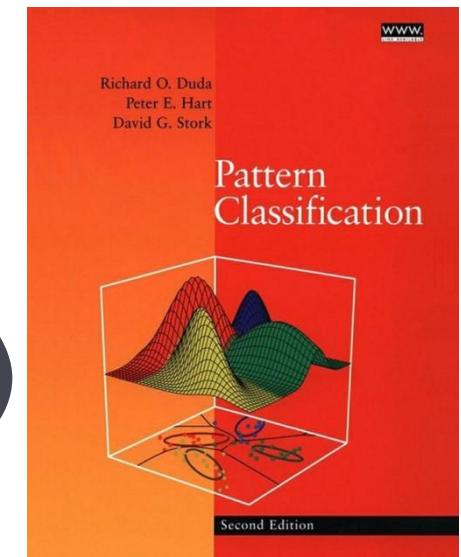


Sensordatenverarbeitung

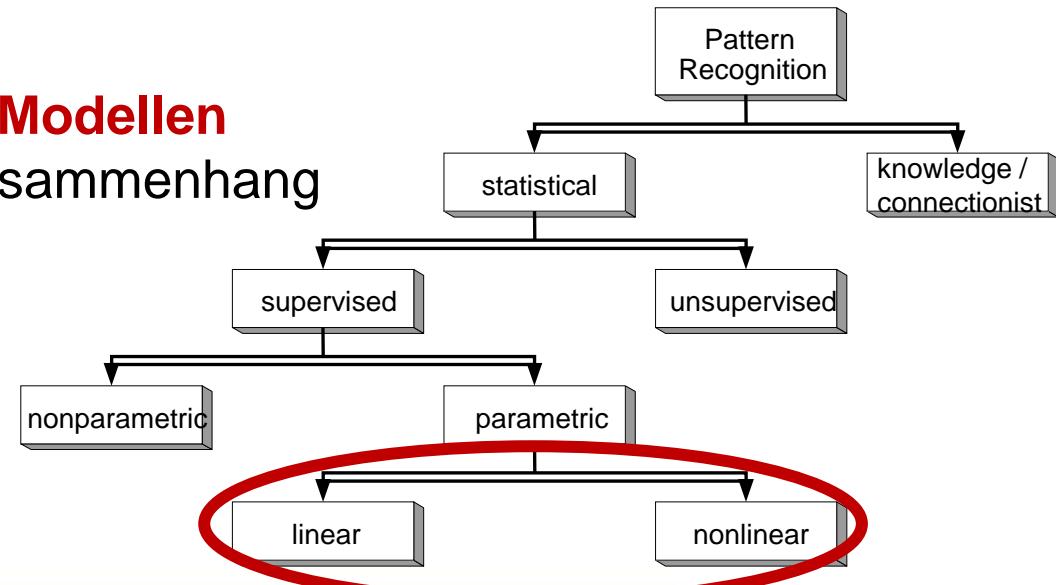
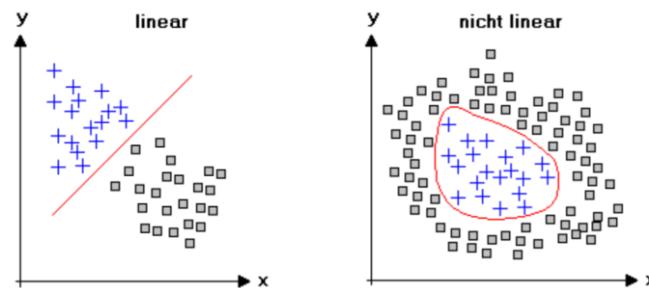
KLASSIFIZIERUNG (11C)

06.01.2024



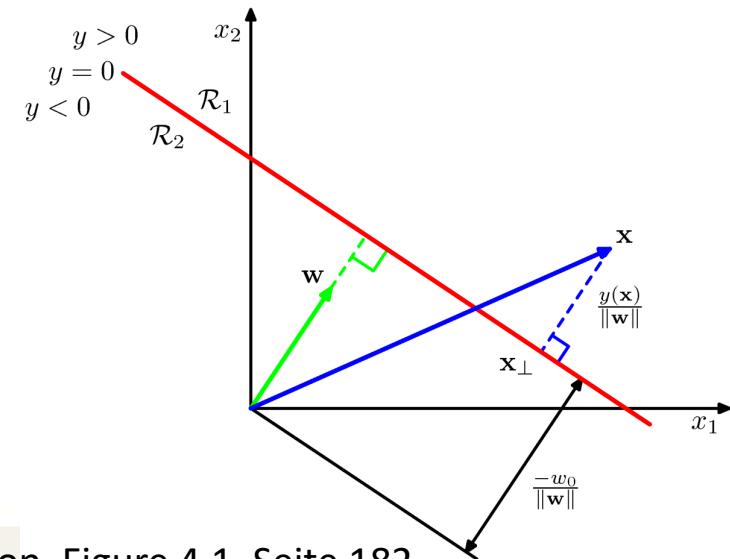
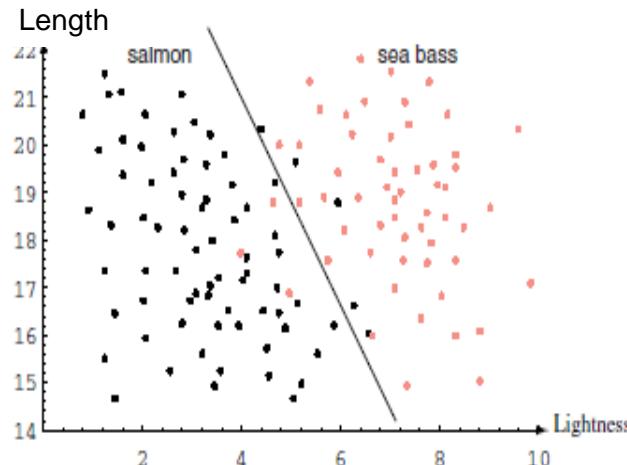
Teil C

- **Lineares statistisches Modell** basieren auf der Grundannahme, dass ein linearer Zusammenhang (eine „geradlinige Beschaffenheit“) zwischen den beobachteten Daten und bekannten Einflussvariablen (Merkmale) besteht
- Die Parameter des Modells sind linear $y_k(x) = w_k^T x + b_k$
- Beispiel: Lachse und Barsche lassen sich im 2D-Raum (Größe, Helligkeit) durch eine **Entscheidungsgerade** fehlerfrei trennen ...
- ... Im 3D-Raum durch eine Ebene, im >3D-Raum durch eine Hyperebene



Nochmal Einführungsbeispiel

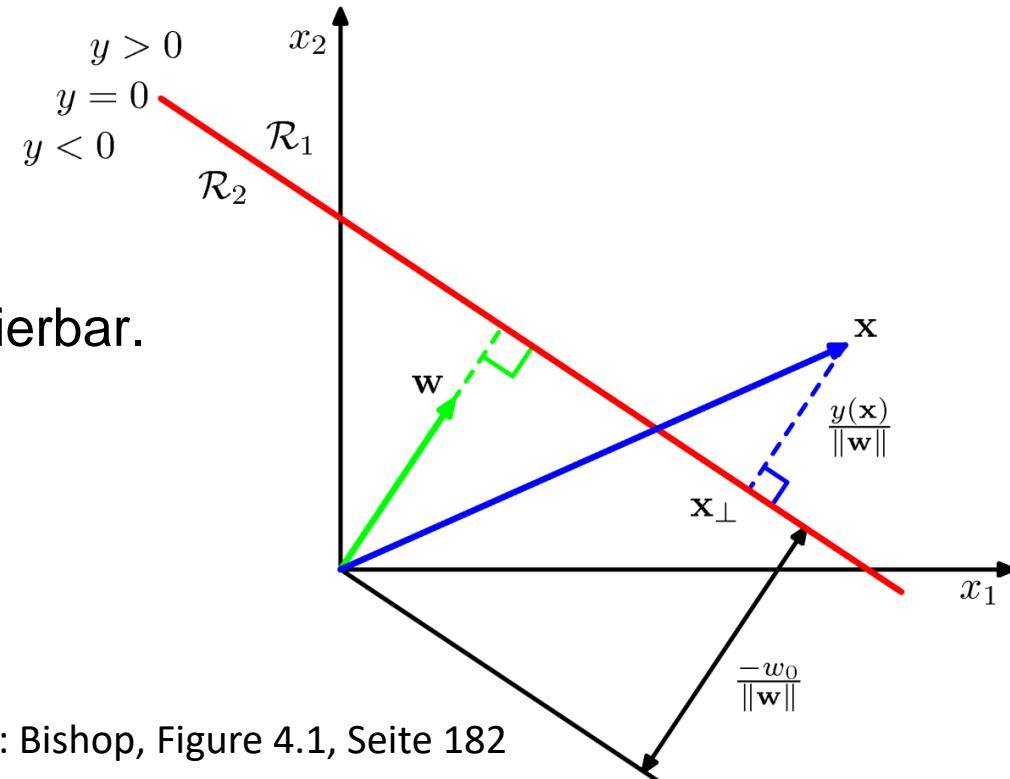
- Lachse und Barsche im zweidimensionalen Merkmalsraum
 - x_1, \dots, x_N ist die Menge der Trainingssamples
(Lachse und Barsche in 2-dimensionaler Darstellung = Vektor Helligkeit, Länge)
 - *Zielwert*: Für K Klassen definieren wir für jedes Eingabesample einen Vektor mit K Komponenten, der angibt, zu welcher Klasse ein Sample gehört. Bei 2 Klassen und mit x_i ein Sample aus Klasse 2 definieren wir $t_i = (0, 1)^T$.
- Ein linearer Klassifikator ist gegeben durch die lineare Funktion
 - $y(x) = w^T x + b$. Mit Gewichtsvektor w und bias b
 - Sample x (*Fisch*) wird der Klasse C_1 zugeordnet, falls $y(x) \geq 0$, sonst Klasse C_2
 - $\mathcal{R}_p, \mathcal{R}_2$: "Entscheidungsregionen"



Quelle: Bishop, Figure 4.1, Seite 182

- Der Merkmalsraum wird in zwei Klassen geteilt, die durch eine Hyperebene mit der Gleichung $w^T x = 0$ separiert sind.
- Der skalare Parameter b bestimmt die Position der Hyperebene.
- Wenn es eine Hyperebene gibt, die alle Eingabevektoren fehlerlos trennt, heißen die Klassen *linear separierbar*.

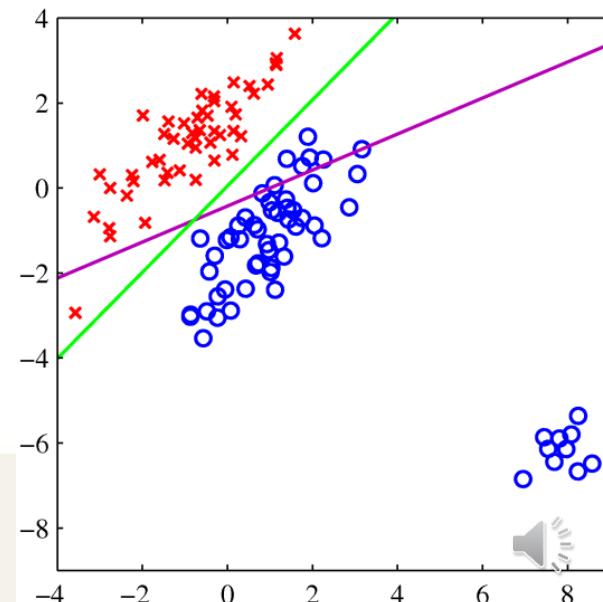
In der Praxis: oft nicht linear separierbar.



Quelle: Bishop, Figure 4.1, Seite 182

- Wie findet man $y_k(x) = w_k^T x + b_k$
- Zwei Verfahren, um die Parameter zu finden:
 - Minimierung des quadratischen Fehlers (Least-Squares LS)
 - Der Perzeptron-Algorithmus
- Minimierung des quadratischen Fehlers (LS):
 - Idee: Fehlerfunktion für die Klassifikation definieren, die sich analytisch minimieren lässt. Der quadratische Fehler ist immer positiv, daher findet man die lineare Funktion dadurch, dass man ableitet und Nullstelle sucht
 - Also: Abstand der Datenpunkte zu Funktionswert ermitteln, quadrieren, und die Parameter so wählen, dass die Summe der quadrierten Abstände möglichst gering wird
 - Problem: korrekt klassifizierte Daten können zur Schätzung einer Lösung führen, die falsche Klassifikationen ergibt (siehe Abbildungen)
 - Grüne Trennungslinie wäre gut, LS ergibt aber violette Trennungslinie wegen der (korrekt klassifizierten) Samples unten rechts!

SVM, Perzeptron machen diesen Fehler nicht



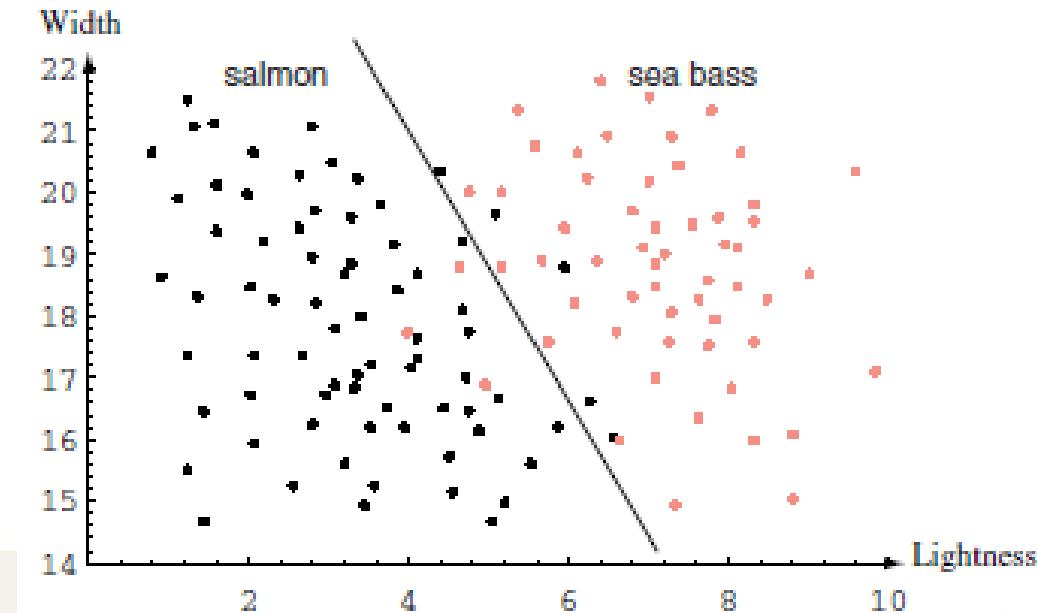
Quelle: Bishop

- Beispiel
 - Gegeben: Fisch mit Länge (Width) von **250cm** und Helligkeit (Lightness)
 - Gesucht: Klassenzugehörigkeit (hier Lachs oder Seebarsch)
 - Entscheidung auf Basis der Linearen Klassifikation (unten): Seebarsch
 - Problem: Ein Seebarsch von 250cm??? Ist vielleicht eher ein Hai?
- Lösung: Statt ja/nein ist es oftmals besser, anzugeben **wie sicher** ein Sample zu einer Klasse gehört \Rightarrow Wahrscheinlichkeit

- $P(A=\text{Seebarsch}|B=250\text{cm})$

gibt die Wahrscheinlichkeit an,
dass es sich um die Klasse
„Seebarsch“ handelt, gegeben
dass das Merkmal „Width“ den
Wert 250cm hat

- $P(A|B)$ sprich
„P von A gegeben B“



- $P(A=\text{Seebarsch}|B=250\text{cm})$ Wahrscheinlichkeit für eine Klasse A (Seebarsch) gegeben einen Merkmalsvektor B (z.B. Länge=250cm)
- $P(A=\text{Seebarsch}|B=250\text{cm})$ kann nicht direkt bestimmt werden
- $P(A=\text{Seebarsch}|B=250\text{cm})$ ist die A-Posteriori Wahrscheinlichkeit (*a-posteriori = man muss erst B=250cm beobachtet haben ...*)
- Daher modellieren man über Bayes Theorem:

$$P(A | B) = \frac{P(B | A) \cdot P(A)}{P(B)}$$

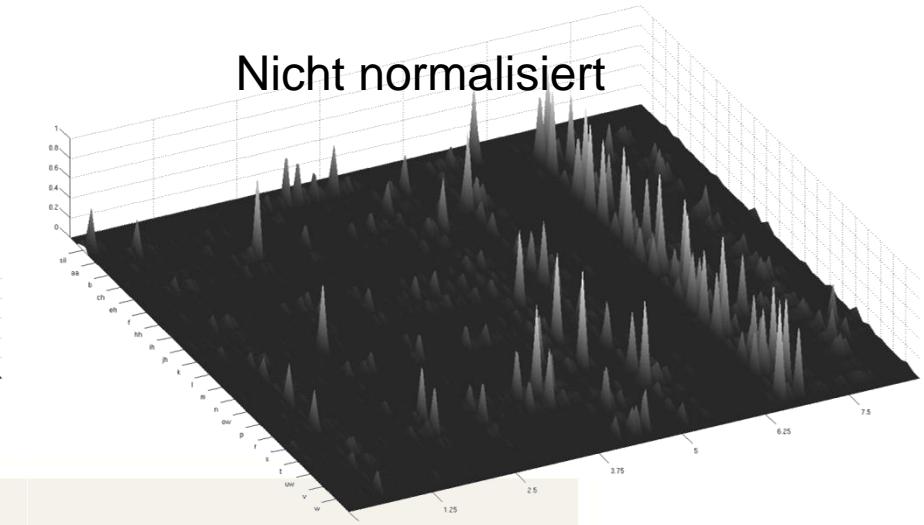
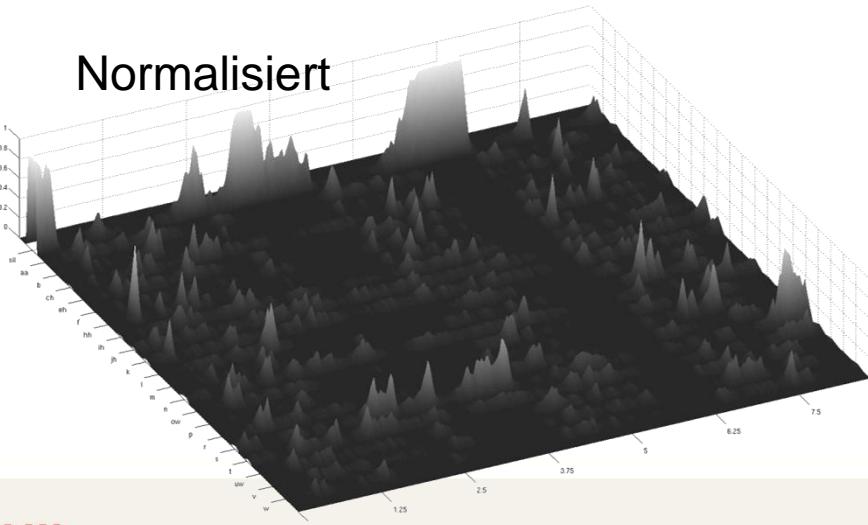
- $P(B=250\text{cm}|A=\text{Seebarsch})$ = Wahrscheinlichkeit für die Länge 250cm (Merkmal B) gegeben dass es sich um einen Seebarsch (Klasse A) handelt
- $P(A=\text{Seebarsch})$ = A-priori Wahrscheinlichkeit für Klasse Seebarsch
- $P(B=250\text{cm})$ = A-priori Wahrscheinlichkeit für Fischlänge 250cm (*a-priori = weiß man schon VOR der Beobachtung*)

- Wenn nur die höchste Wahrscheinlichkeit (beste Klasse) gesucht wird, kann die Normalisierung $P(B=b)$ weggelassen werden

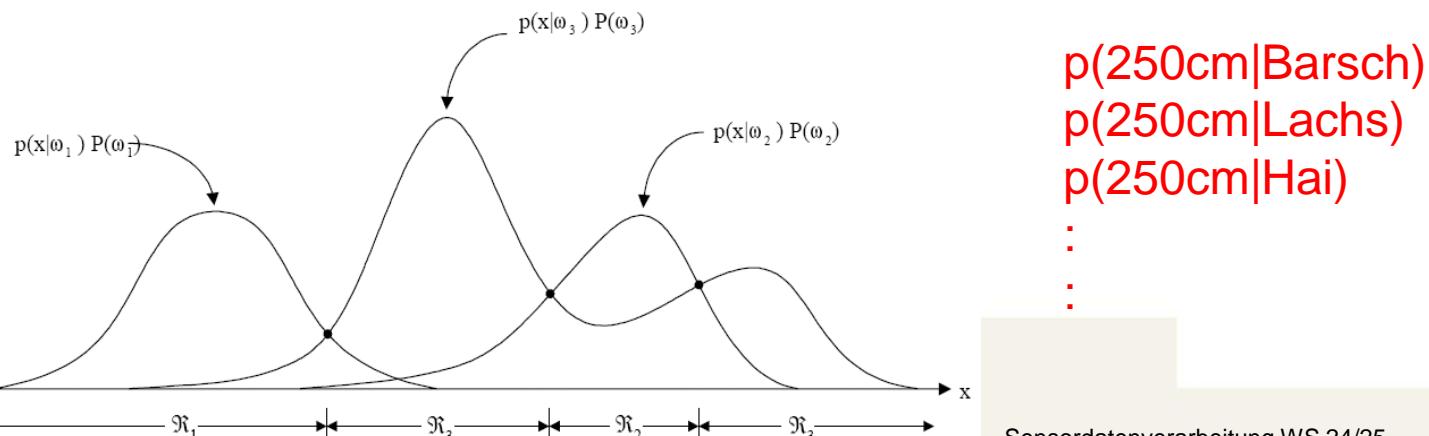
- es wird für denselben Fisch aka gleiches Merkmal B die beste Klasse A gesucht)

$$\operatorname{argmax}_a P(A = a|B = b) = \operatorname{argmax}_a \frac{P(B = b|A = a) \cdot P(A = a)}{P(B \cancel{\neq} b)}$$

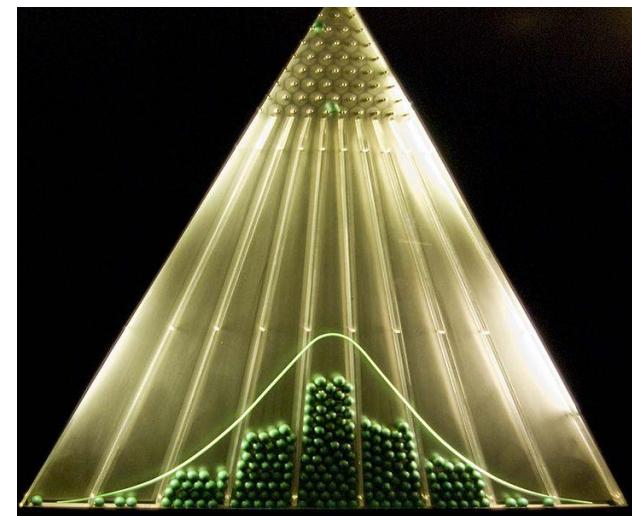
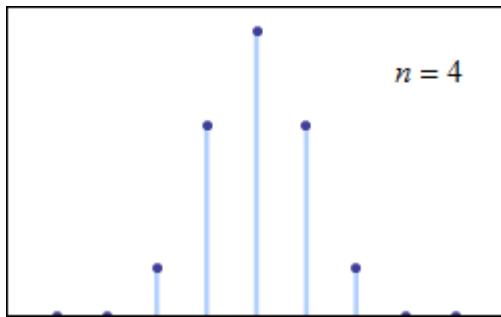
- Ohne Normalisierung sind die Wahrscheinlichkeiten stark korreliert
- Also keine echten Wahrscheinlichkeiten P mehr
(in einigen praktischen Anwendungen braucht man keine W-keiten)



- Für die Auswertung gemäß Bayes brauchen wir $p(x|C_i)$ für jede Klasse C_i
- Gaußklassifikator modelliert die Verteilung von Datenpunkten mit Hilfe der (mehrdimensionalen) **Gaussverteilung** oder **Normalverteilung**
- Für jede Klasse von Samples werden die Parameter derjenigen Gaussverteilung bestimmt, die die Lage der Trainingssamples im Merkmalsraum (Featureraum) möglichst gut beschreibt.
 - Es werden also für JEDE Klasse getrennte Verteilungen modelliert
- Die eigentliche **Klassifikation** basiert dann auf der Auswertung der **Dichtefunktion der Gaussverteilung**.
 - *Es ergeben sich Entscheidungsregionen (müssen nicht zusammenhängen)*
 - Siehe unten: Ein Sample x , das in Region R_i fällt, wird als Klasse i klassifiziert.



- Aus Stochastik: **zentraler Grenzwertsatz** - die Summe vieler unabhängiger Zufallsvariablen konvergiert gegen eine Gaussverteilung.
- Viele zufällige Prozesse, besonders mit gewissen Fehlern oder Toleranz kann man gut mit einer Gaussverteilung modellieren.
- Beliebige Verteilungen lassen sich gut als Summe von Gaussverteilungen darstellen (Gauss'sche Mischverteilungen – GMM)
- Mathematisch sind Gaussverteilungen einfach
- Natürliche Prozesse oft sehr nahe an der Gaussverteilung

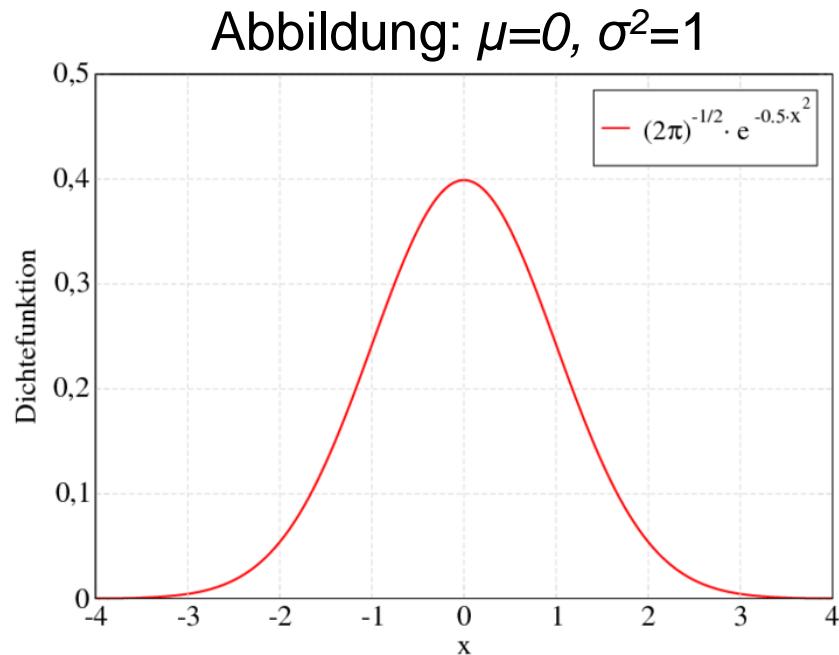


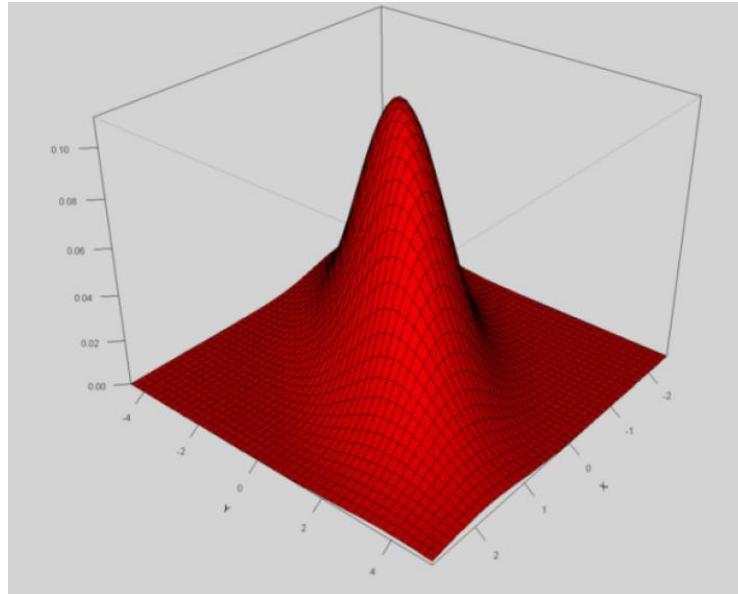
Für eine skalare Variable x hat die Gaussverteilung die folgende Dichtefunktion:

$$N(x | \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{1}{2\sigma^2}(x - \mu)^2\right]$$

Dabei ist μ der Mittelwert und σ^2 die Varianz der Verteilung.

Die Dichtefunktion ist ein kontinuierliches Maß für die „Wahrscheinlichkeitsmasse“, die auf einen bestimmten Punkt der x -Achse entfällt.





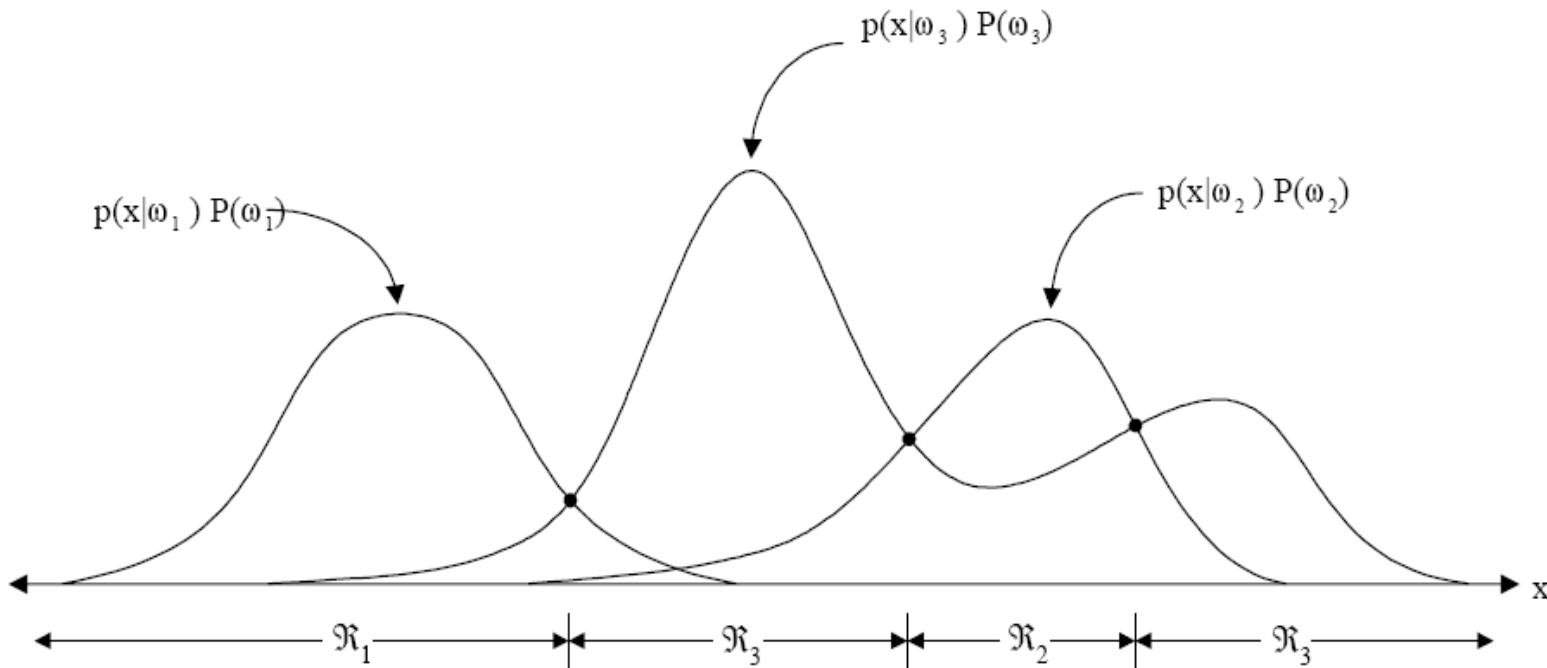
2-dimensional Gaussverteilung

- Wahrscheinlichkeitsdichtefunktion einer D -dimensionalen Gaussverteilung

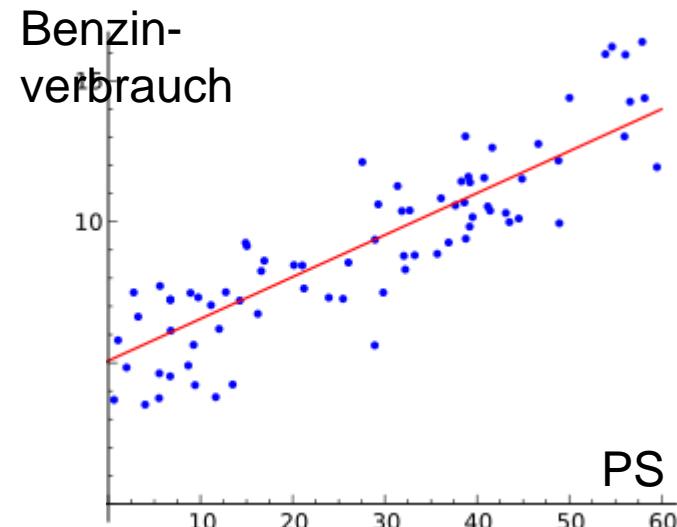
$$N(x | \mu, \Sigma) = \frac{1}{(2\pi)^{\frac{D}{2}} |\Sigma|^{\frac{1}{2}}} e^{-\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu)}$$

- Um einen Klassifikator basierend auf Gaussverteilungen zu definieren, verwenden wir für jede zu unterscheidende Klasse eine Gaußdichtefunktion.
 - Seien also $N(\cdot | \mu_i, \Sigma_i)$ ($i=1, \dots, N$) die Dichtefunktionen der Verteilungen der Klassen C_i ($i=1, \dots, N$).
- Die Verteilungen werden also für jede Klasse getrennt modelliert.
- Die **Klassifikation** basiert auf der Auswertung der Dichtefunktion
 - Für Eingabesample x wird für jede Klasse der Dichtewert berechnet, d.h.
 $y_i = N(x | \mu_i, \Sigma_i)$
 - x wird derjenigen Klasse zugeordnet, **für die y_i den höchsten Wert annimmt**
 - Beachte: Die Werte y_i stellen *keine* Wahrscheinlichkeiten dar (sie können sogar viel größer als 1 sein).
- Hinweis: Im Prinzip geht diese Methode auch für andere Verteilungen als Gaussverteilungen.
- Dazu müssen die Gaussverteilungen für jede Klasse trainiert werden
 - Einfachste und gängigste Methode: **Maximum-Likelihood (ML)-Abschätzung**
 - Verfahren: Expectation-Maximization-Algorithmus (EM-Algorithmus)

- Beispiel für Entscheidungsregionen bei einem Gauss-Klassifikator
- Es gibt drei Entscheidungsregionen, die nicht zusammenhängend sein müssen.



- Bei **Klassifikation** geht es um das Bestimmen der vorliegenden Klasse
 - Beispiele: Klassen = Lachs oder Seebarsch
- Wenn sich die abhängige Variable nicht in verschiedene Klassen einteilen lässt, spricht man von **Regression** (später)
 - Beispiel: Vorhersage von Benzinverbrauch in Abhängigkeit der PS
- Auch bei der Regression verwendet man Trainingsdaten um den Zusammenhang zu lernen:
 - Vorhersage für unbekannte Daten
 - Aber nicht für bestimmten **Klassen**



- Einführung Klassifizierung
- Taxonomie der Ansätze in Mustererkennung
- Clustering
- Klassifikatoren
 - nichtparametrische Verfahren
 - lineare Klassifikatoren
 - Bayes Entscheidungstheorie
 - Gaussklassifikatoren
- Abgrenzung zur Regression

Bremen Big Data Challenge

- ▲ Zeitraum: 01. bis 31. März 2023
- ▲ Datenanalyse-Wettbewerb
- ▲ Nur Programmierkenntnisse nötig
- ▲ Für Studierende in Bremen und umzu
- ▲ Teams mit bis zu 3 Personen
- ▲ Geldpreise in Höhe von 1.500€
- ▲ Sachpreise für Mitglieder der ersten 30 Teams

<http://bbdc.csl.uni-bremen.de>



! Anrechnung als Informatik-Seminar möglich !

(Lösung im Seminar präsentieren, 6-Seiten schreiben, 3 ECTS)

