	Korrelation . . . . .	7
3.1.8	Einsatz statistischer Merkmale . . . . .	7
3.2	Merkmalstypen . . . . .	7
3.3	Merkmale für Zeitreihen . . . . .	7
3.3.1	Summarische Merkmale . . . . .	7
3.3.2	Autokorrelation, Grundfrequenz . . . . .	8
3.3.3	Phasendifferenz . . . . .	8
3.4	Merkmale für kinetische Systeme ??? . . . . .	8
<b>4</b>	<b>Bayessche Entscheidungstheorie</b>	<b>8</b>
4.1	Einführung . . . . .	8
4.2	Bayes'sche Entscheidungstheorie . . . . .	9
4.2.1	Definitionen . . . . .	9
4.2.2	Bayes'sche Entscheidungstheorie - formal . . . . .	9
4.2.3	Anwendung . . . . .	9
4.2.4	Klassifikation und Diskriminanten . . . . .	9

4.2.5	Normalverteilung . . . . .	10
4.3	Diskriminanten für die Normalverteilung . . . . .	10
4.4	Empirischer Fall . . . . .	11
4.5	Diskrete Merkmale . . . . .	11
<b>5</b>	<b>Parameterschätzung</b>	<b>11</b>
5.1	Einführung . . . . .	11
5.2	Maximum Likelihood-Schätzer . . . . .	12
5.3	Bayessche Parameterschätzung . . . . .	13
<b>6</b>	<b>Dimensionsreduktion</b>	<b>13</b>
6.1	Einführung . . . . .	13
6.2	Hauptkomponentenanalyse . . . . .	14
6.3	Diskriminantenanalyse . . . . .	14
6.4	Nichtlineare Diskriminanten . . . . .	14
<b>7</b>	<b>Nicht-parametrische Methoden</b>	<b>15</b>
7.1	Einführung . . . . .	15
7.2	Parzen Windows . . . . .	16
7.3	k Nearest Neighbours . . . . .	17
<b>8</b>	<b>Support Vector Machines</b>	<b>17</b>
8.1	Einführung . . . . .	17
8.2	Entscheidungsgrenzen . . . . .	17
8.3	Lagrange-Multiplikatoren . . . . .	18
8.4	Bestimmung der Entscheidungsgrenze . . . . .	18
8.5	Kerne und Dimensionen . . . . .	19

# 1 Überblick Klassifikation

## 1.1 Einführendes Beispiel

- Ziel: Bestimmung von Fischen auf der Basis von Kamerainformationen
- Verwendung der Kamera zum Merkmale (Features) bestimmen des aktuellen Fisches: Länge, Helligkeit und Breite
- Annahme: Die Modelle (Beschreibung) der Fische unterscheiden sich
- Klassifikation: Finde zu gegebenen Merkmalen das am besten passende Modell (Welche Beschreibung der Fische passt am besten zu dem aktuellen Fisch)

## 1.2 Entscheidungsgrenzen

Komplexe Entscheidungsgrenze  $\Rightarrow$  Mehr Parameter werden für die Bestimmung benötigt  $\Rightarrow$  Weniger Trainingsdaten stehen zur Bestimmung des einzelnen Parameters zur Verfügung  $\Rightarrow$  Parameter wird ungenauer bestimmt und ist anfälliger gegen Schwankungen der Trainingsdaten  $\Rightarrow$  Mehr Features  $\Rightarrow$  höhere Dimensionalität des Merkmalsraums  $\Rightarrow$  größere inhärente Komplexität der gegebenen Form von Entscheidungsgrenzen  $\Rightarrow$  schlechtere Bestimmung der Parameter

## 1.3 Mustererkennungssysteme

- 1) **Sensing:** Erfassung der Umwelt mittels Sensoren, z.B: Kamera, Bewegungssensoren, Mikrofon, RFID-Lesegerät, Problem: Eigenschaften und Begrenzungen (Bandbreite, Auflösung, Empfindlichkeit, Verzerrung, Rauschen, Latenz, ...) des Sensors beeinflussen Problemschwierigkeit
- 2) **Segmentation:** Identifikation der einzelnen Musterinstanzen (Identifikation der für unser Problem relevanten Daten), z.B: Fisch auf dem Fließband
- 3) **Merkmalsberechnung:** Bestimmung von Features, gesucht sind dabei Features, welche eine Diskriminierungsfähigkeit haben (Zwischen zwei gleichen Klassen ähnlicher Wert, zwischen zwei verschiedenen Klassen großer Werteunterschied) und Invariant gegenüber Signaltransformationen (Rotation, Translation, Skalierung, perspektivische Verzerrung) sind; Problem: Wie kann ich aus einer großen Auswahl an Merkmalen die am besten geeigneten finden?

- 4) **Klassifikation:** Annahme: Modelle (Grundlegende Eigenschaften) der Klassen (verschiedene Fische) unterscheiden sich; Ziel: Zuweisung von Problem instanzen aufgrund deren Merkmale zu der am besten entsprechenden Klasse
- 5) **Nachbereitung:** Entscheidung auf Basis Klassifikation, Problem: Wie können wir Kontextinformationen nutzen? Können wir verschiedene Klassifikatoren zusammen nutzen?

## 1.4 Entwurf von Mustererkennungssystemem

- 1) **Daten sammeln:** Wie kann man wissen, wann eine Menge von Daten ausreichend groß und repräsentativ ist, um das Klassifikationssystem zu trainieren und zu testen?
- 2) **Merkmale bestimmen:** Gesucht: Einfach zu extrahierende Merkmale mit hoher Diskriminativität, Invariant gegenüber irrelevanten Transformationen, unempfindlich gegenüber Rauschen, Problem: Wie kann ich A-priori-Wissen nutzen?
- 3) **Modell auswählen:** Wie erkennt man, wann ein Modell sich in der Klassifikation signifikant von einem anderen Modell – oder vom wahren Modell – unterscheidet? Wie erkennt man, dass man eine Klasse von Modellen zugunsten eines anderen Ansatzes ablehnen sollte? Versuch-und-Irrtum oder gibt es systematische Methoden?
- 4) **Klassifikator trainieren:** Verwende gesammelte Daten, um Parameter des Klassifikators zu bestimmen
- 5) **Klassifikator evaluieren:** Wie bewertet man die Leistung? Wie verhindert man Overfitting/Underfitting?

Für weitere Informationen sind folgende Referenzen zu konsultieren: [1, S. 3-16]

## 2 Grundlagen Signalverarbeitung

In diesem Abschnitt werden die Grundlagen der digitalen Signalverarbeitung beschrieben. In der digitalen Signalverarbeitung werden Methoden und Techniken behandelt welche aus analogen Sensorwerten eine digitale Information erstellen.

### 2.1 Digitalisierung

Die gemessenen Werte der Sensoren werden durch unterschiedliche Ausgangsspannungen realisiert d.h. in Abhängigkeit von der gemessenen Größe ändert sich die gemessene Spannung am Ausgang des Sensors.

Diese analogen Signale werden im ersten Schritt zeitlich diskretisiert d.h. die kontinuierlichen Signale werden durch Abtastung angenähert. Unter Abtastung versteht man die Erhebung eines Wertes zu einem Zeitpunkt. Die Häufigkeit der Abtastung wird in Herz (Hz) angegeben d.h. Abtastung mit 10 Hz entspricht 10 maliges abtasten des analogen Signales innerhalb von einer Sekunde. Durch diesen Schritt erhalten wir eine Folge von gemessenen Spannungen.

Im nächsten Schritt werden die abgetasteten Werte diskretisiert (Diskretisierung der Amplituden) d.h. jeder Spannung wird ein digitaler Wert zugewiesen. Dies geschieht mittels einem A/D-Wandler, welcher entsprechend von Grenzwerten (1/2 Spannung, 1/4 Spannung, 1/8 Spannung) entsprechende Bits setzt und diese Information ausgibt.

#### 2.1.1 Dithering

Ein Problem bei der Diskretisierung der Amplitude ergibt sich dadurch, dass bei einem sehr geringen Sensorwert das LSB nicht gesetzt wird. Dies hat zur Folge das Informationen verloren gehen. Um dies zu verhindern wird Zufallsrauschen auf den aktuellen Sensorwert addiert. Dadurch wird der Grenzwert manchmal überschritten. So nähert sich der Erwartungswert den Realwert an.

#### 2.1.2 Abtastung

Ein Signal nur dann kann korrekt abgetastet werden, wenn es keine Frequenzanteile enthält, die oberhalb der halben Abtastrate liegen. (Abtasttheorem)

$$f_{max} \leq \frac{1}{2} f_{sample}$$

Aus dem Abtasttheorem folgt: Wenn wir ein Signal mit  $f_{max}$  korrekt abzutasten wollen, müssen wir dieses Signal mit einer Frequenz von  $2 * f_{max}$  abtasten.

## 2.2 Lineare Systeme

Ein lineares System erfüllt folgende Eigenschaften:

**Homogenität**  $f(c * a) = c * f(a)$  d.h. eine Veränderung des Input-Signales hat eine identische Änderung des Output-Signals zu Folge

**Additivität**  $f(a + b) = f(a) + f(b)$  d.h. wenn das Input-Signal aus zwei überlagerten Signalen besteht können wir diese getrennt auswerten und anschließend die Ergebnisse addieren

**Translationsinvarianz**  $f(n) = y(n) \rightarrow f(n + s) = y(n + s)$  d.h. ein zeitlicher Versatz des Input-Signals hat den selben zeitlichen Versatz im Output-Signal zur Folge

**Kommutativität**  $f(a) = b, g(b) = c \rightarrow g(a) = b, f(b) = c$  d.h. wenn mehrere lineare Systeme in einer Reihe verknüpft sind, können diese vertauscht werden ohne das Ergebnis zu beeinflussen

Aus diesen Eigenschaften folgt: Ein lineares System ist vollständig durch seine Impulsantwort charakterisiert. Eine Impulsantwort erhalten wir durch Eingabe eines Signals, welches genau an einer Stelle einen Wert größer als 0 hat (Deltafunktion). Die daraus resultierende Antwort beinhaltet alle Eigenschaften des linearen Systemes d.h. unter Verwendung der o.g. Eigenschaften können wir nachfolgend auf Basis der Impulsantwort ermitteln, welches Ergebnis aus anderen Input-Signalen resultiert.

### 2.2.1 Überlagerung

Aus den Eigenschaften des linearen Systems folgt:  $f(x) = f(x_1 + x_2 + x_3) = f(x_1) + f(x_2) + f(x_3)$  d.h. wir können das Eingangssignal zerlegen (Decomposition) und die zerlegten Signale wieder zusammenführen (Synthese), ohne dass das Ergebnis der Analyse beeinflusst wird.

Diese Überlegung können wir nutzen um das Eingangssignal in mehrere Deltafunktionen zu zerlegen. Anschließend werden diese analysiert und die Teilergebnisse zusammengefasst. Auf diese Weise wird das Ergebnis aus dem Eingangssignal zu ermitteln.

Des Weiteren ist auch eine Zerlegung des Signal in mehrere Cosinus- und Sinus-Signale interessant (siehe ).

### 2.2.2 Faltung

Um die Faltung zu berechnen benötigen wir ein Eingangssignal und die Impulsantwort des Systemes.

Die Grundidee besteht darin, dass wir das Eingangssignal in einzelne Delta-Impulse zerlegen. Für jeden dieser Delta-Impulse wird die entsprechende verschobene und skalierte Kopie der Impulsantwort berechnet. Anschließend werden alle Impulsantworten addiert.

Hierfür ergibt sich somit folgende Formel:  $y[i] = \sum_{j=1}^M h[j] * x[i - j]$  mit  $h$  ist Impulsantwort und  $x$  das Eingangssignal.

Durch eine geeignete Wahl der Impulsantwort können Filter, Ableitungen und Integrale realisiert werden. Im nächsten Abschnitt wird ein Verfahren beschrieben, welches die Faltung nutzt um eine Korrelation zu berechnen.

### 2.2.3 Korrelation

Das Ziel in der Korrelation ist die Erkennung eines bekannten Signales  $t$  innerhalb eines verrauhten Signales  $x$ .

Für die Berechnung der Korrelation nutzen wir die Faltung mit der gespiegelten Impulsantwort  $y[n] = x[n] * t[-n]$ . Als Ergebnis dieser Faltung erhalten wir ein Ausgangssignal  $y$ , welches signifikante Ausschläge im übereinstimmenden Bereich hat.

## 2.3 Fourier-Transformation

### 2.3.1 Allgemein

Mittels einer Fourier-Transformation können wir unser Eingangssignal in eine Summe von Sinus- und Kosinus-Funktionen zerlegen.

Hierfür müssen folgende Bedingungen gelten (Dirichlet-Bedingungen):

1. Anzahl der Unstetigkeiten innerhalb einer Periode ist endlich
2. Anzahl der Maxima und Minima innerhalb einer Periode ist endlich
3. Funktion ist in jeder Periode integrierbar (d.h. die Fläche unter dem Betrag der Funktion ist in jeder Periode endlich)

Die Sinus- und Kosinus-Funktionen werden auch Basisfunktionen genannt und bilden einen Vektorraum.

### 2.3.2 Fourier-Transformation Grundideen

Nachfolgend werden die Grundideen der Fourier-Transformationen erläutert.

Die Ausgangsidee ist dass jedes Signal durch eine Summe von phasenverschobenen Kosinus-Funktionen beschreiben werden kann. Wir sprechen von einer Phasenverschiebung falls zwei Kosinus-Funktionen unterschiedliche Nullstellen haben (d.h. eine Verschiebung auf der x-Achse). Hierfür ergibt sich folgende Formel:  $s[i] = \sum_{k=0}^{N/2} M_k * \cos(2 * \pi * k * i / N + \phi_k)$  wobei  $i = 0, \dots, N-1$ ,  $M_k$ .

Die zweite Idee ist dass jede phasenverschobene Kosinus-Funktion  $M * \cos(x + \phi)$  kann durch eine Summe von Kosinus- und Sinus-Funktion ohne Phasenverschiebung repräsentiert werden:  $M * \cos(x + \phi) = A * \cos(x) + B * \sin(x)$  mit  $A = M * \cos(\phi)$  und  $B = M * \sin(\phi)$  wobei  $M$  ist die Amplitude und  $\phi$  die Phasenverschiebung. Dies erhalten wir durch den Übergang von Kartesischen-Koordinaten in Polar-Koordinaten in der komplexen Zahlenebene. Dadurch bestehen die Polarkoordinaten aus einem Imaginär und einem Realteil.

Da wir uns im diskreten Bereich befinden gilt folgende Eigenschaft: Das Signal aus  $N$  Werten ist ein  $N$ -dimensionaler Vektor. Wie vorher beschrieben bilden die gesuchten Basisfunktionen einen Vektorraum. Für einen  $N$ -dimensionalen Vektorraum benötigen wir also eine Basis mit  $N$  orthogonalen Vektoren (d.h. das Skalarprodukt zweier Basis-Vektoren muss 0 sein).

Die diskreten Sinus und Kosinus-Funktionen  $c_k[i] = \sin(2 * \pi * k * i / N)$  bzw.  $s_k[i] = \cos(2 * \pi * k * i / N)$  sind alle zueinander orthogonal. Da die Summe von 0 bis  $N/2$  läuft erhalten wir genau  $N/2 + 1$  phasenverschobene Kosinus-Funktionen, welche jeweils in eine Sinus und eine Kosinus Funktion zerlegt wird. Somit erhalten wir  $2 * (N/2 + 1) = N + 2$  Basisfunktionen. Da  $s_0 = \sin(0)$  und  $s_{N/2} = \sin(2 * \pi * N/2 * i / N) = \sin(\pi * i)$  jeweils Nullvektoren sind, können diese verworfen werden wodurch wir  $N$  Basisfunktionen erhalten.

Unter einer diskreten Fourier-Transformation verstehen wir die Transformation des Signalsvektors in ihre Sinus- und Kosinus-Basis.

### 2.3.3 Diskrete Fourier-Transformation

Man unterscheidet bei der diskreten Fourier-Transformation zwischen der Zeit-Domäne und der Frequenz-Domäne. Der Übergang von der Zeit-Domäne in die Frequenz-Domäne wird Diskrete-Fourier-Transformation (DFT) genannt. Der rückwärtige Übergang wird Invers-Diskrete-Fourier-Transformation (IDFT) genannt.

Die Zeit-Domäne  $x[]$  besteht aus  $N$ -Samples, welche von 0 bis  $N-1$  nummeriert sind. Die Frequenz-Domäne beinhaltet die durch die DFT erhaltenen Real-  $ReX[]$  und Imaginär-Teile  $ImX[]$ , welche jeweils aus  $N/2 + 1$  Elementen besteht. Die Real-Teile beschreiben Amplituden der Kosinus-Wellen, wobei die Imaginär-Teile die Amplituden der Sinus-Wellen beschreiben.

Die Basisfunktionen d.h. die Funktionen die ein Signal  $x$  zerlegen:

$$c_k[i] = \cos(2 * \pi * k * i / N)$$

$$s_k[i] = \sin(2 * \pi * k * i / N)$$

wobei  $k$  die Wellenzahl ist.  $c_k$  bzw.  $s_k$  ist das Signal der Kosinus- bzw. Sinusfunktion die mit der Amplitude in der Fourierzerlegung auftritt. Alle Basisfunktionen müssen genau so lang wie das Signal sein. Der Parameter  $k$  gibt die Anzahl der Zyklen innerhalb der Signallänge an.  $c_0 = Re[0]$  ist der Gleichstrom-Versatz (DC-Offset).  $s_0 = Im[0]$  und  $s_{N/2} = ImN/2$  sind überall 0, also irrelevant für die gesuchte Basis.

Bisher wissen wir welche Basisfunktionen in dem Signal enthalten sein können. Jedoch fehlt uns der Anteil der Basisfunktion in dem Ausgangssignal d.h. uns fehlt noch die konkrete Berechnung der Real- bzw. Imaginar-Teile (Amplituden der Sinus- bzw. Kosinus-Funktionen). Bevor wir uns damit befassen ist noch eine Vorüberlegung notwendig.

Da wir wissen wollen wie ähnlich unser Eingangssignal zu unserer Basis ist berechnen wir nachfolgend die Korrelation. Die allgemeine Formel ergibt sich wie folgt:

$$\sum_{i=0}^{N-1} x[i] * y[i]$$

mit  $x$  als Eingangssignal und  $y$  unsere Basis.

Ausgehend von dieser Beobachtung und unseren vorher ermittelten Basisfunktionen ergeben sich nachfolgend die Formeln für die Amplituden:

$$ReX[k] = \sum_{i=0}^{N-1} x[i] * \cos(2 * \pi * k * i / N)$$

bzw.

$$ImX[k] = \sum_{i=0}^{N-1} x[i] * \sin(2 * \pi * k * i / N)$$

für  $k = 0, \dots, N/2$ .

## 2.4 TODO

# 3 Merkmale

## 3.1 Statistische Merkmale

### 3.1.1 Erwartungswert

Der Erwartungswert einer diskreten Zufallsvariable  $x$  mit den möglichen Werten  $x_i$  und zugehörigen Wahrscheinlichkeiten  $P(x_i)$  ist:  $E[x] = \sum_{i=1}^n (x_i * P(x_i))$ .

Der Erwartungswert einer kontinuierlichen Zufallsvariable  $x$  mit Wertebereich  $X$  und zugehöriger Wahrscheinlichkeitsdichte  $p(x)$  ist:  $E[x] = \int_X (x * p(x)) dx$ .

Der Erwartungswert einer Zufallsvariable wird auch als Mittelwert  $\mu$  bezeichnet.

Für gleichverteilte diskrete Zufallsvariablen  $x$  mit Werten  $x_1, x_2, \dots, x_n$  gilt:  $P(x_i) = 1/n$  und somit:  $E[x] = \sum_{i=1}^n (x_i * \frac{1}{n}) = \frac{1}{n} * \sum_{i=1}^n x_i$ .

Für die Erwartungswerte von Funktionen einer Zufallsvariablen,  $f(x)$  gilt:  $E[f(x)] = \sum_{i=1}^n (f(x_i) * P(x_i))$  bzw.  $E[f(x)] = \int_X (f(x) * p(x)) dx$ .

### 3.1.2 Momente

Der  $r$ -te Moment einer Zufallsvariable  $x$  ist  $E[x^r]$ . Der erste Moment mit  $r = 1$ :  $E[x] = \mu$  heißt auch Mittelwert.

Der  $r$ -te zentrale Moment ist:  $\mu_r = E[(x - E[x])^r] = E[(x - \mu)^r]$ .

Das zweite zentrale Moment  $\mu_2 = E[(x - \mu)^2] = \sigma^2$  heißt auch Varianz;  $\sqrt{\sigma^2}$  heißt Standardabweichung  $\sigma$ .

Die Schiefe einer Verteilung ist ein Maß für ihre Asymmetrie:  $skew(x) = \frac{\mu_3}{\sigma^3} = \frac{E[(x - \mu)^3]}{\sigma^3}$ . Wenn  $skew(x) > 0$  linkssteil (rechtsschief) bzw.  $skew(x) < 0$  rechtssteil (linksschief).

Die Wölbung einer Verteilung ist ein Maß für ihre Spitzheit:  $kurt(x) = \frac{\mu_4}{\sigma^4}$ . Wenn  $kurt(x) < 3$  flach bzw.  $kurt(x) > 3$  spitz. Falls  $x$  normalverteilt gilt:  $kurt(x) = 3$ .

### 3.1.3 Empirische Werte

Die vorliegenden Messwerte  $x_1, \dots, x_n$  stellen eine Stichprobe aus der Zufallsvariablen  $x$  zugrunde liegenden Verteilung dar. Überlicherweise sind die wahren Parameter  $(\mu, \sigma)$  dieser Verteilung nicht bekannt. Man muss daher diese Parameter auf Basis der Stichprobe schätzen.

Der empirische Mittelwert:  $\bar{x} = \frac{1}{n} * \sum_{i=1}^n x_i$  wobei  $E[\bar{x}] = \mu$ .

Die empirische Varianz ist:  $s^2 = \frac{1}{n-1} * \sum_{i=1}^n (x_i - \bar{x})^2$  wobei  $E[s^2] = \sigma^2$ .

### 3.1.4 Lageparameter

Lageparameter treffen allgemeine Aussagen über die Position der Verteilung und stellen in gewisser Weise die Lage ihres Schwerpunkts dar. Verschiedene Lagemaße sind dabei unterschiedlich robust, zeigen sich also mehr oder weniger empfindlich gegenüber Ausreißerwerten. Beispiele: Mittelwert, Median, Modus.

Der Median ist der kleinste Wert  $x_m$ , bei dem die kumulative Verteilungsfunktion  $F_p(x) = \int_{-\infty}^x p(z) dz$  einen Wert von  $\geq 0.5$  liefert d.h. er teilt die Verteilungsfunktion in zwei (flächenmäßig) gleich große Teile.

Für eine geordnete Stichprobe  $x_1 \leq \dots \leq x_n$  ist der Median der Wert  $x_{(n+1)/2}$  (falls  $n$  ungerade) bzw. der Wert  $1/2 * (x_{n/2} + x_{n/2+1})$ . Daraus folgt dass der Median als Lagemaß wesentlich unempfindlicher ist gegenüber Ausreißer als der Mittelwert.

Der Modus einer Verteilung ist der Wert mit der größten Wahrscheinlichkeit. Gibt es nur einen Modus nur einen Modalwert nennt man die Verteilung unimodal, sonst besitzt Verteilung mehrere Modalwerte und nennt sie deshalb bimodal.

Es gilt:

- links-steile, rechts-schiefe Verteilung: Modus  $<$  Median  $<$  Mittelwert
- rechts-steile, links-schiefe Verteilung: Modus  $>$  Median  $>$  Mittelwert

### 3.1.5 Streuungsmaße

Streuungsmaße beschreiben die Breite bzw. die Ausdehnung einer Verteilung und somit die Abweichung vom Schwerpunkt. Sie sind somit ein Maß für die Variabilität der Daten. Beispiel: Varianz, Quartile, Interquartilsabstand.

- Quartile: drei Quartile teilen die geordnete Datenmenge in vier gleich große Segmente, wobei das zweite Quartil gleichzeitig den Median darstellt
- Interquartilsabstand bezeichnet den Abstand zwischen den 1. und den 3. Quartil und umfasst also die Hälfte der Daten und ist analog zum Median, robuster gegen Ausreißer als die Varianz-

### 3.1.6 Standardisierung

- Problem: Normalverteilung  $N(\mu, \sigma^2)$  hat nicht zwangsläufig eine Fläche von 1 unter der Kurve
- Lösung: Transformation mit  $z = \frac{x-\mu}{\sigma}$  dadurch Normalverteilung  $N(0, 1)$  als Ergebnis
- Mahalanobis-Abstand:  $r = \frac{|x-\mu|}{\sigma}$  (Z-Score) (Musst die Distanz zwischen x und  $\mu$  in Einheiten der Standardabweichung)
- wenn  $z = \frac{x-\mu}{\sigma}$  dann gilt:
  - $skew(x) = \frac{E[(\frac{x-\mu}{\sigma})^3]}{\sigma^3} = E[(\frac{x-\mu}{\sigma})^3] = E[z^3]$
  - $kurt(x) = \frac{E[(\frac{x-\mu}{\sigma})^4]}{\sigma^4} = E[(\frac{x-\mu}{\sigma})^4] = E[z^4]$
  - d.h. Schiefe und Wölbung von x sind das 3 bzw. 4 Moment der standardisierten Verteilung x
- für multivariaten Fall mit  $\Sigma$  die Kovarianzmatrix gilt:  $z = \Sigma^{-1/2} * (x - \mu)$  d.h. aus  $N(\mu, \Sigma)$  wird  $N(0, I)$

### 3.1.7 Korrelation

- x,y Zufallsvariablen und  $\mu_x, \mu_y$  die Erwartungswerte und  $\sigma_x, \sigma_y$  die Standardabweichung
- Korrelationskoeffizient  $p_{xy}$  ist ein Maß für die lineare Abhängigkeit
- $p_{xy} = \frac{E[(x-\mu_x)*(y-\mu_y)]}{\sigma_x * \sigma_y}$
- wenn  $|p_{xy}| = 1$  dann lineare Abhängigkeit zwischen x und y
- wenn x und y unabhängig  $\rightarrow p_{xy} = 0$  anders rum nicht
- empirische Korrelation  $r_{xy}$  zweier Stichproben  $x = x_i, y = y_i$  misst die Abhängigkeit zweier Stichproben:
 
$$r_{xy} = \frac{x' * y'}{||x'|| * ||y'||}$$
 mit  $x'_i = x_i - \bar{x}, y'_i = y_i - \bar{y}$

### 3.1.8 Einsatz statistischer Merkmale

- problemunabhängig d.h. können immer eingesetzt werden da kein Vorwissen über die Problemstruktur

## 3.2 Merkmalstypen

Niveau	Operationen	Lageparameter	Beispiel
nominal	$\{=, \neq\}$	Modus	Geschlecht
ordinal	$\{>, <\}$	+Median	Bundesligatabelle
intervall	$\{-, +\}$	+Erwartungswert	Geburtsjahr
ratio	$\{/, *\}$	+geom. Mittel	Wohnfläche

- Intervall und Ratio Skalen sind metrisch d.h. Abstandsbegriff ist sinnvoll
- Addition auf Intervallskalen für Abstände sinnvoll
- Ratioskalen haben Nullpunkt
- Nominal- und Ordinalskalen sind immer diskret

## 3.3 Merkmale für Zeitreihen

### 3.3.1 Summarische Merkmale

- Gegeben ein Signal  $x = x_1, \dots, x_n$
- Zero Crossing Rate (im Ortsbereich):  $zcr(x) = \frac{1}{n-1} \sum_{i=2}^n \mathcal{I}\{x_i * x_{i-1} < 0\}$  mit  $\mathcal{I}\{A\} = 1$  falls A wahr
- Energie (im Orts- und im Frequenzbereich):  $en(x) = \sum_{i=1}^n x_i^2$
- Entropie (im Orts- und im Frequenzbereich):  $ent(x) = - \sum_{i=1}^n x_i^* * \log(x_i^*)$  wobei  $x_i^* = \frac{x_i}{\sum_{j=1}^n x_j}$
- Schwerpunkt (spectral centroid)(Frequenzbereich):  $sc(x) = (\sum_{k=1}^K f_k * |x_k|^2) / (\sum_{k=1}^K |x_k|^2)$
- Bandbreite (bandwidth)(Frequenzbereich)  $bw(x) = (\sum_{k=1}^K (f_k - sc)^2 * |x_k|^2) / (\sum_{k=1}^K |x_k|^2)$

### 3.3.2 Autokorrelation, Grundfrequenz

- Signal  $x = x_1, \dots, x_n$
- Autokorrelation:  $R_x(\tau) = (x \star x)(\tau) = \sum_i^n x_i * x_{i-\tau}$
- $\tau$  = Verzögerungsparameter (Lag)
- üblich Autokorrelation:  $ACF_x(\tau) = \frac{R_x(\tau)}{R_x(0)}$
- Verwendung: Bestimmung der Grundfrequenz – > das zweite Maximum der ACF liefert die  $1/f_0$  Periodendauer
- d.h. Tau als Parameter  $\rightarrow$  durchprobieren bis zweite Maximum gefunden, Begründung:  $\tau = 0$  ist immer das erste Maxima
- Maximum des Fourierspektrums nicht geeignet da im niedrigen Frequenzbereich nur eine grobe Auflösung

### 3.3.3 Phasendifferenz

- zwei Sinus Signale:  $x_i = \sin(w * i), y_i = \sin(w * i + \phi)$  d.h. gleiche Frequenz aber Phasenverschiebung
- $\phi = \arccos(r_{xy})$

## 3.4 Merkmale für kinetische Systeme ????

## 4 Bayessche Entscheidungstheorie

### 4.1 Einführung

- Natur hat einen Zustand  $\omega$  aus einer Wertemenge  $\Omega = \{\omega_1, \omega_2, \dots\}$
- Beispiel  $\omega$ : Fisch ist Wolfsbarsch ( $\omega_1$ ) oder Lachs( $\omega_2$ )
- $\omega$  ist meistens unbekannt d.h. eine Zufallsvariable
- Wenn es gleich viele Lachse und Wolfsbarsche gibt, ist es gleichwahrscheinlich die eine oder andere Art zu tippen
- a priori Wahrscheinlichkeit = Vorwissen
- $P(\text{"Barsch"}) + P(\text{"Lachs"}) = 1$
- wenn wir nur die a priori Wahrscheinlichkeiten kennen ist die Wahl mit der größten Wahrscheinlichkeit immer die beste
- Typischerweise haben wir mehr Informationen d.h. zusätzlichen Wissen für unsere Entscheidung
- Beispiel: die Helligkeit (Merkmal  $x \in \mathbb{R}$ ) hängt von der Fischart  $\omega$  ab d.h. wir haben klassenabhängige Wahrscheinlichkeiten
- $p(x|\omega)$  ist die klassenabhängige Wahrscheinlichkeit des Merkmals x
- uns interessiert  $P(\omega_i|x)$  also die Wahrscheinlichkeit dass die Welt den Zustand  $\omega_i$  hat nachdem wir x beobachtet haben
- Verwendung: Satz von Bayes:  $P(\omega_i|x) = \frac{p(x|\omega_i) * P(\omega_i)}{p(x)}$  mit  $p(x) = \sum_{i=1}^2 p(x|\omega_i)P(\omega_i)$  (für unseren Fall von zwei Fischen)
  - $P(\omega_i|x)$ : Die Wahrscheinlichkeit des wahren Zustands, nachdem wir den Beweis haben (Posteriori)
  - $p(x|\omega_i)$ : Die Passfähigkeit des Beweises x zum möglichen wahren Zustand  $\omega$  (Likelihood)
  - $P(\omega_i)$ : Die Wahrscheinlichkeit des wahren Zustands bevor wir den Beweis gesehen haben
- Nun Entscheidung für  $\omega$  mit der größten Posteriori  $P(\omega_i|x)$



## 4.2 Bayes'sche Entscheidungstheorie

### 4.2.1 Definitionen

- Merkmalsvektor  $x$  als Element eines  $d$ -dimensionalen Euklidischen Merkmalsraums  $R^d$  d.h.  $x \in R^d$
- $\Omega = \{\omega_1, \omega_2, \dots, \omega_c\}$  eine endliche Menge von möglichen Zuständen (Klassen, Kategorien)
- $A = \{\alpha_1, \dots, \alpha_a\}$  eine endliche Menge von möglichen Aktionen
- Eine Kostenfunktion  $\Lambda(\alpha_i|\omega_j)$  die angibt welche Kosten die Aktion  $\alpha_i$  verursacht wenn der Zustand  $\omega_j$  ist

### 4.2.2 Bayes'sche Entscheidungstheorie - formal

- $p(x|\omega_j)$  und  $P(\omega_j)$  sind wie vorher die klassenbedingte Wahrscheinlichkeit von  $x$  gegeben  $\omega_j$  bzw. die a-priori Wahrscheinlichkeit von  $\omega_j$
- Es gilt analog für die a-posteriori-Wahrscheinlichkeit:  $p(\omega_j|x) = \frac{p(x|\omega_j)*P(\omega_j)}{p(x)}$  wobei  $p(x) = \sum_{i=1}^c p(x|\omega_i)P(\omega_i)$
- gegeben ein Merkmalsvektor  $x$ . Nun soll eine Aktion  $\alpha_i$  durchzuführen. Welche Kosten entstehen dabei?
- Wenn Zustand  $\omega_j$  ist sind die Kosten von  $\alpha_i$  nach Def:  $\Lambda(\alpha_i|\omega_j)$
- Zustand  $\omega_j$  ist jedoch nicht gegeben, wir kennen nur  $P(\omega_j|x)$ . Können aber den Erwartungswert der Kosten d.h. das bedingte Risiko  $R(\alpha_i|x)$  bestimmen:  $R(\alpha_i|x) = E_{\omega|x}[\Lambda(\alpha_i|\omega)] = \sum_{j=1}^c \Lambda(\alpha_i|\omega_j)P(\omega_j|x)$
- Bayessche Entscheidungsregel besagt nun: wenn  $x$  gegeben, wähle Aktion  $\alpha_i$  für die das bedingte Risiko  $R(\alpha_i|x)$  minimal ist
- sei  $\alpha(x)$  eine Entscheidungsregel welche für ein gegebenes  $x$  eine Aktion  $\alpha_i$  auswählt
- dann ist das Gesamtrisiko unter dieser Entscheidungsregel:  $R_\alpha = \int R(\alpha(x)|x)p(x)dx$
- $R_\alpha$  wird minimal wenn man für jeden Punkt  $x$  das Minimum wählt d.h. die Aktion  $\alpha_i$  für die  $R(\alpha_i|x)$  minimal ist
- das minimale unvermeidbare Risiko  $R^*$  heißt Bayes-Risiko

### 4.2.3 Anwendung

- ???

### 4.2.4 Klassifikation und Diskriminanten

- Klassifikatoren können durch Diskriminantenfunktionen repräsentiert werden
- Klassifikator für die Klassen  $\omega_1, \dots, \omega_c$  besteht aus einer Menge von Diskriminantenfunktionen  $g_i(x), i \in \{1, \dots, c\}$
- dieser Klassifikator weist  $x$  einer Klasse  $\omega_i$  gdw.  $g_i(x) > g_j(x), \forall j \neq i$
- Bayes-Klassifikator für den allgemeinen Fall mit Risiko kann wie folgt definiert werden:  $g_i(x) = -R(\alpha_i|x)$
- für die Klassifikation mit minimalen Fehler vereinfacht sich dies zu:  $g_i(x) = P(\omega_i|x) \propto p(x|\omega_i)P(\omega_i)$
- für eine gegebene Menge von Diskriminanten  $g_i(x)$  liefert eine Transformation mit einer monoton wachsenden Funktion  $f()$  eine äquivalente Menge von Diskriminanten  $f(g_i(x))$
- dadurch alternative Formulierungen:
  - $g_i(x) = P(\omega_i|x) = \frac{p(x|\omega_i)*P(\omega_i)}{p(x)}$
  - $g_i(x) = p(x|\omega_i) * P(\omega_i)$
  - $g_i(x) = \ln(p(x|\omega_i)) * \ln(P(\omega_i))$
- Ein Klassifikator (Entscheidungsregel) für  $c$  Klassen zerlegt den Merkmalsraum  $\mathbb{R}^d$  in maximal  $c$  Entscheidungsregionen  $R_i, i \in \{1, \dots, c\}$
- Falls für einen Merkmalsvektor  $x$  gilt  $g_i(x) > g_j(x), \forall j \neq i$  dann liegt  $x$  in der Entscheidungsregion  $R_i$  und  $x$  wird als  $\omega_i$  klassifiziert

- verschiedene Entscheidungsregionen werden durch Entscheidungsgrenzen voneinander getrennt
- Entscheidungsgrenzen sind die Bereiche in denen die beiden größten Diskriminanten denselben Wert annehmen

#### 4.2.5 Normalverteilung

- Formel (univariant):  $p(x) = N(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} \exp[-\frac{1}{2}(\frac{x-\mu}{\sigma})^2]$
- Mittelwert (erster Moment):  $\mu$ , Varianz (zweite zentrale Moment):  $\sigma^2$
- Der zentrale Grenzwertsatz: Die Summe von n unabhängigen Zufallsvariablen konvergiert gegen Normalverteilung mit  $n \rightarrow \infty$ . Da viele natürliche Vorgänge einer großen Zahl unabhängiger Störfaktoren unterliegen, ist die Normalverteilungsannahme sinnvoll.
- hat hohe Entropie d.h. die Verteilung mit der größten Unsicherheit über die Werte einer Zufallsstichprobe
- Formel (multivariant):  $N(x; \mu, \Sigma) = \frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} * \exp[-\frac{1}{2} * (x - \mu)^t \Sigma^{-1} (x - \mu)]$ 
  - $x = (x_1, \dots, x_d)^t$  ein d-dimensionaler Spaltenvektor
  - $\mu = (\mu_1, \dots, \mu_d)^t$  ein d-dimensionaler Mittelwert-Vektor
  - $\Sigma$  ist eine  $d \times d$  Kovarianzmatrix,  $|\Sigma|$  die Diskriminante,  $\Sigma^{-1}$  Inverse Matrix
  - $\mu = E[x] = \int x * p(x) dx$  (können Komponentenweise bestimmt werden  $\mu_i = E[x_i]$ )
  - $\Sigma = E[(x - \mu)(x - \mu)^t] = \int (x - \mu)(x - \mu)^t p(x) dx$  ( $\sigma_{ij} = E[(x_i - \mu_i)(x_j - \mu_j)]$ )
- für Kovarianzmatrix  $\Sigma$  gilt:
  - $\sigma_{ii}$  sind die Varianzen der entsprechenden  $x_i$  also  $\sigma_i^2$
  - $\sigma_{ij}$  sind Kovarianzen von  $x_i$  und  $x_j$ , Wenn  $x_i$  und  $x_j$  unabhängig dann gilt  $\sigma_{ij} = 0$
  - falls alle nicht-diagonalelemente Null sind, ist die multivariate Verteilung einfach das Produkt der d univarianten Verteilungen der Komponenten von x
  - die Kovarianzmatrix ist symmetrisch und positiv (semi-) definit (M positiv definit:  $x^t M x > 0 \forall x \neq 0$ )
- Stichproben aus multivarianter Gaußverteilungen bilden typischerweise eine Häufung deren Mittelpunkt von  $\mu$  und deren Form von  $\Sigma$  definiert wird
- Orte gleicher Wahrscheinlichkeitsdichte liegen auf Hyperellipsoiden für die  $(x - \mu)^t \Sigma^{-1} (x - \mu)$  konstant ist
- die Hauptachsen der Hyperellipsoide sind die Eigenvektoren von  $\Sigma$ , die Eigenwerte geben die Länge dieser Achsen an
- $r^2 = (x - \mu)^t \Sigma^{-1} (x - \mu)$  ist die quadrierte Mahalanobis-Distanz; Konturen konstanter Dichte sind Hyperellipsoide konstanter Mahalanobis-Distanz zu  $\mu$

#### 4.3 Diskriminanten für die Normalverteilung

- vorher gezeigt dass Diskriminante  $g_i(x) = \ln(p(x|\omega_i)) * \ln(P(\omega_i))$  Klassifikation mit minimaler Fehlerrate liefert
- wenn  $p(x|\omega_i) = N(\mu_i, \Sigma_i)$  einsetzen in  $g_i(x)$ :  $g_i(x) = -\frac{1}{2}(x - \mu_i)^t \Sigma_i^{-1} (x - \mu_i) - \frac{d}{2} \ln(2\pi) - \frac{1}{2} \ln(|\Sigma_i|) + \ln(P(\omega_i)) \rightarrow$  Entfernung von  $-\frac{d}{2} \ln(2\pi)$  da konstant und nicht von der Klasse abhängt
- Also erhalten wir:  $g_i(x) : g_i(x) = -\frac{1}{2}(x - \mu_i)^t \Sigma_i^{-1} (x - \mu_i) - \frac{1}{2} \ln(|\Sigma_i|) + \ln(P(\omega_i))$
- Entscheidungsgrenzen:
  - $\Sigma_i = \sigma^2 I$  d.h. Alle Merkmale unabhängig, gleiche Varianzen, auf der Hauptdiagonalen stehen die gleichen Werte; Die Klassen bilden kugelförmige Häufungen gleicher Größe um ihre jeweiligen Mittelpunkte.
    - \* es gilt:  $\Sigma_i^{-1} = (1/\sigma^2)I$  und  $|\Sigma_i| = \sigma^{2d}$
    - \* einsetzen in Formel:  $g_i(x) = -\frac{1}{2\sigma^2}(x - \mu_i)^t (x - \mu_i) - \frac{1}{2} \ln(\sigma^{2d}) + \ln(P(\omega_i))$
    - \* Konstanten entfernen:  $g_i(x) = -\frac{1}{2\sigma^2}(x - \mu_i)^t (x - \mu_i) + \ln(P(\omega_i))$
    - \* Euklidische Distanz:  $(x - \mu_i)^t (x - \mu_i) = \|x - \mu_i\|^2$

- \* Also:  $g_i(x) = -\frac{\|x - \mu_i\|^2}{2\sigma^2} + \ln(P(\omega_i))$
- \* Expandieren:  $g_i(x) = -\frac{1}{2\sigma^2}(x^t x - 2\mu_i^t x + \mu_i^t \mu_i) + \ln(P(\omega_i))$
- \* Also erhalten wir eine lineare Diskriminante:  $g_i(x) = w_i^t * x + w_{i0}$  (lineare Maschine,  $w_{i0}$  ist Schwellwert)
- \* d.h. die Entscheidungsgrenzen zwischen den Regionen werden durch die lineare Gleichung:  $g_i(x) = g_j(x)$  definiert (Hyperebenen)
- \* die Hyperebene ist orthogonal zum Vektor  $w$ , der Verbindungsline zwischen den beiden Mittelwertvektoren
- \* Line wird in  $x_0$  geschnitten falls die a-priori-Wahrscheinlichkeiten gleich sind, liegt  $x_0$  zwischen  $\mu_i$  und  $\mu_j$
- $\Sigma_i = \Sigma$  d.h. alle Klassen haben gleiche Kovarianzmatrix d.h. Die Klassen bilden hyperellipsoide Häufungen gleicher Größe, Form und Orientierungen um ihre jeweiligen Mittelpunkte.
  - \* wieder einsetzen in  $g_i$ :  $g_i(x) = -\frac{1}{2}(x - \mu_i)^t \Sigma_i^{-1} (x - \mu_i) - \frac{1}{2} \ln(|\Sigma_i|) + \ln(P(\omega_i))$
  - \* Konstanten entfernen:  $g_i(x) = -\frac{1}{2}(x - \mu_i)^t \Sigma_i^{-1} (x - \mu_i) + \ln(P(\omega_i))$
  - \* Ausmultiplizieren und Symmetrie von  $\Sigma$  liefert wieder lineare Maschine
  - \* nicht mehr orthogonal zur Verbindungsline der Mittelwerte, bei gleicher a-priori-Wahrscheinlichkeit  $x_0$  immer noch auf halben Wege zwischen  $\mu_i$  und  $\mu_j$
- $\Sigma_i = \text{beliebig}$ 
  - \* einsetzen und ausmultiplizieren ergibt Quadratische-Diskriminante
  - \* beliebige Normalverteilungen führen zu Entscheidungsgrenzen welche allgemeine Hyperquadriken sind
  - \* Bereits kleine Anzahl von Klassen können die Entscheidungsgrenzen komplexe Formen annehmen lassen

## 4.4 Empirischer Fall

- im Anwendungsfall eher unrealistisch das man alle Parameter kennt d.h. muss diese auf Basis von Trainingsdaten schätzen
- wenn man Normalverteilung annehmen möchte:
  - $\hat{P}(\omega_i) = \frac{n_i}{n}$
  - $\hat{\mu}_i = \frac{1}{n_i} \sum_{j=1}^{n_i} x_{ij}$
  - $\hat{\Sigma} = \frac{1}{n-c} \sum_{i=1}^c \sum_{j=1}^{n_i} (x_{ij} - \hat{\mu}_i)(x_{ij} - \hat{\mu}_i)^t$
  - mit  $n_i$  Anzahl der Trainingsdatensätze für die Klasse  $i$ ,  $n$  Anzahl aller Trainingsdatensätze und  $x_{ij}$  Trainingsdatensatz Nummer  $j$  für Klasse  $i$
  - dies nennt man Lineare Diskriminanzanalyse
- für Fall 3:
  - $\hat{P}(\omega_i) = \frac{n_i}{n}$
  - $\hat{\mu}_i = \frac{1}{n_i} \sum_{j=1}^{n_i} x_{ij}$
  - $\hat{\Sigma}_i = \frac{1}{n_i - 1} \sum_{j=1}^{n_i} (x_{ij} - \hat{\mu}_i)(x_{ij} - \hat{\mu}_i)^t$
  - dies nennt man Quadratische Diskriminanzanalyse
- Bias/Variance Tradeoff = ???

## 4.5 Diskrete Merkmale

- Integrale durch Summen ersetzen, sonst alles wie bisher

# 5 Parameterschätzung

## 5.1 Einführung

- Ziel: Konstruieren eines Klassifikators
- Benötigen: a-priori-Verteilung  $P(\omega_i)$  und klassenbedingten Verteilungen  $p(x|\omega_i)$

- Problem: Verteilungen sind unbekannt
- haben Anzahl an Stichproben  $D = (x_1, \dots, x_n)^t$  mit dazugehörigen Klassen  $t = (\omega_{i1}, \dots, \omega_{in})^t$  mit  $D$  ist eine  $n \times d$  Matrix,  $d$  ist die Anzahl der Dimensionen des Merkmalsraums. Jede Zeile von  $D$  ist eine Beobachtung  $x_i$
- $n_i$  Anzahl der Trainingsdaten der Klasse  $\omega_i$  und es gilt  $n = \sum_{i=1}^c n_i$
- $P(\omega_i)$  können wir schätzen durch z.B. Zahlen der Häufigkeiten:  $\hat{P}(\omega_i) = \frac{n_i}{n}$
- $p(x|\omega_i)$  ist das Problem wir müssen eine Funktion  $f(x)$  schätzen d.h. den Funktionswert an unendlich vielen Stellen  $x \in \mathbb{R}^d$ , Problem: Raumvolumen wächst exponentiell  $\rightarrow$  bei einer festen Anzahl von Trainingsdaten sind  $x_i$  immer weiter im Raum verstreut d.h. die leeren Bereiche welche  $f(x)$  raten muss werden immer größer
- Lösung: Annahme das  $f(x)$  keine beliebige Form hat sondern aus einer Funktionsfamilie  $f(x|\theta)$  stammt, aus der die gesuchte Funktion durch einen Parametervektor  $\theta$  mit niedriger Dimensionalität bestimmt werden wird
- Beispiel:  $f(x|\theta) = N(x|\mu, \Sigma)$  mit  $\theta = (\mu, \Sigma)$  d.h. wir müssen nicht mehr unendlich viele Werte aus  $\mathbb{R}$  schätzen sondern nur noch wenige Parameter
- Aufgabe: Auf Basis der Annahme der parametrischen Form  $p(x|\theta_i)$  der Verteilungsfunktion  $p(x|\omega_i)$  bestimme den Parametervektor  $\theta_i$  mit Hilfe der Trainingsdaten  $D^i, t^i$
- Lösungen:
  - Maximum Likelihood (Punktschätzung)
  - Bayes'sch (Bestimmung der a-posteriori Verteilung von  $\theta$ )
- Also: Wir haben Trainingsdaten  $D = x_1, \dots, x_n$  und wir nehmen an, dass  $x_k$  aus einer parametrischen Verteilung  $p(x_k|\theta)$  stammen. Gesucht ist nun eine Schätzung für die wahren Parameter die am besten mit den Trainingsdaten vereinbar ist

## 5.2 Maximum Likelihood-Schätzer

- Annahme: der Parametervektor  $\theta$  hat einen wohldefinierten festen Wert der uns unbekannt ist
- Maximum-Likelihood-Schätzer:  $\hat{\theta} = \operatorname{argmax}_{\theta \in \Theta} p(D|\theta)$  also derjenige Wert  $\hat{\theta}$  der die Likelihood der Trainingsdaten maximiert, Anwendung Satz von Bayes:  $p(\theta|D) = \frac{p(D|\theta)p(\theta)}{p(D)}$
- Frage: Wo kommt der her? Man erwartet eigentlich  $p(\theta|D)$
- Antwort: eigentlich interessiert uns  $p(\theta|D)$  aber uns interessiert nicht die gesamte Verteilung sondern nur der Wert  $\operatorname{argmax}_{\theta} p(\theta|D)$
- Annahme: Alle Werte von  $\theta$  sind gleich wahrscheinlich  $p(\theta) = c$
- also  $\operatorname{argmax}_{\theta} p(\theta|D) = \operatorname{argmax}_{\theta} \frac{p(D|\theta)p(\theta)}{p(D)} = \operatorname{argmax}_{\theta} \frac{p(D|\theta)c}{c} = \operatorname{argmax}_{\theta} p(D|\theta) = \hat{\theta}_{ML}$
- Gegeben:
  - Menge von Daten:  $D = x_1, x_2, \dots, x_n$
  - parametrische Verteilungsfunktion:  $p(x|\theta)$
  - $p$ -dimensionalen Parametervektor:  $\theta = (\theta_1, \dots, \theta_p)^t$
- Gesucht:  $\hat{\theta}$  der  $p(D|\theta)$  maximiert
- Annahme:  $x_k$  sind unabhängige Stichproben aus  $p(x|\theta)$  d.h. es gilt:  $p(D|\theta) = \prod_{k=1}^n p(x_k|\theta)$
- Vorgehen:
  1. Bestimme die Log-Likelihood-Funktion:  $l(\theta) = \log \prod_{k=1}^n p(x_k|\theta) = \sum_{k=1}^n \log(p(x_k|\theta))$
  2. Wende Gradientenoperator  $\nabla_{\theta} = (\frac{\partial}{\partial \theta_1}, \dots, \frac{\partial}{\partial \theta_p})^t$  auf  $l(\theta)$  an d.h. bestimme:  $\nabla_{\theta} l = \sum_{k=1}^n \nabla_{\theta} \ln(p(x_k|\theta))$
  3. Löse die Gleichung:  $\nabla_{\theta} l = 0$
  4. Lösung der Gleichung ist  $\hat{\theta}$

- Bias: Der Erwartungswert der Schätzwerte entspricht nicht den realen Parametern. ML-Schätzer sind jedoch asymptotisch erwartungstreu
- Maximum-Likelihood ist sehr einfach anzuwenden
- Wenn die a-posteriori Verteilung unimodal ist, dann liefert ML Ergebnisse die ähnlich gut sind wie Bayes'sche Schätzung, ML konvergiert mit  $n \rightarrow \infty$  gegen die wahren Parameter

### 5.3 Bayessche Parameterschätzung

- Aufgabe: Bestimme  $p(x|D)$  anhand einer Stichprobe  $D$  die aus einer parametrischen Verteilung  $p(x|\theta)$  gezogen wurde. Dabei habe  $p(x|\theta)$  eine bekannte funktionale Form, nur  $\theta$  ist nicht festgelegt
- Wir stellen uns  $\theta$  als Zufallsvariable vor die eine a-priori-Verteilung (vorher  $\theta$  unbekannt und fest jetzt unbekannt und aus einer Verteilung)  $p(\theta)$  hat aus der wir eine a-posteriori-Verteilung  $p(\theta|D)$  berechnen können
- es ergibt sich folgende a-posteriori Verteilung für die Daten:  $p(x|D) = \int p(x|\theta)p(\theta|D)d\theta$
- Vergleich zu ML-Ansatz:  $p(x|D) = p(x|\hat{\theta}_{ML})$  mit  $\hat{\theta}_{ML} = \operatorname{argmax}_{\theta} p(D|\theta)$
- Ziel: Bestimmung der Parameterverteilung  $p(\theta|D)$
- Bayessche Satz:  $p(\theta|D) = \frac{p(D|\theta)p(\theta)}{\int p(D|\theta)p(\theta)d\theta}$  mit  $p(D|\theta)$  ist Likelihood
- Annahme: Unabhängigkeitsannahme für  $D$  d.h. die Merkmale hängen nicht voneinander ab es gilt also:  $p(D|\theta) = \prod_{k=1}^n p(x_k|\theta)$

## 6 Dimensionsreduktion

### 6.1 Einführung

- wenn Merkmale unabhängig sind dann lassen sich theoretisch Klassifizierer mit sehr hoher Präzision realisieren
- für zwei-Klassen-Problem  $p(x|\omega_i) = N(\mu_i, \Sigma)$  der Bayes-Fehler:  $P(e) = \frac{1}{\sqrt{2*\pi}} \int_{r/2}^{\infty} e^{-u^2/2} du$  mit  $r^2$  ist die quadrierte Mahalanobis-Distanz
- mit wachsendem  $r$  sinkt der Fehler, bei unabhängigen Merkmalen ist  $\Sigma = \operatorname{diag}(\sigma_1^2, \dots, \sigma_d^2)$
- man erhält:  $r^2 = \sum_{j=1}^d \frac{(\mu_1^{(j)} - \mu_2^{(j)})^2}{\sigma_j^2}$  d.h. je größer  $d$  (Anzahl der Dimensionen / Merkmale) desto besser ist die Klassifikation (theoretisch, praktisch eher nicht so)
- im Bayesschen Framework, wenn wir die Formen der Verteilungen kennen können wir die Anzahl der Features beliebig erhöhen und der Ansatz sorgt von selbst dass diese für irrelevante Features entsprechend gering gewichtet werden
- Probleme:
  - in hohen Dimensionen sind bereits geringe Unterschiede zwischen wahrer und angenommener Verteilung sehr groß
  - in hohen Dimensionen ist die Anzahl der Trainingsdaten nicht mehr ausreichend um die erforderliche Anzahl der Parameter ausreichend zuverlässig zu schätzen
- Lösung:
  - Hauptkomponentenanalyse (PCA): Projektion in einem Raum mit reduzierter Dimensionalität die die Daten am besten repräsentiert
  - Fisher-Diskriminante: Projektion in einem Raum mit reduzierter Dimensionalität in der die Daten unterschiedlicher Klassen am besten getrennt sind

## 6.2 Hauptkomponentenanalyse

- Aufgabe: reduziere die Anzahl der Dimensionen auf  $q < d$
- Ansatz: finde einen  $q$ -dimensionalen Teilvektorraum  $E^q$  mit Basis:  $E = (e_1, \dots, e_q)^t$  so dass die Projektion der  $x_k$  in den Raum  $E^q$  möglichst gut repräsentiert sind d.h einen minimalen quadratischen Fehler verursacht
- Kriterium zu minimieren:  $J(E) = \sum_{k=1}^n (E^t(Ex_k) - x_k)^2$
- $E$  ist eine  $q \times d$  Matrix die Vektoren von einem  $d$ -dimensionalen Vektorraum  $E^d$  in einen  $q$ -dimensionalen Untervektorraum  $E^q$  abbildet
- $x'_k = Ex_k$  die Projektion eines  $d$ -dimensionalen Vektors  $x_k$  in  $E^q$
- $x''_k = E^t x'_k$  ist die Rücktransformation des reduzierten Vektors in den ursprünglichen Vektorraum, in dem der Fehler (Distanz zwischen  $x''_k$  und  $x_k$ ) gemessen
- $q$  kann nicht größer als  $n$  sein da  $n$  Datenvektoren spannen höchstens  $n$ -dimensionalen Vektorraum auf
- Vorgehen:
  1. Berechne empirische Kovarianzmatrix  $\hat{\Sigma}$
  2. Berechne die Eigenwerte  $\lambda_i$  und Eigenvektoren  $e_i$  von  $\hat{\Sigma}$
  3. wähle  $q < d$  Eigenvektoren zu den größten Eigenwerten als dimensionsreduzierten Merkmalsraum mit Transformationsmatrix mit  $E = (e_1, \dots, e_q)^t$
  4. Die neuen  $q$ -dimensionalen Merkmalsvektoren  $x'$  ergeben sich jetzt durch die Transformation  $x' = Ex$

## 6.3 Diskriminantenanalyse

- Aufgabe: Suche eine Transformation die das Verhältnis der Varianz zwischen den Klassen zur Varianz innerhalb der Klasse maximiert
- Die Transformation erzeugt Räume mit maximal  $c - 1$  Dimensionen da es nur  $c$  Klassen gibt, können  $c$  Klassenmittelpunkte einen Raum mit maximal  $c - 1$  Dimensionen aufspannen
- wir definieren:
  - Streuung innerhalb der Klassen:  $S_W = \sum_{i=1}^c \sum_{k=1}^{n_i} (x_{ik} - \mu_i)(x_{ik} - \mu_i)^t$
  - Streuung zwischen den Klassen:  $S_B = \sum_{i=1}^c n_i(\mu_i - \mu)(\mu_i - \mu)^t$
  - mit  $x_{ik}$  ist Datensatz  $k$  zu Klasse  $i$ ,  $n_i$  ist die Anzahl der Datensätze für Klasse  $i$ ,  $\mu$  ist der Mittelwert über alle Klassen
- Die Determinante einer Streuungsmatrix ist ein Maß für das Raumbolumen das die zugehörigen Daten beanspruchen
- Streuungsverhältnis:  $\frac{|S_B|}{|S_W|}$
- Gesucht: Transformation  $W$  in einen  $q \leq c - 1$  dimensionalen Raum  $E^q$  so dass das Streuungsverhältnis in  $E^q$  maximal wird,  $W$  ist eine  $q \times d$  Matrix
- mit  $W$  gegeben:  $\hat{S}_B = \sum_{i=1}^c n_i(W\mu_i - W\mu)(W\mu_i - W\mu)^t = \dots = W * S_B * W^t$  analog:  $\hat{S}_W = W * S_W * W^t$
- Aufgabe: minimiere  $J(W) = \frac{|W * S_B * W^t|}{|W * S_W * W^t|}$
- Lösung: Eigenwertproblem (ausgelassen) lösen

## 6.4 Nichtlineare Diskriminanten

- Gegeben: Feature Space welcher nicht linear trennbar ist
- Aufgabe: Finde einen Vektor  $\phi = (\phi_1, \dots, \phi_f)$  von  $f$  (nichtlinearen) Transformationen  $\phi_i(x)$  und betrachte lineare Diskriminante in  $f$ -dimensionalen Raum  $E^f$

## 7 Nicht-parametrische Methoden

### 7.1 Einführung

- Bisher:
  - vollständigen Wissen über die Verteilung: Bayessche Entscheidungsregel
  - oder: wir kennen zumindestens die funktionale Form der Verteilung und können die Parameter der Verteilung aus den Trainingsdaten schätzen
- Nun: Verteilungsfunktionen sind nicht bekannt
- Problemstellung:
  - Gesucht: unbekannte Verteilung  $p(x)$
  - Gegeben: Menge von Stichproben  $x_i \sim p(x)$  aus dieser Verteilung
  - Idee: Dichte der  $x_i$  in der Nähe von  $x$  verwenden um damit auf die Wahrscheinlichkeit  $p(x)$  zu schließen  $\rightarrow$  je dichter die  $x_i$  in einer Region liegen desto wahrscheinlicher dürfte es ja sein, Werte aus dieser Region zu erhalten vorausgesetzt  $p(x)$  ist nicht böartig zerklüftet
- Gegeben eine Region  $R \subset R^n$  im Merkmalsraum und eine Verteilungsfunktion  $p(x)$
- Wahrscheinlichkeit  $P$  dass ein  $x \in R^n$  in  $R$  liegt dann:  $P := \Pr(x \in R) = \int_R p(x') dx'$
- nun seien  $n$  Stichproben  $x_1, \dots, x_n$  mit  $x_i \sim p(x)$  gegeben
- Wahrscheinlichkeit dass genau  $k$  mit  $k < n$  dieser Stichproben in  $R$  liegen:  $P_k = \binom{n}{k} P^k (1 - P)^{n-k}$  (binomialverteilt)
- Erwartungswert:  $E[k] = \sum_k k \cdot P_k = n \cdot P$
- mit  $E[k]$  lässt sich nun  $P$  umgekehrt schätzen:  $P \approx \frac{k}{n}$
- Mit der Annahme dass  $p(x)$  ist kontinuierlich und einer Umgebung von  $x$  nicht allzu stark schwankt: Wenn  $V$  das Volumen der Region  $R$  ist dann gilt:  $k/n \approx P = \int_R p(x') dx' \approx \int_R p(x) dx' = p(x) \int_R 1 dx' = p(x) \cdot V$
- damit ergibt sich:  $p(x) \approx \frac{k/n}{V}$
- Bedeutung: wir können  $p(x)$  Abschätzen mit Hilfe einer Stichprobe der Größe  $n$ , wenn wir die Anzahl  $k$  der Samples in einer Region  $R$  um  $x$  in Relation zur Gesamtzahl zur Größe der Region  $V$  setzen
- zu zeigen: der Schätzwert kann die gesuchte Wahrscheinlichkeit  $p(x)$  beliebig genau approximieren wenn man  $n$  nur groß genug wählt d.h.:  $p(x) = \lim_{n \rightarrow \infty} \frac{k_n/n}{V_n}$  wenn man  $k_n$  und  $V_n$  geeignet definiert
- damit  $p(x) = \lim_{n \rightarrow \infty} \frac{k_n/n}{V_n}$  gilt müssen folgende Bedingungen gelten:
  - $\lim_{n \rightarrow \infty} V_n = 0$  d.h. der Fehler die Durchschnittsbildung im Raumvolumen verschwindet
  - $\lim_{n \rightarrow \infty} \frac{k_n/n}{V_n} = P$  d.h. die möglicherweise irrationale Zahl  $P$  kann durch die Anzahl der Samples in  $R$  beliebig genau approximiert werden
  - $\lim_{n \rightarrow \infty} \frac{k_n/n}{V_n} = 0$  mit schrumpfenden Raumvolumen sinkt auch die relative Anzahl von Treffern im Volumen
- Zwei Ansätze:
  - Parzen-Window-Ansatz (Kern-Dichte-Schätzer): Volumen  $V_n$  in Abhängigkeit von  $n$  festlegen und dann zeigen dass Konvergenz mit  $n \rightarrow \infty$  erreicht wird
  - k-nearest-Neighbour: Anzahl der Treffen in Abhängigkeit von  $n$  festlegen und hierfür Konvergenz zeigen

## 7.2 Parzen Windows

- Idee: um den Punkt  $x$ , dessen  $W_k$  wir bestimmen wollen, eine Testregion mit vorgegebene Volumen  $V_n$  legen und dann zählen, welcher Anteil  $k_n$  der Stichprobe in dieses Volumen fällt
- das Verhältnis  $k_n/n$  ist dann eine Schätzung für  $p(x)$ : Schätzung mit  $n$  Samples:  $p_n(x)$
- Annahme: Testregion  $R_n$  für  $n$  Stichproben ist Hyper-Cube mit Kantenlänge:  $h_n$  dann gilt:  $V_n = h_n^d$  d.h. Das Volumen ist die  $d$ -te Potenz der Kantenlänge ( $d$  = Dimensionenzahl)
- Vorgehen:
  - Fensterfunktion:  $\phi(u) = \begin{cases} 1 & |u_j| \leq 1/2; i = 1 \dots, d \\ 0 & \text{sonst} \end{cases}$  (stanzt einem im Ursprung zentrierten Hyperwürfel der Kantenlänge 1 aus)
  - für Hyperwürfel mit kantenlänge  $h_n$  zentriert um  $x$  gilt:  $\phi(\frac{x-x_i}{h_n}) = \begin{cases} 1 & \text{falls } x_i \text{ im Hyperwürfel um } x \text{ liegt} \\ 0 & \text{sonst} \end{cases}$
  - $x_i$  die Punkte welche sich im Hypercube befinden
  - Anzahl der Stichproben im Fenster:  $k_n = \sum_{i=1}^n \phi(\frac{x-x_i}{h_n})$
  - es folgt aus  $p_n(x) = \frac{k_n/n}{V_n}$  und  $k_n$ :  $p_n(x) = 1/n * \sum_{i=1}^n \frac{1}{V_n} \phi(\frac{x-x_i}{h_n})$
  - $p_n(x)$  ist der Durchschnitt von  $n$  Funktionen die von  $x$  und den  $x_i$  abhängen
- Erweiterungen:
  - $\phi(\frac{x-x_i}{h_n})$  kann unterschiedlich interpretiert werden: 1) Funktion von  $x_i$  die ihren Wert in Abhängigkeit davon liefert ob  $x_i$  im Fenster um  $x$  liegt ( $k_n$  = Anzahl der  $x_i$  im Fenster um  $x$ ) ODER 2) Funktion von  $x$  die ihren Wert in Abhängigkeit davon liefert ob  $x$  im Fenster um das jeweilige  $x_i$  liegt ( $k_n$  = Anzahl der Fenster um die  $x_i$  in denen  $x$  liegt)
  - Andere Fenster
    - \* Hyperwürfel mit 1/0 Zählung erzeugt Diskontinuität d.h scharfe Kanten
    - \* besser glatte Fenster: Gewicht des Samples sinkt kontinuierlich mit der Entfernung
    - \* Betrachtung von  $\phi$  als Kern (Kernel)  $K$ : für  $K$  gilt:  $\forall u : K(u) \geq 0; \int K(u) du = 1; \forall u : K(u) = K(-u)$
    - \* Epanechnikov-Kern minimiert den erwarteten integrierten quadratischen Fehler zwischen geschätzter und wahrer Verteilung:  $K_{ep}(u) = \begin{cases} \frac{d+2}{2^{*c_d}} (1 - ||u||^2) & \text{falls } ||u||^2 < 1 \\ 0 & \text{sonst} \end{cases}$
- Betrachtung als Faltung:
  - $f_n(x') = \sum_{i=1}^n \delta(x' = x)$  d.h. eine Stichprobe wird durch  $\delta$ -Funktionen repräsentiert
  - für einen Kern definieren wir:  $K_h(x, x') = \frac{1}{n*h} K(\frac{x-x'}{h})$
  - wir erhalten:  $p_n(x) = K_h(x, x') * f_n(x') = K_h(x, x') * \sum_{i=1}^n \delta(x' = x) = \dots = \frac{1}{n*h} \sum_{i=1}^n K(\frac{x-x_i}{h})$
  - d.h. Kern-Dichte-Schätzung ist die Filterung des Signals an einem Filter mit der Impulsantwort  $K_h$
- Flexibilität:
  - Parzen-Window Methode kann bei jeder Art von Verteilung angewendet werden
  - Anzahl von Dimensionen Grenzen gesetzt: die Anzahl der erforderlichen Samples wächst exponentiell mit  $d$  d.h.  $d > 3$  meist nicht mehr sinnvoll
- Klassifikation:
  - Bestimme  $p(x|w_i)$  mit Hilfe der Parzen-Window-Methode
  - Ordne einen neuen Punkt diejenigen Klasse zu, die die maximale a-posteriori-Wahrscheinlichkeit hat
  - Seien  $x_{ik}, k = 1, \dots, n_i$  die Samples der Klasse  $i$
  - klassenbedingte Wahrscheinlichkeit  $\hat{p}(x|w_i) = \frac{1}{n_i * h_i} \sum_{k=1}^{n_i} K(\frac{x-x_{ik}}{h_i})$
  - a-priori-Wahrscheinlichkeiten:  $\hat{p}(w_i) = \frac{n_i}{n}$
  - Klassifikation durch:  $\operatorname{argmax}_{w_i} \hat{p}(x|w_i) \hat{p}(w_i)$
  - Kombination mit Naiven-Bayes-Klassifikator: Annahme dass die einzelnen Merkmale voneinander unabhängig sind:  $p(x|w_j) = \prod_{i=1}^d p(x_i|w_j)$
  - hierbei sind  $p(x_i|w_j)$  eindimensionale Verteilungen die sich dann auch sinnvoll mit Hilfe des Parzen-Window-Ansatzes approximieren lassen



## 7.3 k Nearest Neighbours

- Dichteschätzung:
  - Ansatz:  $p(x) = \frac{k}{n*V}$
  - Kern-Dichte-Schätzer: Volumenfest bestimmt durch h
  - k Nearest Neighbours: k wird festgelegt und Volumen wird so lange vergrößert bis k Samples gefunden wurden
  - man kann zeigen dass wenn man  $k_n$  langsamer wachsen lässt als n, die erzeugten Volumen  $V_n$  gegen 0 konvergieren:  $p_n(x) = \frac{k_n}{n*V_n} \approx p(x)$  mit  $\lim_{n \rightarrow \infty} p_n(x) = p(x)$
- $p_n(x) = \frac{k_n}{n*V_n} \approx p(x)$  in der Praxis
  - sei  $n = 1$ . Es gibt also genau 1 Sample  $x_1$  und wir haben  $k_n = \sqrt{n} = 1$
  - gegeben sei ein Testpunkt x
  - Das kleinste Volumen  $k_1$ , des kleinsten Hyperwürfels mit Zentrum x, in der wir gerade noch  $x_1$  finden?  $2|x - x_1|$
  - daraus ergibt sich eine sehr schlechte Schätzung:  $p_1(x) = \frac{1}{2|x - x_1|}$
- Verhalten für k-NN Dichteschätzung: für ein n kann die k-NN Dichteschätzung ziemlich stachelig sein
- Klassifikation
  - n Samples  $x_i$  aus c Klassen
  - suchen k nächsten Samples in der Umgebung eines Punktes x
  - $k_j$  Anzahl der Samples aus k welche zur Klasse  $w_j$  gehören ( $\sum_{j=1}^c k_j = k$ )
  - mit  $\hat{p}(x, w_j) = \frac{k_j}{n/V}$  erhalten wir:  $\hat{P}(w_j|x) = \frac{p(x, w_j)}{p(x)} = \frac{k_j}{k}$  also der Stimmanteil für  $w_j$  relativ zur Gesamtanzahl der Stimmen k
  - Klassifikation nach:  $\operatorname{argmax}_{w_j} \hat{P}(w_j|x)$
  - Um einen Punkt x zu klassifizieren: Bestimme die k nächstliegenden Trainingspunkte  $x_i$  und weise x diejenige Klasse zu, die unter diesen k Punkten am häufigsten auftritt

## 8 Support Vector Machines

### 8.1 Einführung

- Geg: Zwei Klassen in einem Merkmalsraum mit nichtlinearer Trennbarkeit
- Lösung: Transformation in einen höherdimensionalen Merkmalsraum
- Allg. die Transformation ist unbekannt welche eine lineare Trennbarkeit erreicht
- Schrotflinten-Verfahren: Wähle eine geeignete Menge von Basisfunktion und probiere rum :D
- Wie findet man einen geeigneten Satz von Transformation  $\phi : R^d \rightarrow R^f$
- Wie findet man eine für die transformierte Punktemenge eine geeignete lineare Entscheidungsgrenzen im  $R^f$ ?
- Wie sorgt man dafür dass die Klassifikationskomplexität...?

### 8.2 Entscheidungsgrenzen

- gegeben eine Transformation  $\phi : R^d \rightarrow R^f$  und Trainingsdaten und Klassenzugehörigkeit
- gesucht lineares Modell:  $y(x) = w^t \phi(x) + b$  so dass die Ebene  $y(x) = 0$  die beiden Klassen trennt
- Gesucht wird als für gegebene  $x_i$  eine möglichst gute Trenngrenze
- Wir suchen diejenige Ebene  $y(x) = 0$  die den Abstand zu den jeweils am nächsten liegenden Punkten beider Klassen maximiert
- Diejenige Punkte die der Ebene  $y(x) = 0$  am nächsten liegen stützen die Ebene ab  $\rightarrow$  Stützvektoren

- Mathematisch: Gesucht  $y(x) = w^t \phi(x) + b = 0$  die den minimalen Abstand der Punkte zu ihr maximiert also die Lösung für:  $\operatorname{argmax}_{w,b} (\frac{1}{\|w\|} \min_i (t_i (w^t \phi(x_i) + b)))$
- Problem: schwieriges Optimierungsproblem
- Lsg: Umwandlung in ein einfacheres
- Beobachtung: Wenn wir von gegebenen  $w, b$  zu skalierten Werten  $k^* w$  und  $k^* b$  übergehen ändert sich das Ergebnis nicht d.h.  $\frac{t_i (w^t \phi(x_i) + b)}{\|w\|} = \frac{t_i ((k^* w)^t \phi(x_i) + k^* b)}{\|k^* w\|}$
- für gegebenes  $w, b$  finden wir also immer ein  $k$  so dass für die skalierte Lösung  $\frac{t_i ((k^* w)^t \phi(x_i) + k^* b)}{\|k^* w\|} = \frac{t_i (w'^t \phi(x_i) + b')}{\|w'\|} = 1$
- Ansatz: suche einfach  $k$  so dass gilt:  $t_i (w'^t \phi(x_i) + b') = t_i ((k^* w)^t \phi(x_i) + k^* b) = 1$  gilt: wir benennen  $w'$  und  $b'$  um in  $w$  und  $b$
- für die der optimalen Entscheidungsgrenze am nächsten liegen gilt:  $t_i (w^t \phi(x_i) + b) = 1$
- Für alle Punkte fordern wir also:  $t_i (w^t \phi(x_i) + b) \geq 1$  (kanonische Repräsentation der Entscheidungsgrenze)
- Punkte mit Wert = 1 sind aktive Punkte
- Vereinfachtes Optimierungsproblem:  $\operatorname{argmax}_{w,b} (\frac{1}{\|w\|} \min_i (t_i (w^t \phi(x_i) + b))) = \operatorname{argmax}_{w,b} \frac{1}{\|w\|}$  und die kanonische Repräsentation erfüllt
- Statt  $\frac{1}{\|w\|}$  zu maximieren können wir auch  $\|w\|^2$  minimieren d.h. die suchen  $\operatorname{argmin}_{w,b} \frac{1}{2} \|w\|^2 + \text{Nebenbedingungen}$
- Problem: quadratische Programmierung d.h. Konvexe Problemstellung besitzen ein globales Minimum

### 8.3 Lagrange-Multiplikatoren

- gegeben Funktion  $f(x)$  die wir optimieren wollen und eine Bedingung  $g(x) = 0$  die erfüllt werden muss
- $\nabla_x f(x) + \lambda \nabla_x g(x) = 0$ . Man bezeichnet  $\lambda$  als Lagrange-Multiplikator
- erweiterte Funktion:  $L(x, \lambda) = f(x) + \lambda g(x)$
- Optimum muss gelten:  $\nabla_x L(x, \lambda) = 0$  sowie  $\nabla_\lambda L(x, \lambda) = g(x) = 0$
- Wie mit Ungleichungen?
  1. Optimum liegt innerhalb des Constraintbereichs also  $g(x_B) > 0$  und Gradient von  $g$  spielt keine Rolle also  $\lambda = 0$
  2. Optimum liegt auf dem Constraint also  $g(x_A) = 0$  und Gradient ist antiparallel zum Gradient von  $f$  d.h.  $\nabla f(x) = -\lambda \nabla g(x)$  für  $\lambda > 0$

### 8.4 Bestimmung der Entscheidungsgrenze

- Die Formulierung mit Lagrange-Multiplikatoren  $a_i \geq 0$  für die Nebenbedingung ergibt sich zu:  $L(w, b, a) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^n a_i (t_i (w^t \phi(x_i) + b) - 1)$
- Minuszeichen weil wir in Bezug auf  $w$  und  $b$  minimieren
- Ableitungen:
  - $w = \sum_{i=1}^n a_i t_i \phi(x_i)$  (verbraucht den 1/2 Faktor)
  - $0 = \sum a_i * t_i$
- Ableitung einsetzen um  $w$  und  $b$  zu eliminieren
- neuen Problem:  $\hat{L}(a) = \sum a_i - \frac{1}{2} \sum_i \sum_j a_i a_j t_i t_j k(x_i, x_j)$  mit  $a_i \geq 0$  und  $\sum_i a_i t_i = 0$
- Die Kerne  $k(x_i, x_j)$  sind hierbei definiert durch  $k(x, x') = \phi(x)^t \phi(x')$
- Wenn wir das duale Optimierungsproblem gelöst haben erhalten wir für die  $a_i$  über die sich dann auch  $b$  bestimmen lässt
- Klassifikation über:  $y(x) = w^t \phi(x) + b$

- Wenn wir das gefundene  $w$  einsetzen mit  $w = \sum_{i=1}^n a_i t_i \phi(x_i)$  ergibt sich:  $y(x) = \sum_{i=1}^n a_i t_i \phi(x_i)^t \phi(x_i) + b = \sum_{i=1}^n a_i t_i k(x_i, x) + b$
- Optimierungsproblem ist definiert durch:
  - Funktion:  $\bar{L}(a) = \sum_{i=1}^n a_i - 1/2 \sum_{i=1}^n \sum_{j=1}^n a_i a_j t_i t_j k(x_i, x_j)$
  - Krush Kuhn-Tucker-Bedingungen:
    - $a_i \geq 0$
    - $t_i y(x_i) - 1 \geq 0$
    - $a_i \{t_i y(x_i) - 1\} = 0$
- für den Punkt  $x_i$  gilt also entweder  $a_i = 0$  oder  $t_i y(x_i) = 1$  d.h. entweder der Punkt spielt in der Klassifikation keine Rolle ODER er hat den minimalen Abstand zur Entscheidungsebene (Stützvektor)
- wenn wir einen Wert  $a$  haben dann können wir  $b$  ermitteln
- für einen Stützvektor  $x_j$  gilt:
  - $t_j y(x_j) = 1$
  - also  $t_j (\sum_{i \in S} a_i t_i k(x_i, x_j) + b) = 1$
- $S$  die Menge der Indizes der Stützvektoren ( $a_i$  der anderen Vektoren sind ja ohnehin 0)
- numerisch stabilere Lösung: Durchschnitt der  $b$  Werte über alle Stützvektoren

## 8.5 Kerne und Dimensionen

- Was bedeutet  $\hat{L}(a) = \sum_{i=1}^n a_i - 1/2 \sum_{i=1}^n \sum_{j=1}^n a_i a_j t_i t_j k(x_i, x_j)$ ?
- Ein Kern misst die Ähnlichkeit zweier Vektoren
- um  $\hat{L}$  zu maximieren, müssen wir ähnliche  $x_i, x_j$  finden die unterschiedliche Klassen angehören  $t_i t_j = -1$  und diese mit großen  $a_i, a_j$  wichten
- Diese erhöhen den Wert von  $\hat{L}$  da die Summe mit negativen Vorzeichen in die Maximierung eingeht
- $w = \sum_{i=1}^n a_i t_i \phi(x_i)$  bedeutet dann
- die Summe kann in zwei Teile zerlegt werden: die Summe der Klassen  $t_i = 1$  und die Summe der Klasse  $t_i = -1$ :  $w = \sum_{i \in t_i=1} a_i \phi(x_i) - \sum_{j \in t_i=-1} a_j \phi(x_j)$

## Literatur

- [1] Richard O Duda, Peter E Hart, and David G Stork. *Pattern classification*. John Wiley & Sons, 2012.