Mustererkennung mit stochastischem Lernalgorithmus

Diplomarbeit

vorgelegt von
Rüdiger Brause

Hiermit versichere ich,
keine außer den angegebenen
Hilfsmittelm benutzt zu haben.

Tübingen, den 1.3.1978

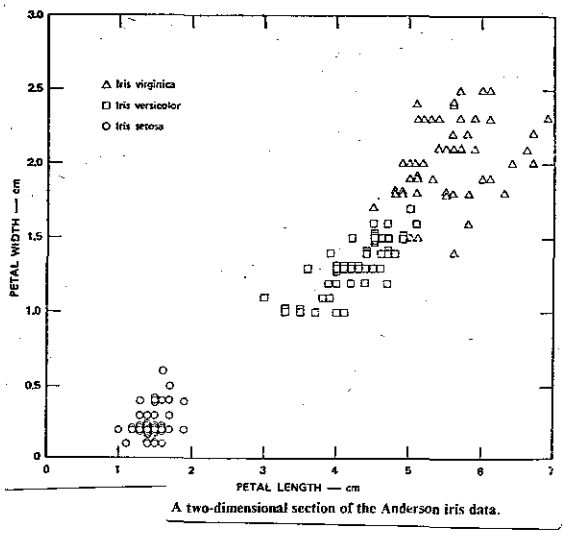
Rüdiger Brause

INHALTSVERZEICHNIS

1. EINFÜHRUNG

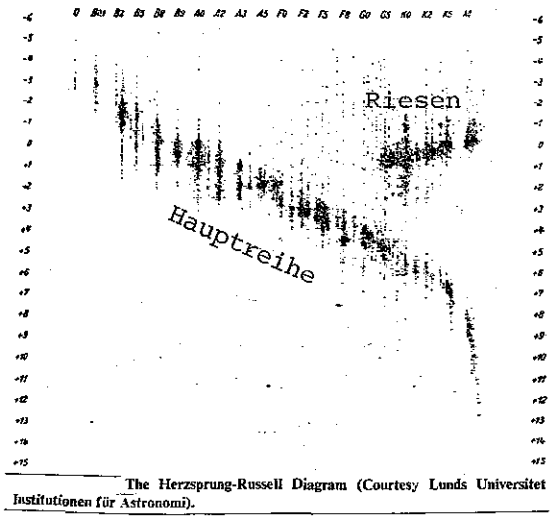
Informationsverarbeitende dynamische Systeme gleich welcher Art empfangen von ihrer Umwelt Signale oder Zeichen (Muster), verarbeiten diese und antworten auf sie mit zweckentsprechenden Reaktionen bzw. passen sich an die sich ändernden Umweltsituationen an. Letzteres entweder infolge gezielter, optimaler technischer Konstruktion oder biologischer Evolution (survival of the fittest). Dazu ist es notwendig, die außerordentliche Vielfalt der Input-Muster des Systems, die zur selben Reaktion Anlaß geben, in Situationsklassen zusammenzufassen. Denn nur durch eine solche Klassifikation (d.h. Informationsreduktion) ist das System in der Lage, schnell zu entscheiden und dabei adäquat zu reagieren.

Beispiele für Klassenbildungen zeigen Abbildung 1a und 1b.



A two-dimensional section of the Anderson iris data.

Abb.1a



The Herzsprung-Russell Diagram (Courtesy Lunds Universitet Institutionen für Astronomi).

Abb.1b

Im Herzsprung-Russel Diagramm sind Sterndaten eingetragen, geordnet nach ihrer Spektralklasse(horizontal) und ihrer absoluten Intensität(vertikal). Die sich ergebenden Klassen unterteilen die Datenmuster in die Gruppen der Sternriesen, der Superriesen,der Hauptreihe und der Zwerge.Zu sehen sind in dieser Abbildung die Sterngruppe der Hauptreihe und der Riesen.

Drei verschiedene Irisarten,die mit ihren Abmessungsmerkmalen aufgetragen wurden, zeigt Abb.1a .Die teilweise "unscharfe" und "verschwommene" Abgrenzung der Klassen macht die Probleme der Klasseneinteilung beim Mustererkennen deutlich.

Eine sich ändernde Umwelt erzeugt jedoch auch neue Muster oder Zeichen, die für das System unbekannt sind. Lassen sich diese nicht in die bereits vorliegenden Klassen einordnen, so muß die ursprüngliche Klasseneinteilung der Input-Muster durch eine neue Klassifikation ersetzt werden. Das System muß diese Veränderung dabei möglichst selbständig vornehmen, wenn es adaptiv, also selbstlernfähig und reagierend sein soll. Beispiele liegen auf der Hand: Jede Markterhebung ändert die Selbsteinschätzung der Wirtschaftssituation eines Unternehmens (des Systems), kann zu einer neuen Klassifikation der bisherigen Marktdaten führen und bewirkt als Lernreaktion eine Neuorientierung der Produktion. Ähnlich ändern sich medizinische Diagnostik oder Wetterprognose durch die Verarbeitung neuer Daten.

Wenn das System ein Ähnlichkeitskriterium der Muster gefunden hat, so ergibt sich das Problem, den Fehler und damit auch die Folgen bei der Klassifizierung zu minimisieren. Die Optimierung der Klasseneinteilung wird als *Lernen* bezeichnet. Erst wenn dies ausreichend durchgeführt wurde, ist es für das System möglich, erfolgreich Muster zu erkennen. Für technisch konstruierte Mustererkennungssysteme läßt sich diese "Lernphase" durch Eingabe bekannter Trainingsmuster bewältigen. Dabei wird jede Klassifizierung mit "richtig" oder "falsch" bewertet und bewirkt so eine Korrektur der Klasseneinteilung. In der nachfolgenden"Erkennungsphase""braucht so die Klasseneinteilung nicht mehr verändert werden.

Bei diesem "Lernen" wird aber die Information über die optimale Klasseneinteilung schon vorausgesetzt, was aber bei biologischen und hochflexiblen technischen Systemen normalerweise gerade nicht der Fall ist. Diese Information muß erst während eines *Selbstlernens* vom System aus den Mustern erschlossen werden.

Das zentrale Problem dabei besteht darin, *wie* die Information zur Optimierung der Klasseneinteilung aus den Mustern gewonnen wird.

Die vorliegende Arbeit versucht, dieses Problem mit dem Formalismus des "stochastischen Lernens" zu untersuchen und theoretisch sowie mittels Computersimulation Möglichkeiten und Grenzen der Lösungen anzugeben.
Zunächst wird daher das Problem der Mustererkennung als Problem der Einordnung einer stochastischen Variablen x in vorhandene Klassifikationen behandelt (Kapitel 1.2).
Um die Zahl der Klassen und die Wahrscheinlichkeit, ihnen ein x zuzuordnen, zu bestimmen, wird dann die Wahrscheinlichkeitsdichtep(x) aus den vorliegenden x rekonstruiert (Kap. 2).
Dabei werden einfache Fehler (Kap. 3) oder mit einer Strafe gewichtete Fehler mit Hilfe sogenannter *Straffunktionen* (Kap. 5) so klein wie möglich gemacht.
Anschließend werden konkret eine definitive Straffunktion sowie ein Lernalgorithmus ausgewählt und deren Charakteristika für die weitere Anwendung diskutiert. (Kap. 6).

In Kap. (7) wird untersucht, ob bei diesem Lernalgorithmus bei Änderung eines Parameters statt *einer* optimalen Klasseneinteilung (Fixpunkt des Algorithmus) plötzlich mehrere optimale Einteilungen *(Bifurkation)* auftreten können.
Als Resultat wird eine Bedingungsgleichung für p(x) formuliert, bei deren Erfüllung eine Bifurkation auftreten muß.
Für eine Art von Verteilungen wird gezeigt, daß diese Gleichung nicht erfüllt ist. Eine weitere Verteilungsart wird angegeben, bei der die genannte Bifurkation auftritt. Mittels Computersimulation wird hierbei der Übergang von "normalem Lernen" zu "bifurkativem Lernen" bei Parametervariation demonstriert. (Kap.7.3)
Die Konsequenzen für Mustererkennung werden in Kap. 8 besprochen. Im Anschluß daran wird die Anwendung der stochastischen Mustererkennung an biophysikalischen Beispielen illustriert. Unter anderem wird gezeigt, daß man die haploide und diploide Vermehrung von Lebewesen mit Lernalgorithmen beschreiben kann, deren Lernziel die maximale Fitness der Gesamtpopulation darstellt.

1.2 Die Problemstellung

Die Notwendigkeit objektiver Klassifizierung häufig wiederkehrender Muster ,sei es in der Arztpraxis oder beim Sortieren von Briefen anhand der Postleitzahlen,zeigt,wie wichtig es ist,eine mathematische Beschreibung der Mustererkennung einzuführen.

Dabei möchte ich nur Muster betrachten,die quantitativ (Daten in Form von Zahlen)vorliegen.Qualitative Daten,wie sie z.B. die chinesischen Schriftzeichen 天 下 為 公 bilden,werden am Besten nicht durch stochastische Mustererkennung sondern eher durch Entscheidungsbäume(Abb.1.2) erkannt,ähnlich wie ein Satz einer Sprache mit der Grammatik zerlegt und dabei eingeordnet werden kann.

Dazu siehe [9] Kap.8 und [4] Kap.5

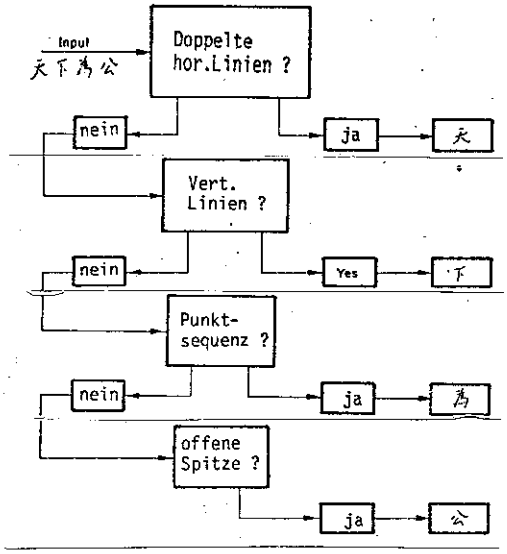
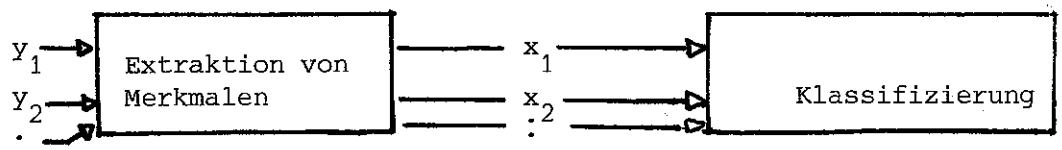


Abb. 1.2

Angenommen,es sei gelungen,alle für eine Klassifizierung relevanten Daten herauszufinden *(Feature selection)*;wie,wollen wir hier nicht behandeln.



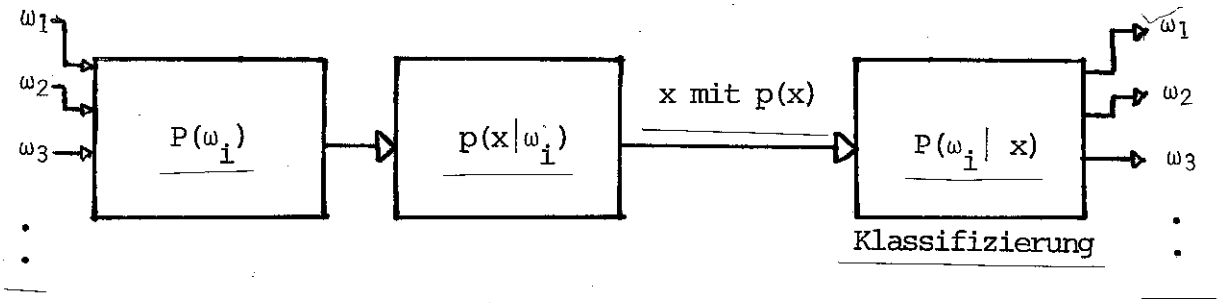
sehr viele Daten

wenige,relevante, typische Daten *(Features)*

Das so erhaltene n-Tupel von n charakteristischen Daten $(x_1,\ldots,x_n)$ ist eine vom Zufall abhängige Variable $\underline{x}$. Sei Ω die Menge aller $\underline{x}$.

Definition: Eine KLASSE Ω_i ist eine Teilmenge von Ω

Für die Erzeugung und Erkennung der Muster möchte ich folgende Bezeichnungen einführen:



Für die Klassifizierung ω_i (wenn $x \in \Omega_i$) ist

$P(\omega_i)$ *a priori*-Wahrscheinlichkeit dafür, daß ω_i vorliegt

$p(x/\omega_i)$ *likelihood*-Wahrscheinlichkeitsdichte von x, wenn ω_i vorliegt

$P(\omega_i/x)$ *a posteriori*-Wahrscheinlichkeit für ω_i, wenn x vorliegt.

Zwei Fälle sind zu unterscheiden:

a) QUELLE BEKANNT

Wenn $p(x/\omega_i)$ und $P(\omega_i)$ bekannt sind, läßt sich mit

$$p(x) = \sum_i p(x/\omega_i) \cdot P(\omega_i) \qquad \text{Formel (1.2a)}$$

$$\sum_i P(\omega_i) = 1 \qquad \text{Formel (1.2b)}$$

und der Bayesregel $p(y/x) \cdot p(x) = p(x/y) \cdot p(y)$

$P(\omega_i/x)$ bestimmen:

$$P(\omega_i/x) = \frac{p(x/\omega_i)}{p(x)} \cdot P(\omega_i) \qquad \text{Formel (1.2c)}$$

Wählt man bei der Klassifikation der x die Klasse Ω_i, für die $P(\omega_i/x)$ am größten ist, so läßt sich zeigen, daß dann der mittlere Fehler am kleinsten wird. (*Bayes-Entscheidungskriterium*)
In der Praxis wird dieser Gedanke benutzt, um Maschinen für Klassifikationen einzurichten. Bekannte Trainingsmuster werden eingegeben und nach jeder Klassifikation wird signalisiert, ob von der Maschine richtig oder falsch eingeordnet wurde. (Perzeptron-Alg., siehe)
Durch die 'richtig'-'falsch' Meldungen der Menschen lernt der Automat $P(\omega_i)$, durch die Trainingsmuster $p(x/\omega_i)$.

Die *Klassengrenze* ist dabei operational durch die Klassifikationsvorschrift definiert.
Falls $\bigcap_i \Omega_i = 0$ und $\bigcup_i \Omega_i = \Omega$, so läßt sich die Klassengrenze zwischen zwei Klassen auch als Hyperfläche definieren, die beide Teilmengen Ω_i und Ω_j voneinander trennt.

b) QUELLE UNBEKANNT

Wenn über die Quelle nichts bekannt ist, muß das lernende System $P(\omega_i/x)$ direkt lernen.
Da in diesem Fall kein Lehrer dabei ist wie bei a), möchte ich den Vorgang als *Selbstlernen* bezeichnen im Unterschied zu *Lernen* in a).
Das Klassifizieren und iterative Verschieben der Klassengrenzen geschieht hier durch Bilden eines internen Gütekriteriums. Ziel der Klassifizierung und Einordnung der x ist es, diese Güte so groß wie möglich, oder, invers gesehen, eine Strafe bzw. Kosten so klein wie möglich zu machen.
In der Auswahl und im Einsatz von Straffunktionen steckt natürlich auch hier wieder die Funktion eines Lehrers; nur muß er hier nicht bei jedem einzelnen x entscheiden, dies tut der "Schüler" selbst[+].
Im Folgenden soll nur noch dieser Fall behandelt werden.

[+] Psychologen dürfte diese Idee als 'Über-ich' bekannt sein.

1.3 Straffunktionen und Klassengrenzen

Wie hängen nun die Straffunktionen mit den Klassengrenzen zusammen? Dazu untersuchen wir ein System mit zwei Klassen. Sei x das auftretende Muster, $x \in \Omega = \Omega_1 \cup \Omega_2$

Dann lassen sich die Klassen durch eine Hyperfläche mit der Gleichung $h(x)=0$ trennen.

$$\text{Klasse } 1 = \{x / h(x) > 0\}$$
$$\text{Klasse } 2 = \{x / h(x) \leq 0\}$$

Diese Hyperfläche und andere p(x) charakterisierende Größen (z.B.Maximum oder Mittelwert) fassen wir zum Parametervektor $d=(h,c)$ zusammen, den das System lernen soll.

Angenommen, wir verbinden jedes x mit einer Strafe, deren Größe von der Klasse abhängt, in die x eingeordnet wird, dann gibt es für dieses System zwei verschiedene Strafen, die Funktionen von x sind.

$$\left.\begin{array}{ll} & F_1(x,c) \text{ ,wenn } h(x)>0 \\ \text{und} & F_2(x,c) \text{ ,wenn } h(x)\leq 0 \end{array}\right\} := Q(x,c,h)$$

Ein Zusammenhang zwischen $h(x)$, $F_1(x,c)$ und $F_2(x,c)$ ergibt sich aus der Forderung, daß

$$Q(x,c,h) \leq Q(x,c,\tilde{h}) \text{ für beliebige } \tilde{h}(x)$$

(Klassengrenze mit kleinster Strafe)

zu

$$\boxed{h(x) = F_2(x,c) - F_1(x,c)}$$

Beweis :

Sei $Q(x,c,h) := Q(x,d)$

und $Q(x,c,\tilde{h}) := Q(x,\tilde{d})$

Sei $\tilde{h}(x) > 0$. Dann ist $Q(x,\tilde{d}) = F_1(x,c)$.

Dann gilt für $Q(x,\tilde{d})$:

Wenn $h(x)>0$, so ist $Q(x,d) = F_1(x,c) = Q(x,\tilde{d})$

$h(x)\leq 0$, $\quad Q(x,d) = F_2(x,c)$

Dann ist auch

$h(x) = F_2(x,c) - F_1(x,c) = Q(x,d) - Q(x,\tilde{d}) \leq 0$

oder $Q(x,\tilde{d}) \geq Q(x,d)$.

Sei andererseits $\tilde{h}(x) \leqslant 0$, also $Q(x,\tilde{d}) = F_2(x,c)$.

Wenn $h(x) \leqslant 0$, so ist $Q(x,d) = F_2(x,c) = Q(x,\tilde{d})$.

$h(x) > 0$, $\quad Q(x,d) = F_1(x,c)$

und damit

$$h(x) = Q(x,\tilde{d}) - Q(x,d) \geqslant 0$$

also auch $\quad Q(x,\tilde{d}) \geqslant Q(x,d)$ $\quad$ q.e.d.

Anschaulich ist dies sowieso klar:

Die insgesamt kleinste Strafe wird bei gegebenen x und d dann erreicht, wenn jeweils das kleinste F(x,c) genommen wird,

also

in Klasse 1: $F_1(x,c)$, wenn $F_1(x,c) < F_2(x,c)$

oder $F_2(x,c) - F_1(x,c) = h(x) > 0$

in Klasse 2: $F_2(x,c)$, wenn $F_2(x,c) \leqslant F_1(x,c)$

oder $F_2(x,c) - F_1(x,c) = h(x) \leqslant 0$

Auf der Hyperfläche $h(x) = 0$ gilt $F_1(x,c) = F_2(x,c)$.

Aus $Q(x,d)$ wird damit

$$Q(x,d) \rightarrow Q(x,c) = \min\{F_1(x,c), F_2(x,c)\}$$

1.4 Veranschaulichung des Konzepts

Die Trennung der Klassen läßt sich leicht veranschaulichen. Sei $m=2, x \in R^2$, also ein Muster, das durch 2 Variable charakterisiert ist und geometrisch als Punkt einer Ebene erscheint. Punkte, die dicht beieinander liegen, lassen sich als ähnliche Muster und "zu einer Klasse gehörig" deuten. Ein Beispiel ist die charakterisierung einer chronischen Bauchspeicheldrüsenentzündung durch die chemischen Konzentrationen bestimmter Stoffe x und y im Blut der Patienten:

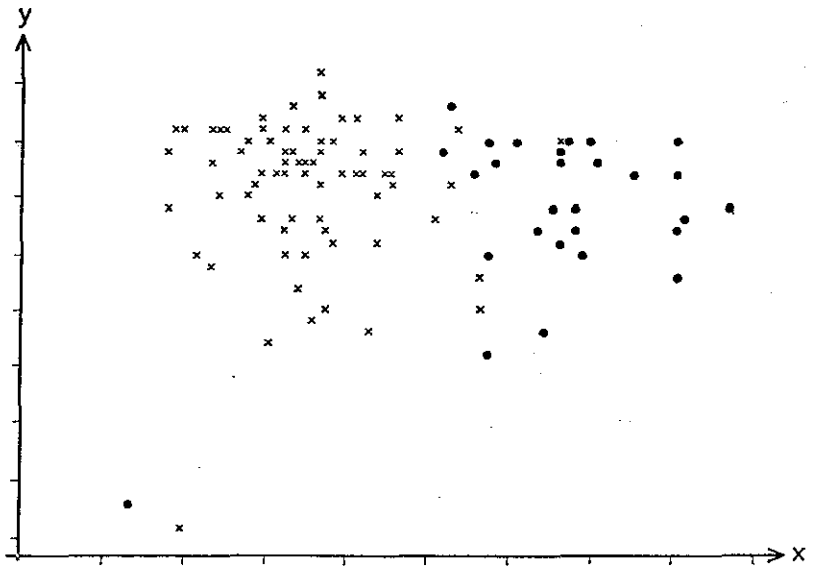


Abb.1.4a Chronical pancreatitis (×) and normal patients (●).

Abbildung aus [2]

Die Aufgabe ist also nun, den Punkthaufen (*Cluster*) der Kranken von dem Cluster der Gesunden mit möglichst geringem Fehler durch eine Linie zu trennen, wobei die Klassengrenze mit jedem neuen Patienten, jedem neuen Punkt, der hinzukommt, korrigiert werden soll.

Diese Aufgabe versuchen die in Kap. 3.3 und Kap. 6 besprochenen Verfahren so zu lösen, daß entweder der einfache Fehler beim Ziehen der Klassengrenze oder der gewichtete Fehler, also die Folgen,möglichst gering werden. Normalerweise wird dabei angenommen, daß nur *eine* optimale Klassengrenze existiert. Wie in Kap.8 gezeigt werden wird, ist diese Annahme aber nicht selbstverständlich.Unter bestimmten Umständen kann es auch zwei optimale Klasseneinteilungen geben.

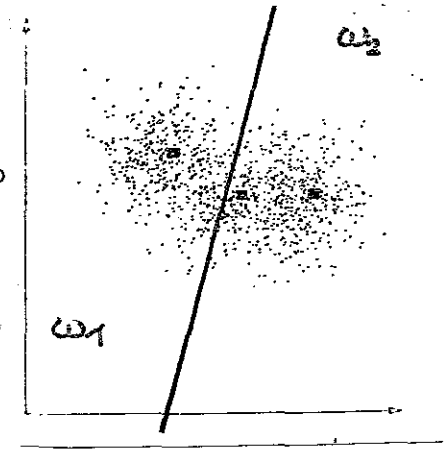


Abb.1.4b

Lage der Klassengrenze nach 1000 Iterationen.

Abb.1.4b zeigt eine Verteilung p(x),die sich aus drei sich durchdringenden Clustern besteht. Wenn nur zwei Klasseneinteilungen bestehen, so wird jedes Verfahren, das den Erwartungswert der x einer Klasse zur Iteration der Klasseneinteilung benutzt, die Klassengrenze in die Mitte des mittleren Clusters legen (Kap. 7.2.2).Der Fixpunkt der Iteration liegt dann auf der Symmetriachse der spiegelsymmetrischen Verteilung. Ist aber die Distanz $\bar{y}$ zwischen den Clustern größer als eine mit einer Funktion m(y)=1 gegebenen kritischen Distanz,

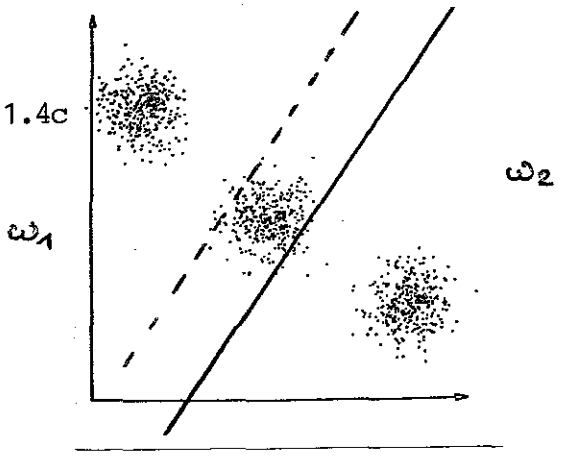
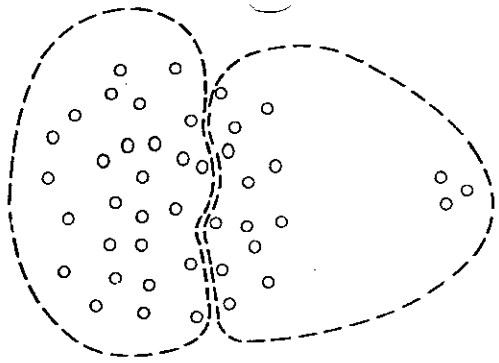


Abb.1.4c

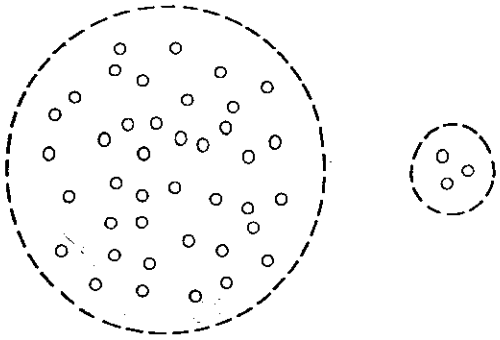
so trennen die Verfahren die drei Cluster in eine Gruppe mit zwei und eine Gruppe mit einem Cluster.
Die nebenstehende Abbildung zeigt einen solchen Fall.Die zweite möliche Lage der Klassengrenze ist gestrichelt eingezeichnet (Vgl. Abb.7.2.1c).Dies läßt sich so interpretieren, als ob

bei "unscharfen" Konturen es besser ist, einfach symmetrisch zu trennen.Bei "scharfen" Konturen aber erst ist es möglich, mit der bestehenden Information eine differenziertere Clustertrennung durchzuführen.

Dieses Problem ist auch schon andeutungsweise in der Literatur erwähnt worden. [3] 6.8.1



(a)



(b)

Obige Abbildung zeigt ein Beispiel daraus. Das aus der Lage der Punkte errechnete $R = J_1+J_2$ mit $J_i = \sum_{\Omega_i}(x-c_i)^2$ und $c_i = \frac{1}{n_i}\sum_{\Omega_i} x$ ist für die Klassenbildung a) kleiner als für b),obwohl für unser Gefühl eher b) "die richtige" Klassenbildung darstellt.

1.5

Gütekriterium und Bayes-Classifier

In Kap. 1.2a wurde als Entscheidungskriterium vorgeschlagen, sich für das ω_i mit dem größten $P(\omega_i/x)$ zu entscheiden. Zweifelsohne läßt sich damit die mittlere Wahrscheinlichkeit der Fehler klein machen, aber es wird dabei nichts darüber ausgesagt, welche Folgen eine solche Entscheidung nach sich zieht. Wenn ein Frosch eine Fliege für einen Storch hält und flieht, so ist die Strafe die er verspürt, wesentlich geringer, als wenn er einen Storch für eine Fliege hält und gefressen wird. Es ist daher sinnvoll, eine Gewichtung F_{ij} für eine Fehlentscheidung einzuführen:

F_{ij} := Strafe, wenn für Klasse i entschieden wird, aber Klasse j vorliegt.

F_{ij} soll von x und dem zu lernenden Parametervektor c abhängen: $F_{ij}=F_{ij}(x,c)$

Die Strafe oder das Risiko R, wenn das Muster x vorliegt und für ω_i entschieden wird, ist

$$R(\omega_i/x)=\sum_j F_{ij}P(\omega_j/x) \quad \textit{(Bayes-Risiko)}$$

und das *mittlere Risiko* R_i

$$R_i=\int_{\Omega_i} R(\omega_i/x)p(x)dx=\int_{\Omega_i}\sum_j F_{ij}P(\omega_j/x)p(x)dx$$

Die Gesamtstrafe aller möglichen Einordnungen ist

$$R=\sum_i R_i=\sum_i\int_{\Omega_i}\sum_j F_{ij}P(\omega_j/x)p(x)dx$$

Mit den Formeln (1.2a) und (1.2c) und der Vereinfachung

$F_{ij}(x,c)=F_i(x,c)$, wenn die Strafe nur von ω_i abhängt

ist

$$\begin{aligned} R=R(c)&=\sum_i\int_{\Omega_i} F_i(x,c)\sum_j P(\omega_j)p(x/\omega_j)dx \\ &=\sum_i\int_{\Omega_i} F_i(x,c)p(x)dx \\ &=\sum_i E_i(F_i(x,c))=E(Q(x,c)) \\ &=\sum_i J_i(c) \end{aligned}$$

$$E(x):=\int_\Omega xp(x)dx$$

Erwartungswert von x

mit

$$J_i(c):=E_i(F_i(x,c)) \qquad E_i(x):=\int_{\Omega_i} xp(x)dx$$

R(c) ist also der Erwartungswert der Gesamtstrafe aller Einordnungen von x.
Dabei wird nicht mehr $P(\omega_i)$ und $p(x/\omega_i)$ als bekannt vorausgesetzt, sondern nur $F_i(c,x)$ und $p(x)$.
Ziel des Lernprozesses wird es sein, durch iteratives Verbessern von c den Erwartungswert der Gesamtstrafe, oder, was dasselbe ist, den Erwartungswert des gewichteten Fehlers der Klassifizierung so klein wie möglich zu machen.

$$R(c) \stackrel{!}{=} \min$$

Für n=2 (2 Klassen) möchte ich die Folgerungen aus dieser Forderung ziehen; der allgemeine Fall ist bei [4] Kap.7.3 behandelt.

Mit $\Theta(a) = \begin{cases} 0 & a<0 \\ 1 & a\geq 0 \end{cases}$

ist
$$R(c) = \sum_{i=1}^{2} \int_{\Omega_i} F_i(x,c)\,p(x)\,dx$$
$$= \int_{\Omega} \Big(\Theta(F_2-F_1)\cdot F_1(x,c) + \Theta(F_1-F_2)\cdot F_2(x,c) \Big) p(x)\,dx$$

Die Forderung $R(c) \stackrel{!}{=} \min$ ist äquivalent mit $\nabla_c R(c) \stackrel{!}{=} 0$

$\Theta_1 := \Theta(F_2-F_1)$
$\Theta_2 := \Theta(F_1-F_2)$

und damit
$$0 \stackrel{!}{=} \nabla R(c) = \int_{\Omega} \Big(\nabla_c(\Theta_1 F_1) + \nabla_c(\Theta_2 F_2) \Big)\, p(x)\,dx$$
$$= \int_{\Omega} \Big(\nabla_c \Theta_1 F_1 + \Theta_1 \nabla_c F_1 + \nabla_c \Theta_2 F_2 + \Theta_2 \nabla_c F_2 \Big) p(x)\,dx$$

$\Theta(a-b) = 1-\Theta(b-a)$
$$= \int_{\Omega} \Big(\nabla_c \Theta(F_2-F_1) F_1 - \nabla_c \Theta(F_2-F_1) F_2 \Big) p(x)\,dx$$
$$+ \int_{\Omega} \Big(\Theta\, \nabla_c F_1 + \Theta\, \nabla_c F_2 \Big) p(x)\,dx$$

Der erste Summand (Variation der Klassengrenzen λ) ergibt null:

$$\int_{\Omega} \Big(\nabla_c \Theta(F_2-F_1)\cdot(F_1-F_2) \Big) p(x)\,dx \qquad \lambda := \{x / h(x)=0\}$$
$$= \int_{\Omega} \Big(\delta(F_2-F_1)\cdot \nabla_c (F_2-F_1)\cdot(F_1-F_2) \Big) p(x)\,dx$$
$$= \int_{\lambda} \Big(\underbrace{\nabla_c (F_2-F_1)}_{\lambda \text{ beschränkt}} \cdot \underbrace{(F_1-F_2)}_{=0} \Big) p(x)\,dx \quad = 0, \text{da } (F_1-F_2)=0 \text{ auf } \lambda$$

Daher ist

$$\nabla_c R(c) = \int_{\Omega} \left(\Theta_1 \nabla_c F_1 + \Theta_2 \nabla_c F_2\right) p(x)\,dx$$

$$= \sum_{i=1}^{2} \int_{\Omega_i} \nabla_c F_i\, p(x)\,dx \stackrel{!}{=} 0$$

Die Minimisierung der Gesamtstrafe geschieht also *nur noch* durch optimales Einstellen von $\underline{c}=(c_1,\ldots,c_n)$ in den einzelnen Gebieten;das optimierende Verschieben der Grenzen im ersten Summanden brachte den Anteil null.

Mit $\nabla_c F_i(x,c) := f_i(x,c)$, $j_i(c) := E_i(f_i(x,c))$

läßt sich das Lernziel umschreiben zu

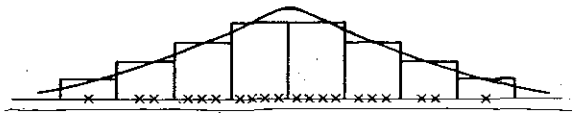
$$\sum_{i=1}^{2} E(f_i(x,c)) = \sum_{i=1}^{2} j_i(c) \stackrel{!}{=} 0$$

2. Kapitel Methoden zur Schätzung von p(x)

Im folgenden Abschnitt sollen die Charakteristika einiger Methoden zur Schätzung von p(x) besprochen werden. Allen ist gemeinsam,daß die Klassengrenzen bei dieser Art von Mustererkennung nicht durch das Optimieren eines Parameters eingestellt werden,sondern direkt aus p(x) ermittelt werden;beispielsweise als lokale Minima von p(x).

2.1 Histogramm-Methode

Der n-dim. Raum wird in gleichgroße Zellen unterteilt. Die auftretenden $\underline{x}$ werden in diese Zellen eingegliedert und für jede Zelle gezählt.



Problem:
Wenn die Zellen groß gemacht werden,so ist das Histogramm als Näherung für p(x) sehr ungenau. Wird aber die Zellengröße klein gemacht,so ist die Zahl der Zellen und damit in der Praxis die Zahl der Computerspeicherplätze groß.
Ein möglicher Ansatz aus diesem Dilemma ist die Einführung von Zellen,deren Größe von der Zahl der $\underline{x}$ abhängt,die darinnen sind.Dort,wo viele $\underline{x}$ auftreten,wird die Zellgröße kleiner und die Histogrammunterteilung feiner gemacht;wenig benutzte Zellen werden dafür größer.

2.2 Parzen window

Um die Zahl der $\underline{x}$ in einer Zelle zu ermitteln,kann man auch eine "Fensterfunktion" (*window*) definieren,die eins wird,wenn $\underline{x}$ in diese Zelle fällt,sonst aber null ist.

$$\varphi(\underline{x},\underline{x_i}) = \begin{cases} 1 & \underline{x} \text{ aus Zelle } i \;\hat{=}\; \frac{\underline{x}-\underline{x_i}}{h} \leq \frac{1}{2} \\ 0 & \text{sonstig} \end{cases}$$

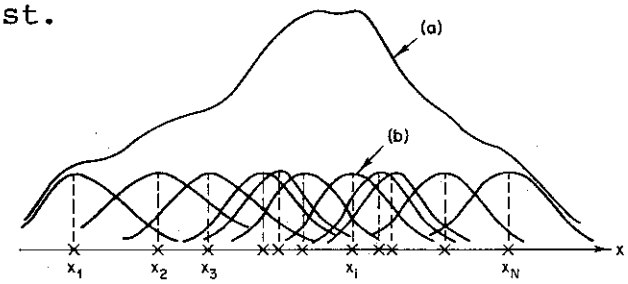
Die Zahl k der $\underline{x}$ in der Zelle i nach n Mustern $\underline{x}(1)\ldots\underline{x}(n)$ ist somit

$$k_i = \sum_{j=1}^{n} \varphi\left(\frac{x(j)-x_i}{h}\right)$$

h=Seitenlänge der Zelle

x_i=Zentrum der Zelle i

Für jede Zelle oder Hyperwürfel in $\mathbb{R}^m$ ergibt sich somit ein $p_i(x)$, deren Überlagerung $p(x)$ ist, zu $p_i(x) = k_i/(n.h^m)$
Anstelle der $\varphi(x,x_i)$, die nur 0 oder 1 werden können, lassen sich auch andere Funktionen ($\exp(-x^2)$ usw.) *(kernels)* verwenden. Wird die Zelle mit wachsender Zahl der $\underline{x}$ kleiner gemacht, so daß immer nur ein $\underline{x}$ in jeder Zelle ist, so setzt sich $p(x)$ schließlich aus vielen $p_i(x)$ zusammen, deren Zentrum jeweils ein $\underline{x}$ ist.
Macht man h, und damit die Zellengröße, sehr klein, so geht $\varphi(x,x_i)$ in eine δ-Funktion über und $p(x)$ erscheint sehr verrauscht. Macht man h dagegen groß, so ist die Näherung sehr ungenau.

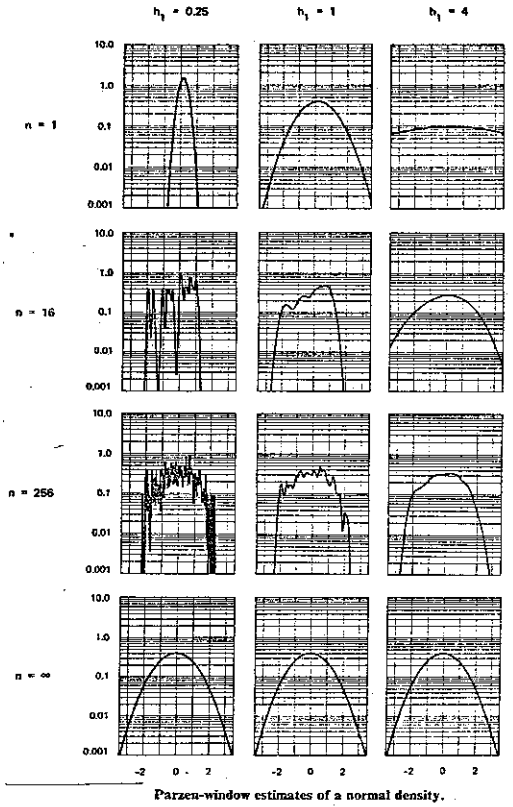


Approximation of a density function by the sum of normal kernels: (a) $p(x)$; (b) $(Nh)^{-1}\varphi[(x - x_i)/h]$.

Einen Schätzungsprozeß für $p(x)$ zeigt Abb. 2.2, der Parameter $h_1 = h/\sqrt{n}$ wird horizontal variiert.

2.3 K-Nearest Neighbor Approach

Bisher wurde $p(\underline{x})$ aus der Dichte der Punkte um ein $\underline{x}_i$ herum bestimmt. Dafür wird die Zahl der Nachbarn (Zahl der $\underline{x}$ in Zelle i) bestimmt. Man kann aber auch die Zahl der Nachbarn konstant halten und die Zellengröße bestimmen, in der sie liegen. Letzteres wird in diesem Algorithmus durchgeführt. Dazu wird für jedes $\underline{x}(i)$ eine Liste seiner k nächsten Nachbarn aufgestellt. Der Abstand des weitesten Nachbarn auf dieser Liste ist ein Maß für die Dichte der



Parzen-window estimates of a normal density.

Abb. 2.2

Muster $\underline{x}$ um $\underline{x}(i)$, also auch von $p(\underline{x}(i))$.

Klassifikationsregel:

Seien x(1)...x(n) schon klassifiziert.

Dann wird x(n+1) folgendermaßen eingeordnet:
Die k nächsten Nachbarn werden ausgesucht.

Seien n_1 davon aus ω_1, n_2 aus ω_2, so ist der Fehler am kleinsten, wenn für $\underline{x}(n+1)$

$$\text{Klassifizierung}(\underline{x}(n+1)) \text{ sei} \begin{cases} \omega_1, \text{wenn } n_1 < n_2 \\ \omega_2, \text{wenn } n_2 > n_1 \end{cases} \quad \begin{matrix} \text{(Mehrheitsvotum der} \\ \text{Nachbarn)} \end{matrix}$$

Der Fehler dieser Entscheidung wird in ,S.105 behandelt.

Nachteile:

a) Alle $\underline{x}$ müssen gespeichert sein (Computerspeicherplatz!)

b) Der einfache Fehler wird klein, nicht aber der gewichtete.

Mit der Einführung einer Bewertung der Klassifikation durch einen überwachenden Lehrer lassen sich "unnütze" $\underline{x}$ aussondern, die nicht gespeichert werden, und die Zahl der $\underline{x}$ auf wenige Repräsentanten beschränken, die dicht an den Klassengrenzen lokalisiert sind und zur Klassentrennung ausreichen. *(HART 1968)*

2.4 Schätzung von p(x) und Ermittlung der Klassenzahl

Angenommen, p(x) sei als Überlagerung von Kernels wie in b) beschrieben dargestellt. Dann läßt sich ein Algorithmus angeben, mit dem die Zahl der Klassen ermittelt werden kann.

Sei x_1 zufällig gewählt. Dann ist

$$p_1(x) := \varphi(x, x_1)$$

$\lim_{n \to \infty} p_n(x) = p(x)$

Suche x_2 mit $\quad p_1(x_2) = \max_i \left\{ \varphi(x_i, x_1) \right\}$

Suche x_3 mit $\quad p_2(x_3) = \max_i \left\{ \frac{1}{2} \left(\varphi(x_i, x_1) + \varphi(x_i, x_2) \right) \right\}$

$\vdots$

Suche x_n mit $\quad p_{n-1}(x_n) = \max_i \left\{ \frac{1}{n} \sum_{j=1}^{n} \varphi(x_i, x_j) \right\}$

Die Entscheidung darüber, welches x_i zu wählen ist, hängt immer davon ab, wie nahe x_i beim Maximum der iterierten Wahrscheinlichkeitsdichte $p_n(x)$ ist.

Trägt man nun $p_i(x)$ als Funktion des n-ten Schrittes auf, so erhält man ein Diagramm ähnlich Abb.

Die nach unten gerichteten Zacken (lokale Minima) von $p_n(x)$ zeigt den Übergang von einer Klasse zur nächsten an. Die Zahl der Klassen läßt sich aus der Zahl der Minima ablesen: In Abb. 2.4 sind es 5 Klassen.

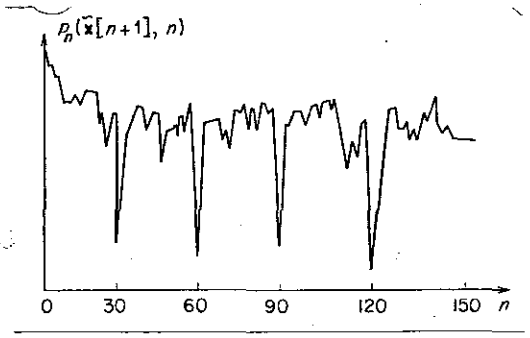


Abb. 2.4 aus [11], S. 113

Fehler der Methode:

Bei einer quadratischen Abstandsfunktion als Kernel

$$\varphi(x, x_i) := 1 - (x - x_i)^2$$

ist

$$p_n(x_{n+1}) = \max_i \left\{ 1 - \frac{1}{n} \sum_{j=1}^{n} (x_i - x_j)^2 \right\}$$

$$= 1 - \min_i \left\{ \frac{1}{n} \sum_{j=1}^{n} (x_i - x_j)^2 \right\}$$

$$= 1 - \min_i M\left((x - x_i)^2\right)$$

Diese Methode minimisiert bei einer quadratischen Abstandsfunktion den quadratischen Mittelwert.

Kapitel 3 Clustersuch algorithmen

In 2.4 wurde ein Verfahren vorgestellt,bei dem,nach einer Schätzung von p(x) mit Hilfe von Kernels,die Anzahl der Klassen bestimmt wurde.
Im folgenden Abschnitt möchte ich verschiedene Verfahren besprechen,mit denen man ohne Schätzung von p(x) die Klassenzahl und die Klassengrenzen bestimmen kann,wobei nur der einfache Fehler der Klassifizierung klein gemacht wird.
Alle Verfahren benutzen feste Parameter,die erst durch wiederholtes Experimentieren mit dem jeweiligen Datenmaterial des speziellen Problems optimal gewählt werden können.
Um die Verfahren zu verdeutlichen,sind sie teilweise in Form von Flußdiagrammen aufgeführt.

3.1 Maximaler Abstands-Algorithmus

Angenommen,alle $\underline{x} \in \Omega$ liegen vor und der Abstand D ist willkührlich gewählt.
Ein zufälliges x_i wird als erstes Cluster-Zentrum gewählt.
Alle $\underline{x}$,die einen Abstand kleiner als D zu x_i haben,gehören nun zum Cluster 1und werden aussortiert.
Für den Rest wird für Cluster 2 wieder obiges Verfahren angewendet und solange wiederholt,bis alle $\underline{x}$ eingeteilt sind.
Wenn nicht alle $\underline{x}$ vorliegen,läßt sich das Verfahren auch sequentiell anwenden:

Z_1=Zentrum 1:=$\underline{x}(1)$

Für alle schon bestehenden Cluster prüfen:

Wenn $|x(n)-Z_i| < D$,dann ist $\underline{x}$ aus Cluster i

Sonst: $Z_m := \underline{x}(n)$

Wichtige Daten für dieses Verfahren sind die kleinste und die größte Abweichung der $\underline{x}$ von Z_i und die Varianz.

3.2 Maximin-Algorithmus

Alle $\underline{x}$ liegen vor, D sei willkürlich gewählt.

Dann sei $Z_1 := x(1)$

Es habe x(i) den größten Abstand zu x(1). Dann ist

$Z_2 := x(i)$

Klassenzuordnung:

Alle $\underline{x}$ werden zu den ihnen am nächsten liegenden Z_i zugeordnet.

Nun wird von allen Klassen(Cluster)dasjenige x(j)heraus-gesucht, das den größten Abstand zu seinem Zentrum Z_i hat.

Ist dieser Abstand für alle i,k größer als $|Z_i - Z_k|/D$?

Wenn ja, so ist

$Z_{neu} =: x(j)$

und es geht weiter bei "Klassenzuordnung"

Wenn nein, so ist das Verfahren zu Ende.

3.3 K-Mean-Algorithmus

Bei diesem Verfahren wird die Straffunktion $J_k = \sum_i (x(i) - Z_k)^2$ minimisiert.

Am Anfang werden k Clusterzentren gewählt, z.B. die ersten x(1)...x(k).

Klassenzuordnung:

Alle $\underline{x}$ werden zu den ihnen am nächsten liegenden Z_i zugeordnet.

Dann werden die Z_i durch den Mittelwert aller Klassen-mitglieder ersetzt:

$$Z_i := \frac{1}{n_j} \sum_j^{n_j} x_j$$

n_j = Zahl der Klassenmitglieder

Wenn sich ein Z_i dabei verändert hat, so ist eine neue Klasseneinteilung notwendig und es geht weiter bei "Klassenzuordnung"

Wenn die Lage aller Zentren gleich geblieben ist, so ist der Algorithmus zu Ende.

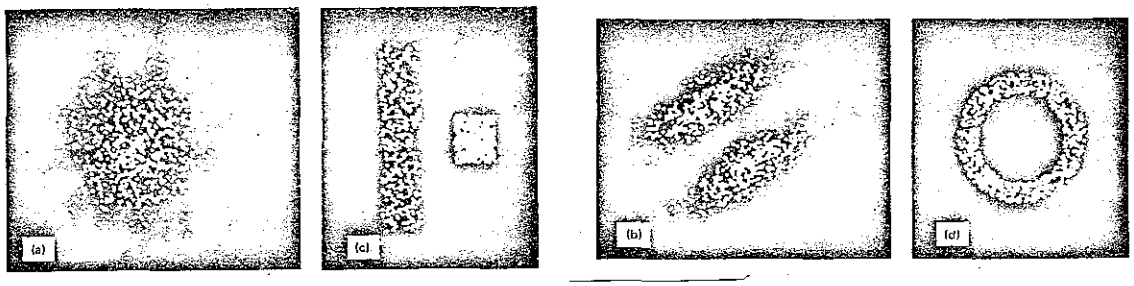
3.4 Isodata-Algorithmus

Dieses Verfahren ist ein heuristischer Ansatz,der in der Praxis an Computern entstanden ist.
Es ist relativ kompliziert und besteht aus 14 Grundschritten, die ich nicht alle aufführen möchte.(siehe [9] 3.3.6)

1. Sechs Prozeßparameter werden gewählt,darunter k und θ_N.
2. Eine Klasseneinteilung aller $\underline{x}$(wie in b)und c)) wird durchgeführt.Es werden k Klassen angenommen.
3. Alle Cluster mit weniger als θ_N Mitgliedern werden aufgelöst und zu Punkt 2 übergegangen.
   Wenn keine auflösbaren Cluster mehr da sind,folgt Punkt 4
4. Die Zentren Z_i werden wie in 3.3 durch den Schwerpunkt der x_i der Klasse ersetzt.
5. Nun wird die mittlere Abweichung der x_i von ihrem Klassenzentrum Z_i für jede Klasse bestimmt.
6. Diese Abweichungen werden über alle Klassen gemittelt.

In den folgenden Schritten werden die berechneten Abweichungen dazu benutzt,Cluster zu spalten oder mehrere wieder zu vereinen.Anschließend berechnet man alle Abweichungen neu und geht wieder zu Punkt 2 über.
Dies wird solange durchgeführt,bis bestimmte Parametergleichungen erfüllt sind.

Um die Probleme der Clusteranalyse zu verdeutlichen,sind in Abb. 3.4 verschiedene Clusterverteilungen gezeigt,die alle die gleichen Varianzen haben.



Data sets having identical second-order statistics.

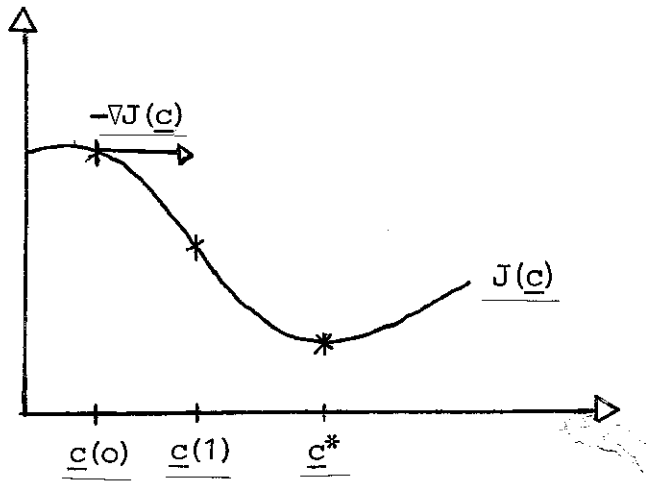
aus [3]

Abb. 3.4

Kapitel 4 Methoden der Parametervariation

4.1 Die Gradientenmethode

In Kap. 1.5 zeigte sich, daß man den gewichteten Fehler $R(\underline{c})$ mit der Variation eines Parametervektors $\underline{c}$ minimal machen kann. Dazu betrachten wir allgemein das Problem, das Minimum $J(\underline{c}^*)$ einer vorgegebenen Funktion $J(\underline{c})$ zu finden.
Dies löst sich am Besten mit Hilfe folgender Iteration:



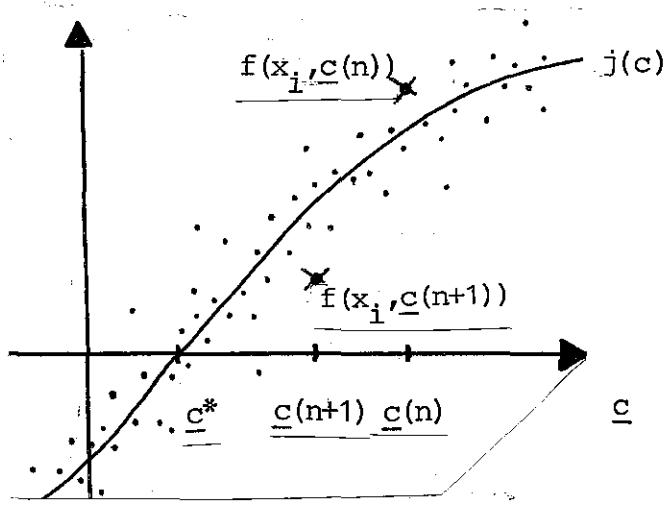
Sei ein Wert $\underline{c}_o$ gegeben. Dann gibt der Gradient $\nabla_c J(\underline{c})$ in $\underline{c}_o$ die Richtung der stärksten Steigung und $-\nabla_c J(\underline{c}_o) := \underline{g}$ in die stärkste Gefällerichtung. Dieser Gefälle-Vektor zeigt daher auch in Richtung des Minimums von $J(\underline{c})$ und ermöglicht ein verbessertes $\underline{c}(1) = \underline{c}_o + \underline{g}$, das dichter an $\underline{c}^*$ liegt.
Der allgemeine Algorithmus enthält noch einen Proportionalitätsfaktor $\underline{\underline{\gamma}}$, um je nach Funktion die Schrittweite zu korrigieren:

$$\underline{c}(n+1) = \underline{c}(n) - \underline{\underline{\gamma}} \cdot \nabla_c J(\underline{c}(n))$$

Wenn $\underline{c}$ ein Vektor ist, so ist $\underline{\underline{\gamma}}$ im allgemeinen eine Matrix; häufig reicht es, die Einheitsmatrix mit dem skalaren Faktor γ zu nehmen. Für schnellere Konvergenz empfiehlt es sich manchmal auch, eine Matrix mit nichtdiagonalen Termen $\neq 0$ zu verwenden.
Bei sehr starken und lokal ausgeprägten Minima bewirkt der Gradient große anstelle von kleinen Schritten. Bei solchen Funktionen werden stattdessen $1/\nabla_c J(\underline{c}) = \dfrac{\nabla_c J(\underline{c})}{|\nabla_c J(\underline{c})|^2}$ bessere Ergebnisse liefern.
Für das Problem, das Minimum von $R(\underline{c})$ zu finden, ist aber der Erwartungswert $J(\underline{c}) = \int_\Omega F(\underline{x}, \underline{c})\, d\underline{x}$ als Funktion explizit garnicht gegeben. Mit gegebenen $\underline{x}$ und $\underline{c}$ läßt sich nur $F(\underline{x}, \underline{c})$ als "verrauschte Form von $J(\underline{c})$" bilden.
Wie läßt sich nun trotzdem das Minimum finden?

4.2 Der Robbins-Monro Algorithmus

Robbins und Monro(1951) gaben als erste eine Methode an, wie man die Nullstelle einer"verrauschten"Funktion findet.



Sei $\underline{x}$ eine stochastische Variable,so ist $f(\underline{x},c)$ ebenso eine.

Würde man sehr viele $\underline{x}$ abwarten,so könnte man den Erwartungswert $j(c)$

$$j(c) := E\left(f(\underline{x},\underline{c})\right)$$

angenähert bilden und so z.B. mit der Gradientenmethode das Minimum von $j(c)^2$,also $\underline{c}^*$ mit $j(c^*)=0$ finden.

Die Methode der *stochastischen Approximation* von Robbins und Monro kommt aber ohne das lange Abwarten aus.

Mit ähnlichen Überlegungen wie bei der Gradientenmethode erhielten sie ein dem $\underline{c}^*$näheres $\underline{c}(n+1)$:

$$\underline{c}(n+1)=\underline{c}(n)-\gamma_n f(x(n),c(n))$$

Für die Konvergenz dieses Algorithmus läßt sich(s.[5] S.207) u.a.zeigen,daß $\lim_{n\to\infty} P(c^*=c(n)) = 1$

der Abstand $|c^*-c(n)|$ in endlicher Zeit also sehr klein wird, wenn folgende Voraussetzungen erfüllt sind:

$f(\underline{x},\underline{c})$, $j(c)$ } beschränkt im betrachteten Intervall

$E(f(\underline{x},\underline{c})-j(\underline{c}))=0$

$\lim \gamma_n = 0$

$\sum_{n=1}^{\infty} \gamma_n > \infty$

$\sum_{n=1}^{\infty} \gamma_n^2 < \infty$

Dieses Verfahren läßt sich verallgemeinern zum Satz von Dvoretzky.

4.3 Der Satz von Dvoretzky

Sei T_n eine Transformationsabbildung und es gelte

$$\underline{c}(n+1)=T_n(\underline{c}(1),\underline{c}(2),\ldots,\underline{c}(n)) + \underline{x}(n)$$

($\underline{x}(n)$: Zufallsvariable)

Unter folgenden Voraussetzungen

1) $$T_n(\underline{r}_1 \ldots \underline{r}_n)-\underline{c}^* \leqslant \max\left\{\alpha_n, (1+\beta_n)\cdot|\underline{r}_n-\underline{c}^*|-\gamma_n\right\}$$

mit $\underline{r}_1,\ldots,\underline{r}_n$ beliebig

$$\lim_{n\to\infty}\alpha_n = 0 \quad , \quad \sum_{n=1}^{\infty}\beta_n < \infty \quad , \quad \sum_{n=1}^{\infty}\gamma_n > \infty$$

2) $$\sum_{n=1}^{\infty} E(\underline{x}_n^2) < \infty$$

3) $$E(\underline{x}_n/\underline{c}(1),\ldots,\underline{c}(n))=0$$

Die Funktion ist gleichmäßig verrauscht, unabhängig vom Index n

gelten folgende Aussagen:

$$\lim_{n\to\infty} E\left((\underline{c}(n)-\underline{c}^*)^2\right)= 0$$

und $$P\left(\lim_{n\to\infty}\underline{c}(n)=\underline{c}^*\right)=1$$

Mit folgender Transformation ergibt sich der Algorithmus von Robbins und Monro: $\alpha_n := \gamma_n$, $\beta_n := \gamma_n^2$, $\gamma_n := \gamma_n$

$$T_n(\underline{c}(1),\ldots,\underline{c}(n)) =: \underline{c}(n)+\gamma_n(j^*-j(\underline{c}(n)))$$

und

$$\underline{x}(n) =: \gamma_n(j(\underline{c}_n)-f(\underline{x}_n,\underline{c}_n))$$

ist

$$\underline{c}(n+1)=\underline{c}(n)+\gamma_n(j^*-f(\underline{x}_n,\underline{c}_n))$$

Für eine Nullstelle mit $j^*=0$ reduziert sich dies zu

$$\underline{c}(n+1)=\underline{c}(n)-\gamma_n f(\underline{x}_n,\underline{c}_n) \qquad \text{(Robbins-Monro)}$$

4.4 Der stochastische Lernalgorithmus

Das Problem beim stochastischen Lernen ist es,das mittlere Risiko $R(\underline{c})$ der Klassifizierung zu minimisieren.
Dies läßt sich mit der Gradientenmethode durchführen, wenn bei $R(\underline{c}) = \sum_i J_i(\underline{c})$ die Straffunktionen $J_i(\underline{c})$ bekannt sind. Beim stochastischen Lernen sind aber die Straffunktionen $J_i(\underline{c})$ nur als Erwartungswerte der stochastischen Funktionen $F_i(\underline{x},\underline{c})$ zu ermitteln.Trotzdem lassen sich Verfahren angeben,wie sich mit iterativer Veränderung des Parametervektors $\underline{c}$ die stochastischen Funktionen,und damit auch $R(\underline{c})$, minimisieren lassen.
Da minimales $F_i(\underline{x},\underline{c})$ erreicht wird,wenn der Gradient $f(\underline{x},\underline{c}) := \nabla_{\underline{c}} F_i(\underline{x},\underline{c})$ null wird,so ist mit dem stochastischen Algorithmus von Robbins und Monro zur Nullstellenbestimmung einer stochastischen Funktion $f(\underline{x},\underline{c})$ auch eine Methode gefunden, $F_i(\underline{x},\underline{c})$,damit $J_i(\underline{c})$ und damit auch $R(\underline{c})$ zu minimisieren.
Die Konvergenzaussagen von Robbins und Monro gelten damit auch für den stochastischen Lernalgorithmus

$$\underline{c}_i(n+1) = \underline{c}_i(n) - \gamma_n \frac{\partial F_i(\underline{x},\underline{c}_i)}{\partial \underline{c}_i}$$

$$\underline{c}_j(n+1) = \underline{c}_j(n) \quad \text{mit } j \neq i \quad \text{und} \quad h_{ij}(x) < 0$$

Formel (4.4)

wobei für γ_n gelten muß

$$h_{ij}(x) = F_i(\underline{x},\underline{c}_i) - F_j(\underline{x},\underline{c}_j)$$

$$\lim_{n\to\infty} \gamma_n = 0$$

$$\lim_{n\to\infty} \sum_{i=1}^{n} \gamma_i = \infty$$

$$\lim_{n\to\infty} \sum_{i=1}^{n} \gamma_i^2 = M < \infty$$

Kapitel 5 Lineare Klassifizierung

Bei den bisherigen Verfahren zur Trennung der Klassen wurde über die Form der Klassengrenzen nichts ausgesagt. Bei den anschließend folgenden Verfahren werden die Hyperflächen der Klassengrenzen stückweise oder ganz durch Hyper*ebenen* ersetzt. Im zweidimensionalen Fall sind dies einfache Geraden.

Die Gründe für diesen Ansatz möchte ich zuerst etwas näher erläutern. Stetige und differenzierbare Funktionen, wie sie in der Praxis meist auftreten, lassen sich auch als unendliche Reihe (z.B. Fourierreihe) entwickeln

$$f(\underline{y}) = \sum_{i=1}^{\infty} c_i \cdot \varphi_i(\underline{y})$$

und in der Näherung $n \neq \infty$

$$\approx \sum_{i=1}^{n} c_i \varphi_i(\underline{y}) = \underline{\varphi}(\underline{y}) \cdot \underline{c}$$

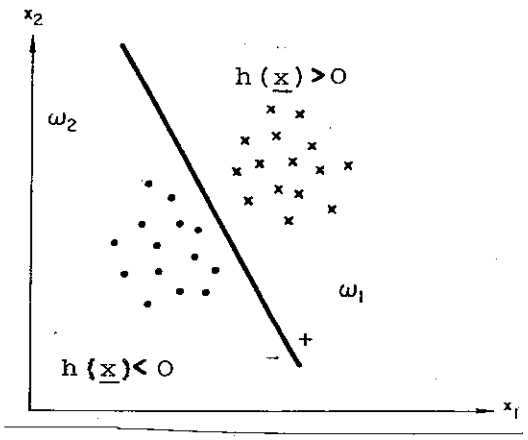
Mit einer geschickten *Feature-selection (s.Kap.* 1.2 *)* lassen sich die Ursprungsmuster $\underline{y}$ so transformieren, daß die resultierenden $\underline{x} = \underline{\varphi}(\underline{y})$ die lineare Gleichung erfüllen

$$f(\underline{x}(\underline{y})) = \underline{x} \cdot \underline{c}$$

Wenn die Hyperfläche $h(\underline{x}) = 0$ eine Hyperebene ist, gilt

$$h(\underline{x}) = \sum_{i=1}^{n} x_i c_i + c_{n+1} = \underline{x} \cdot \underline{c} = 0$$

mit $\underline{x} = (x_1, \ldots, x_n, 1)$ und $\underline{c} = (c_1, \ldots c_n, c_{n+1})^T$



Die Klassifikationsvorschrift lautet somit für den 2-dim.Fall mit zwei Klassen

$$h(\underline{x}) \begin{cases} > 0 \text{ , also } \underline{x} \in \Omega_1 \\ < 0 \text{ , also } \underline{x} \in \Omega_2 \end{cases}$$

$$= x_1 c_1 + x_2 c_2 + c_3$$

Wenn alle $\underline{x}$ schon vorliegen und nur noch die Lage der günstigsten Hyperfläche festzulegen ist,so läßt sich dies mit

$$\begin{matrix} F_1 := \underline{x}(1).\underline{c} \\ \cdot \\ \cdot \\ \cdot \\ F_n := \underline{x}(n).\underline{c} \end{matrix} \quad =: \underline{F} = \underline{\underline{X}}.\underline{c} \qquad \text{mit} \quad \underline{\underline{X}} := \begin{pmatrix} \underline{x}(1) \\ \vdots \\ \underline{x}(n) \end{pmatrix}$$

in das Problem überführen,ein möglichst "günstiges" $\underline{F}$ durch Variieren von $\underline{c}$ zu erreichen.Die verschiedenen Verfahren zur linearen Klassentrennung unterscheiden sich hauptsächlich in den Forderungen,die bei der Optimierung von $\underline{c}$ an ein "günstiges F"gestellt werden.
Meistens wird dabei der einfache Fehler minimiert.

5.1 Perzeptron Straffunktion

In diese Iteration greift(s.Kap.1.2a)eine Lehrerentscheidung ein.
Bei der Iteration wird versucht,die Abstände der falsch klassifizierten $\underline{x}$ von der jeweiligen Trennlinie so klein wie möglich zu machen.

$$F(\underline{x},\underline{c}) = \sum_{\Omega^*} (\underline{x}.\underline{c}) \quad \text{wobei } \underline{x} \in \Omega^*, \text{der Menge aller falsch klassifizierten } \underline{x}$$

5.2 Relaxationsprozeduren

Die Straffunktion $F(\underline{x},\underline{c})$ in a) gibt bei kleinem $\underline{c}$ zu kleine, bei großem $\underline{x}$ zu große Schritte zur $\underline{c}$-Korrektur.Mit den nichtkorrigierten Vektoren $\underline{x}=(x_1,..,x_n)$ und $\underline{c}=(c_1,...,c_n)^T$ ist durch $\underline{x}.\underline{c}-b=0$ eine Hyperfläche definiert.
Relaxations verfahren versuchen,den Abstand der fehlklassifizierten $\underline{x}$ zu dieser Hyperfläche durch Variation der $\underline{c}$ möglichst klein zu machen,indem als Straffunktion das mittlere relative Abstandsquadrat gewählt wird:

$$F(\underline{x},\underline{c}) = \frac{1}{2} \sum_{\underline{x} \in \Omega^*} \left(\frac{\underline{x}.\underline{c}-b}{|\underline{x}|} \right)^2$$

Nachteile von a) und b):

Bei jedem Schritt der Prozeduren, die solange wiederholt werden, bis alle $\underline{x}$ fehlerfrei eingegliedert wurden ($\Omega^* = \emptyset$), muß ein Lehrer da sein, der angibt, ob die Klassifikation richtig war. Sind die Klassen aber nicht linear separierbar ($\Omega_1 \cap \Omega_2 \neq \emptyset$), so konvergiert der Algorithmus nicht; Ω^* ist immer $\neq \emptyset$.

Wenn der Parametervektor $\underline{c}$ ohne direkten Lehrer verbessert wird, so ergibt sich aus b) die Methode von

5.3 Widrow-Hoff

Die Straffunktion ist hier

$$F(\underline{x},\underline{c}) = \left\| \underline{\underline{X}} \cdot \underline{c} - \underline{b} \right\|^2$$

und für $\underline{c}$ gilt der stochastische Algorithmus

$$\underline{c}(n+1) = \underline{c}(n) - \gamma_n \underline{\underline{X}}^T (\underline{\underline{X}} \underline{c}_n - \underline{b}) \quad , \underline{c}(0) \text{ zufällig}$$

Da die Anzahl N der Muster meist größer ist als die Zahl M der Dimensionen von $\underline{x}$, so empfielt es sich, die NxM Matrix $\underline{\underline{X}}$ wieder zu zerlegen und den Iterationsprozeß sequentiell durchzuführen, um Computerspeicherplatz zu sparen:

$$\underline{c}(n+1) = \underline{c}(n) + \gamma_n \left(b_n - \underline{c}(n) \underline{x}(n) \right) \underline{x}(n) , \underline{c}(o) \text{ zufällig}$$

$$\lim_{n \to \infty} \gamma_n = 0$$

Nachteil:

Der Algorithmus optimiert zwar $\left\| \underline{\underline{X}} \cdot \underline{c} - \underline{b} \right\|^2$ bei gegebenen $\underline{b}$, aber es ist durchaus nicht gesagt, daß das angenommene $\underline{b}$ eine gute Trennlinie ist.

Falls die Klassen linear separierbar sind, existieren $\underline{c}^*$ und $\underline{b}^*$ mit $\underline{\underline{X}} \underline{c}^* = \underline{b}^*$. Wenn $\underline{b}^*$ bekannt wäre, würde man mit obiger Methode leicht $\underline{c}^*$ bestimmen können. Das Problem erweitert sich also darauf, außer einem optimalen $\underline{c}^*$ auch ein optimales $\underline{b}^*$ zu finden.

5.4 Ho-Kashyap

Dieses Verfahren löst das Problem,sowohl für $\underline{c}$ als auch für $\underline{b}$ optimale Werte zu finden ,indem nicht nur für das $\underline{c}$,sondern auch für $\underline{b}$ ein stochastischer Algorithmus angesetzt wird.

Das Minimum der Straffunktion von c) ist gegeben,wenn

für $\underline{c}$ $$\nabla_c F(\underline{x},\underline{c},\underline{b}) = 2\,\underline{\underline{X}}^T(\underline{\underline{X}}.\underline{c}-\underline{b}) \stackrel{!}{=} 0$$

und für $\underline{b}$ $$\nabla_b F(\underline{x},\underline{c},\underline{b}) = -2(\underline{\underline{X}}\underline{c}-\underline{b}) \stackrel{!}{=} 0$$

Mit Robbins und Monro lassen sich Algorithmen angeben, um die Nullstellen dieser "verrauschten"Funktionen zu bestimmen.Wenn man mit

$$\underline{c}=\underline{\underline{\tilde{X}}}.\underline{b} \qquad \underline{\underline{\tilde{X}}}=(\underline{\underline{X}}^T.\underline{\underline{X}})^{-1}\underline{\underline{X}}^T$$

die beiden Algorithmen zusammenfaßt, ergibt sich

$$\underline{b}(n+1)=\underline{b}(n)-\gamma_n\tfrac{1}{2}\left(\nabla_b F(\underline{x},\underline{c},\underline{b})-\left|\nabla_b F(\underline{x},\underline{c},\underline{b})\right|\right)$$

Die Differenz der Gradienten wurde eingeführt,um alle positiven Komponenten der $\underline{b}$-Korrektur zu eliminieren, so daß $\underline{b}(n+1)$ in allen Komponenten positiv bleibt.

Mit Einführung des Fehlervektors $\epsilon_n=\underline{\underline{X}}\underline{c}(n)-\underline{b}(n)$

und dem positiven Anteil

$$\epsilon_n^+=\tfrac{1}{2}(\epsilon_n+|\epsilon_n|)$$

ist $$\boxed{\underline{b}(n+1)=\underline{b}(n)+2\gamma_n\epsilon_n^+}\,,\quad \underline{b}(1)>0 \text{ zufällig}$$

Dieses Verfahren wird manchmal auch als LMSE (Least mean square error) Verfahren bezeichnet,da die Straffunktion, die ja auch den Fehler ausdrückt,als Quadratisches Mittel minimalisiert wird.

Es läßt sich zeigen(s.[9] 5.3.3),daß bei separierbaren Klassen nach endlich vielen Schritten $\epsilon_n=0$ ist.

Da dies bei nichtseparierbaren Klassen nicht der Fall ist, bildet dieses Verfahren gleichzeitig einen Test für Separierbarkeit.

Die Matrix $\underline{\underline{\tilde{X}}}$ muß nur einmal berechnet werden;bei sequentiellen Prozessen kann dies auch rekursiv erfolgen(*Bodewig 1956*).

5.5 Eine stochastische Approximation mit einfachem Fehler

Sei die Hyperfläche, mit der zwei Klassen voneinander getrennt werden, mit

$$h^*(\underline{x}) = P(\omega_1/\underline{x}) - P(\omega_2/\underline{x}) \qquad \text{gegeben.}$$

Dann ist

$$h^*(\underline{x}) \begin{cases} <0 & \underline{x} \in \Omega_2 \\ >0 & \underline{x} \in \Omega_1 \end{cases} \qquad \text{z.B.} \quad \begin{array}{l} h^*(x) := \alpha := -1 \\ h^*(x) := \alpha := +1 \end{array}$$

Die lineare Näherung von $h^*(\underline{x})$ sei

$$h(\underline{x}) = \sum_{i=1}^{N} c_i x_i = \underline{c} \cdot \underline{x}$$

mit dem erwarteten Fehler

$$\epsilon^2 = E\left\{(h(\underline{x}) - h^*(\underline{x}))^2\right\}$$

$$= E\left\{(\underline{c}\underline{x} - \alpha)^2\right\} \qquad F_i(x,c) := (\underline{c} \cdot \underline{x} - \alpha)^2$$

Das Minimum des Fehlers ist mit dem stoch. Lernalgorithmus

$$\boxed{\underline{c}(n+1) = \underline{c}(n) + \gamma_n \cdot \left(\alpha_n - \underline{c}(n) \cdot \underline{x}(n)\right) \underline{x}(n)}$$

Trotz anderer Voraussetzungen ist dieser Algorithmus ähnlich dem von Widrow-Hoff.
Für die Konvergenz $\epsilon^2 \to 0$ gelten die Vorraussetzungen von Robbins-Monro; dazu muß $E(\underline{x}\underline{x}^T)$ nichtsingulär sein.

5.6 Abstandsverfahren

Zu der Gruppe der Verfahren, die Parameter variieren, um das Minimum einer Straffunktion zu finden, gehören auch jene, die Straffunktionen verwenden, in denen der Abstand des Musters zu einem zu lernenden Klassenprototyp enthalten ist:

$$F(\underline{x},\underline{c}) = F(\varphi(\underline{x}) - \underline{c})$$

$\varphi(x)$ ist eine Funktion, die dem jeweiligen Problem entsprechend gewählt wird.

Ein Beispiel ist

$$F(\underline{x},\underline{c}) = \tfrac{1}{2}(\varphi(\underline{x}) - \underline{c})^2 \qquad \textit{(Bravermann)}$$

Hierbei ist die Hyperfläche zwischen den Klassen i und j mit

$$h_{ij}(\underline{x},\underline{c}_i,\underline{c}_j)=F_i(\underline{x},\underline{c}_i)-F_j(\underline{x},\underline{c}_j)$$

gegeben.

$$=(\underline{c}_j-\underline{c}_i)\underline{\Psi}(\underline{x}) + |c_i|^2 - |c_j|^2$$

Die Entscheidungsregel lautet dann

$$h_{ij}(\underline{x}) \begin{cases} >0, \text{so } \underline{x}\in\Omega_j \\ <0, \text{so } \underline{x}\in\Omega_i \end{cases}$$

und die Iteration

$$\underline{c}_i(n+1)=c_i(n)+\gamma_n(\underline{\Psi}(\underline{x}(n)-\underline{c}_i(n)))$$

$$\underline{c}_j(n+1)=\underline{c}_j(n) \quad \forall\, j\neq i \quad .$$

Verwendet man die Straffunktion

$$F_i(\underline{x},\underline{c})=\tfrac{1}{2}(\underline{\Psi}(\underline{x})-\underline{c}_i)^2+\sum_{j\neq i}|\underline{c}_j|^2 \qquad \textit{(Dorofeyuk)}$$

so ergibt der Gradient von $F_i(\underline{x},\underline{c})$ die selbe Funktion und damit auch den selben Algorithmus wie vorher. Nur die Entscheidungsfunktion

$$h_{ij}(\underline{x})=F_i(\underline{x},\underline{c}_i)-F_j(\underline{x},\underline{c}_j)$$

$$=(\underline{c}_j-\underline{c}_i)\underline{\Psi}(\underline{x})$$

hat sich durch die Summe der $\underline{c}_j$-Quadrate geändert: Sie ist kürzer geworden.

Im einfachsten Fall ist

$$\underline{\Psi}(\underline{x}) := \underline{x}$$

und damit

$$F_i(\underline{x},\underline{c})=\tfrac{1}{2}(\underline{x}-\underline{c}_i)^2$$

mit der Iteration

$$\underline{c}_i(n+1)=\underline{c}_i(n)+\gamma_n(\underline{x}(n)-\underline{c}_i(n))$$

$$\underline{c}_j(n+1)=\underline{c}_j(n) \quad \forall\, j\neq i$$

Dabei ist $\underline{c}_i$ der Parametervektor der Klasse i, für die gilt

$$F_i(\underline{x},\underline{c}) = \min_k \{F_k(\underline{x},\underline{c}_k)\}$$

5.7 Andere Verfahren

Die vorgelegte Auswahl an Verfahren ist sicher nicht vollständig. Beispielsweise wären noch das Verfahren der linearen Programmierung +) und die Methode der Potentialfunktionen ++) zu erwähnen. Aus Platzgründen möchte ich aber nicht näher darauf eingehen.

+) Die lineare Programmierung versucht, bei gegebenem Kostenvektor $\underline{w}$ die lineare Straffunktion $F(\underline{w},\underline{c}) = \underline{w}.\underline{c}$ durch Verbesserung des $\underline{c}$ mit Hilfe des "Simplex-Algorithmus" so klein wie möglich zu machen. Dabei muß einer Nebenbedingung $\underline{\underline{A}}.\underline{c} \geq \underline{\beta}$ genügt werden.
Diese relativ komplizierte Methode eignet sich besonders für Probleme, bei denen Nebenbedingungen beachtet werden müssen. Beispielsweise läßt sich das Mustererkennen des Perzeptrons so formulieren, daß die Lehrerentscheidung (mißklassifizierte $\underline{x}$) in die Matrix $\underline{\underline{A}}$ eingeht und damit eine Nebenbedingung bildet.

++) Die Methode der Potentialfunktionen ordnet jedem auftretenden $\underline{x}$ eine Ladung zu und betrachtet das Gesamtpotential und seinen Verlauf als Überlagerung aller Einzelpotentiale .
Dieser Ansatz ähnelt sehr den Überlegungen der Parzen-Window Methode, umso mehr, als daß an Stelle der physikalischen Potentialfunktionen auch andere geignete Funktionen Verwendung finden können. Auch die Probleme, geeignete Parameter h der Funktionen zu finden (s. Abb. 2.2), sind dabei die gleichen.

Kapitel 6 Diskussion des Algorithmus

Für die ausführliche Diskussion und Simulation eines Lernvorgangs habe ich mir die Straffunktion

$$F_i(\underline{x},\underline{c}) = \tfrac{1}{2}(\underline{x}-\underline{c}_i)^2$$

ausgewählt. Um dabei das kleinste Risiko $R(\underline{c})$ zu erreichen, soll nach Kap. 1.5

$$\nabla_c R(\underline{c}) = \sum_i \int_{\Omega_i} \nabla_c F_i(\underline{x},\underline{c}_i)\, p(\underline{x})\, d\underline{x} \stackrel{!}{=} 0$$

sein. Mit

$$\nabla_c F_i(\underline{x},\underline{c}) := f_i(\underline{x},\underline{c})$$

lautet die Bedingung

$$\sum_i E(f_i(\underline{x},\underline{c})) \stackrel{!}{=} 0$$

Eine Möglichkeit dies zu erreichen, besteht darin, in jeder Klasse durch Variation des $\underline{c}$

$$E(f_i(\underline{x},\underline{c}_i)) \stackrel{!}{=} 0$$

Ziel des Lernvorgangs werden zu lassen.

Für dieses Ziel, die Nullstelle einer stochastischen Funktion $f(\underline{x},\underline{c})$ zu finden, haben Robbins und Monro (s. Kap. 4.2ff) einen Algorithmus angegeben. Mit dem Gradient der Straffunktion ist

$$\underline{c}_i(n+1) = \underline{c}_i(n) - \gamma_n(\underline{c}_i(n) - \underline{x}(n))$$

$$\underline{c}_j(n+1) = \underline{c}_j(n) \quad \forall j \neq i \quad \text{wobei} \quad F_i(\underline{x},\underline{c}) = \min_k \{F_k(\underline{x},\underline{c})\}$$

Für γ_n gilt dabei

$$\lim_{n\to\infty} \gamma_n = 0 \quad , \quad \sum_{i=1}^{\infty} \gamma_i > \infty \quad , \quad \sum_{i=1}^{\infty} \gamma_i^2 < \infty$$

Um für die Simulation einen einfachen Algorithmus zu erreichen, nehme ich γ_n als Skalar an. Ein Beispiel dafür ist

$$\gamma_n = \frac{\alpha}{n^\beta}$$

Welche Bedingungen gelten dabei für α und β ?

Die Reihe $\sum_{n=1}^{\infty} \frac{1}{n^a}$ { ist divergent für $a \leq 1$; und konvergent für $a > 1$ } s. [7]

Daher sind die Reihen

$$\alpha \cdot \sum_{n=1}^{\infty} \frac{1}{n^{\beta}} > \infty \quad \text{bei} \quad \beta \leq 1$$

und

$$\alpha^2 \cdot \sum_{n=1}^{\infty} \frac{1}{n^{2\beta}} < \infty \quad \text{bei} \quad 2\beta > 1$$

Also ist $\frac{1}{2} < \beta \leq 1$ und $\alpha \in \mathbb{R}$.
Welchen Wert sollten nun α und β für "günstigste Konvergenz" haben?
Der Begriff "günstigste Konvergenz" soll bedeuten, daß der Abstand der für jede Klasse i iterierten $\underline{c}_i(n+1)$ zu den Lernzielen $\underline{c}_i^*$ mit Hilfe der γ_n so klein wie möglich gemacht wird. Sicherlich ist dies nicht für alle möglichen $\underline{c}_i(n+1)$ durch ein bestimmtes γ_n der Fall. Um dieses näher zu untersuchen, muß vorher die Frage geklärt werden: Was ist $\underline{c}_i^*$?

6.1 Das Konvergenzziel des Algorithmus

Der Algorithmus von Robbins und Monro garantiert die Konvergenz von $\underline{c}(n+1)$ zu $\underline{c}^*$, so daß

$$\lim_{n \to \infty} P\left(E(\underline{c}(n+1) - \underline{c}^*) = 0\right) = 1$$

Da

$$E\left(f_i(\underline{x}, \underline{c}^*)\right) = 0$$

ist

$$\int_{\Omega_i} f_i(\underline{x}, \underline{c}_i^*)\, p(\underline{x})\, d\underline{x} = \int_{\Omega_i} (\underline{x} - \underline{c}_i^*)\, p(\underline{x})\, d\underline{x} = 0$$

und

$$\underline{c}_i^* = \frac{\int_{\Omega_i} \underline{x}\, p(\underline{x})\, d\underline{x}}{\int_{\Omega_i} p(\underline{x})\, d\underline{x}} = E(\underline{x}/\omega_i)$$

Das Konvergenzziel der quadratischen Straffunktion ist also der Erwartungswert der $\underline{x}$ in der betreffenden Klasse.
Diese Größe $\underline{c}_i^*$ sagt dabei aber nichts über das vorliegende $p(\underline{x}/\omega_i)$ aus. Die verschiedenen Verteilungen von Abb. 3.4 haben den gleichen Erwartungswert, also auch das gleiche $\underline{c}^*$.

Anders hingegen liegt der Fall, wenn wir eine andere Straffunktion nehmen würden, z.B.

$$\tilde{F}_i(\underline{x},\underline{c}) = 1 - \frac{1}{\sqrt{G_n}} \cdot e^{-\frac{(\underline{x}-\underline{c}_i)^2}{2G_n}}$$

Das erste Glied ihrer Taylorentwicklung ($G_n := 1$) ist gerade obige quadratische Straffunktion

$$F_i(\underline{x},\underline{c}) = \tfrac{1}{2}(\underline{x}-\underline{c}_i)^2$$

Der Grenzwert von $\tilde{F}_i(\underline{x},\underline{c})$ mit $\lim_{n\to\infty} G_n = 0$ ist eine δ-Funktion:

$$\lim_{n\to\infty} \tilde{F}_i(\underline{x},\underline{c}) \simeq \lim_{n\to\infty} 1 - \frac{1}{\sqrt{\pi G_n}} \cdot e^{-\frac{(\underline{x}-\underline{c}_i)^2}{2G_n}} = 1 - \delta(\underline{x}-\underline{c}_i)$$

Damit ist

$$R_i(\underline{c}) = \int_{\Omega_i} \tilde{F}_i(\underline{x},\underline{c})\, p(\underline{x})\, d\underline{x} = \int_{\Omega_i} (1 - \delta(\underline{x}-\underline{c}_i))\, p(\underline{x})\, d\underline{x}$$

$$= P(\omega_i) - p(\underline{c}_i)$$

Das Minimum von $R_i(\underline{c})$ wird bei $p(\underline{c}_i) = \max$, also bei dem Maximum von $p(\underline{x})$ erreicht. Bei den verschiedenen Verteilungen von Abb. 3.4 liegt dies aber sehr unterschiedlich.

6.2 Das optimale γ_n

Zurück nun zur Frage nach dem optimalen γ_n bei $F_i(\underline{x},\underline{c}) = \frac{1}{2}(\underline{x}-\underline{c}_i)^2$.

Sei $\underline{x}(n) \in \Omega_i$

Behauptung:

Wenn $\gamma_n := \frac{1}{n}$ gewählt wird, so ist für beliebiges $n \in \mathbb{N}$ der Erwartungswert E_i immer

$$E_i(\underline{c}_i(n+1)) = E_i(\underline{x}/\omega_i) = \underline{c}_i^*$$

Die Wahl von $\alpha = 1$ und $\beta = 1$ bedeutet, das bei obiger Voraussetzung ($\underline{x} \in \Omega_i$) für beliebige n das erwartete $\underline{c}_i(n+1)$ auch gleichzeitig das Konvergenzziel darstellt.

Beweis:

Beweis durch Induktion:

i) Induktionsanfang:

$$\underline{c}_i(1)=\underline{c}_i(o) + (\underline{x}(1)-\underline{c}_i(o))= \underline{x}(1)$$

$$E_i(\underline{c}_i(1))= E_i(\underline{x})=\underline{c}_i^*$$

ii) Induktionsdurchführung:

Sei $\underline{c}_i(n)=\frac{1}{n}\sum_{j=1}^{n} \underline{x}(j)$ .Dann ist

$$\underline{c}_i(n+1) = \underline{c}_i(n) + \frac{1}{n+1}\cdot\left(\underline{x}(n+1) - \underline{c}_i(n)\right)$$

$$= \underline{c}_i(n)\cdot\left(1-\frac{1}{n+1}\right) + \frac{\underline{x}(n+1)}{n+1} \qquad 1-\frac{1}{n+1} = \frac{n}{n+1}$$

$$= \frac{n}{n+1}\frac{1}{n}\sum_{j=1}^{n} \underline{x}(j) + \frac{\underline{x}(n+1)}{n+1} = \frac{1}{n+1}\sum_{j=1}^{n+1}\underline{x}(j)$$

Also ist

$$E_i(\underline{c}_i(n+1))= E_i\left(\frac{1}{n+1}\sum_{j=1}^{n+1}\underline{x}(j)\right) = \frac{1}{n+1}\sum_{j=1}^{n+1} E_i(\underline{x}(j)) = \frac{n+1}{n+1}\cdot E_i(\underline{x}) = \underline{c}_i^*$$

Die Wahl von $\alpha=1$ und $\beta=1$ bedeutet zwar für den Erwartungswert von $\underline{c}_i(n)$ "günstigste Konvergenz",nicht aber unbedingt auch für das statistisch abweichende $\underline{c}_i(n)$.Dazu tritt noch folgende Schwierigkeit auf:

Am Anfang der Iteration wird die Klassengrenze h_{ij} häufig derart verschoben,daß viele Muster $\underline{x}$,für die ω_i entschieden wurde und die zur Iteration von $\underline{c}_i$ beigetragen haben, sich nach der Verschiebung im Gebiet der Klasse j befinden. Dann aber sind nicht mehr alle Muster $\underline{x}$,die in $\underline{c}_i(n)=\frac{1}{n}\sum_{k=1}^{n}\underline{x}(k)$ eingingen,aus der selben Klasse i;die Voraussetzung des obigen Beweises ist nicht mehr erfüllt.Erst bei großen $n(n\to\infty)$ und $\gamma_n \to 0$ wird die Verschiebung der Grenzen so gering,daß wieder $E_i(\underline{x}) \approx E_i(\underline{x}/\underline{x}\in\Omega_i)=\underline{c}_i^*$ gilt.

Ein wirklich optimales γ_n für jedes stochastisch iterierte $\underline{c}_i(n)$ zu finden, ist recht schwer. (s. [11] 3.11)
Da sich $p(\underline{x}/\omega_i)$ bei der Iteration laufend ändert, müßte sich γ_n ebenfalls als stochastische Variable mit einem stochastischen Algorithmus ergeben, der wiederum einen Parameter enthielte, der zu optimieren wäre.
In dieser Arbeit kommt es nur auf die Tatsache der Konvergenz an, nicht aber auf die Konvergenzgeschwindigkeit.
Deshalb möchte ich auf umfangreiche und komplizierte iterative Bestimmung der γ_n verzichten (s. [10] 2.19) und für die folgenden Untersuchungen $\gamma_n = \frac{\alpha}{n^\beta}$ wählen. Die Koeffizienten α und β möchte ich dabei so korrigieren, daß der Algorithmus für die in den weiteren Abschnitten gebrauchten Wahrscheinlichkeitsverteilungen "befriedigend" konvergiert.

6.3 optimale Koeffizienten α und β

Um die Koeffizienten α und β auszuwählen, wurde der stochastische Algorithmus für verschiedene $p(x)$ bei $\dim(x)=1$, $\dim(c_i)=1$ und zwei Klassen simuliert. Die Klassengrenze $d(n)$ ist im eindimensionalen Fall ein Skalar. Aus je 20 Iterationen wurde $d(n)$ gemittelt und als Funktion von der Nummer n des Iterationsschritts aufgetragen.
Abb. 6.3 a zeigt Iterationen mit unterschiedlichen Schrittweiten der Iteration bei gleichem n. Sowohl zu große Schrittweiten bei kleinem β und großen α ($\alpha=1, \beta=.5$), als auch zu kleine Schrittweiten bei großem β und kleinem α ($\alpha=.1, \beta=1$) bringen schlechtes Konvergenzverhalten.

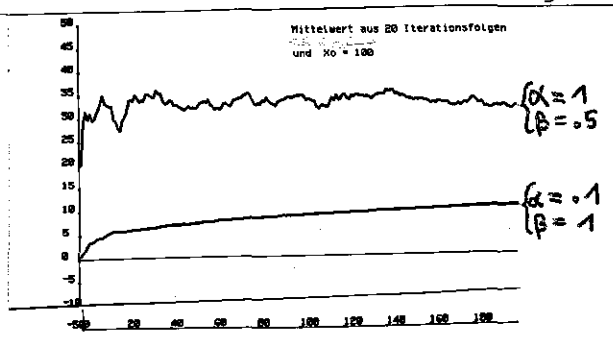


Abb. 6.3 a

Es ist also günstig, bei großem α auch große "Dämpfung" β zu verwenden.

Wenn für solche x^+, die dicht an der Klassengrenze d(n) liegen, $p(x^+) \approx 0$ ist, so tritt der besprochene Effekt der falschen Einordung der x auf Grund der häufig wechselnden Lage der Klassengrenze kaum auf und $\alpha=1, \beta=1$ sind wieder die optimalen Koeffizienten. Abb.6.3 b zeigt, daß aber auch die Koefizienten $\alpha=\beta=0,6$ und $\alpha=\beta=0,8$ keine schlechte Wahl sind.

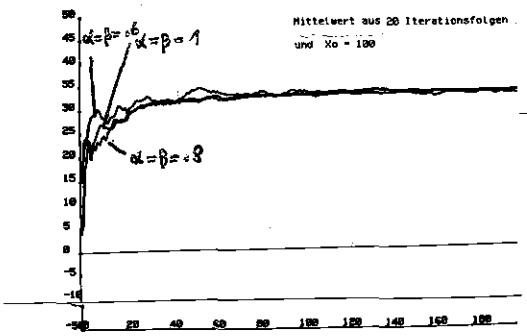


Abb. 6.3 b

Erst wenn die Wahrscheinlichkeitsdichte an den Klassengrenzen merklich >0 ist, zeigen sich Konvergenzverlaufsunterschiede. Im Fall von drei dicht beieinander liegenden Normalverteilungen $p(x) = A \cdot N(-z,1) + B \cdot N(0,1) + C \cdot N(z,1)$ mit $z=1,5$ zeigt Abb.6.3 c die Mittlungen aus je 20 Iterationsverläufen. Der im Fall einer Klasse am günstigsten konvergierende Algorithmus mit $\alpha=1, \beta=1$ ist hier nicht mehr der optimale.

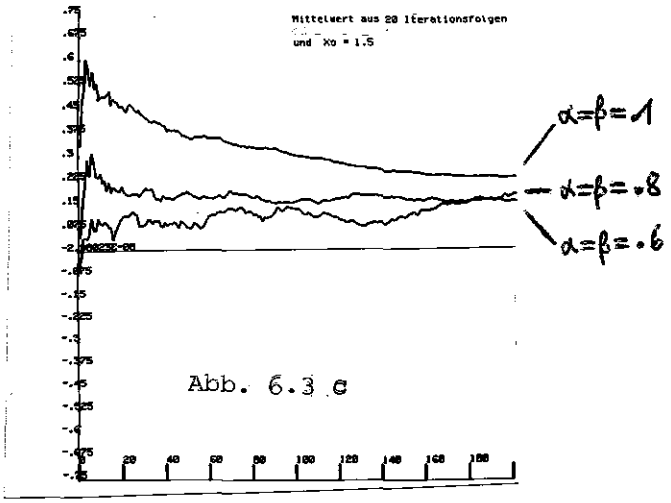
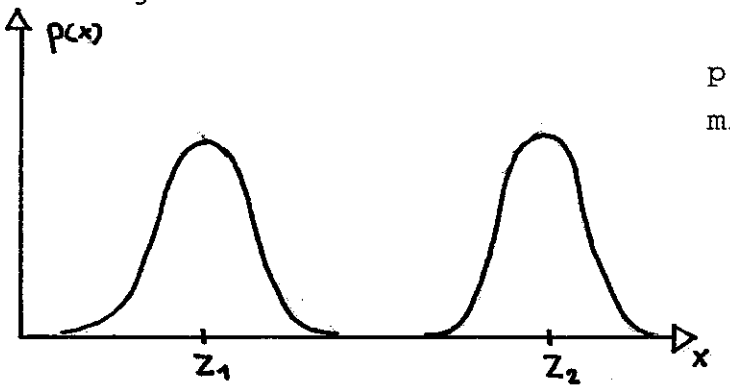


Abb. 6.3 c

Für alle weiteren Simulationen wird deshalb $\alpha = 0,8$ und $\beta=0,8$ gewählt.

6.4 Startwerte und Konvergenz

Um die verschiedenen Probleme des Algorithmus klarer darzustellen, werde ich in den folgenden Abschnitten stets den Fall von eindimensionalem $\underline{x}$ (dim $\underline{x}$=1) und zweier Klassen ω_1 und ω_2 behandeln.
Angenommen, p(x) sei eine Überlagerung aus zwei Normalverteilungen mit σ =1:



$$p(x) = \frac{1}{2}\left(N(z_1,1)+N(z_2,1)\right)$$

mit

$$N(z_1,1) = \frac{1}{\sqrt{2\pi}} \cdot e^{-\frac{(x-z_1)^2}{2}}$$

$$N(z_2,1) = \frac{1}{\sqrt{2\pi}} \cdot e^{-\frac{(x-z_2)^2}{2}}$$

Definition: Die Komponenten des Vektors der eindimensionalen Parameter für n=0 $\underline{C}(n) = (c_1(n), c_2(n))\big|_{n=0}$ heißen *Startwerte* der Iteration.

Definition: Die Erwartungswerte $z_1,\ldots,z_m$ von Einzelverteilungen, deren Überlagerung p(x) darstellt, heißen *Zentren* von p(x).

Im vorliegenden Fall von zwei Normalverteilungen sind z_1, z_2 die Erwartungswerte von $N(z_1,1)$ und $N(z_2,1)$.
Die Grenze d(n) zwischen den beiden Klassen ist mit

$$h_{12}(d) = F_1(d,\underline{c}) - F_2(d,\underline{c}) = 0$$

$$\frac{1}{2}(d-c_1)^2 - \frac{1}{2}(d-c_2)^2 = 0$$

oder

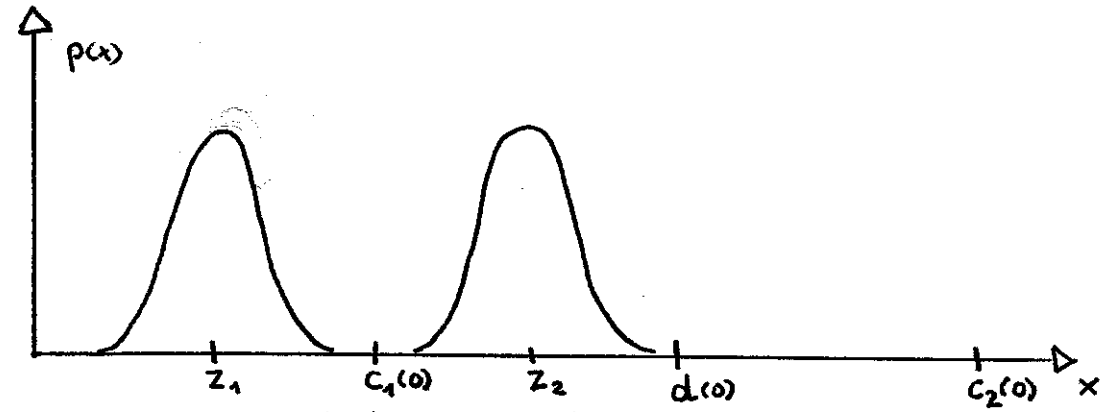
$$d(2c_2 - 2c_1) = c_2^2 - c_1^2$$

$$d = \frac{c_2^2-c_1^2}{2(c_2-c_1)} = \frac{(c_2+c_1)(c_2-c_1)}{2(c_2-c_1)}$$

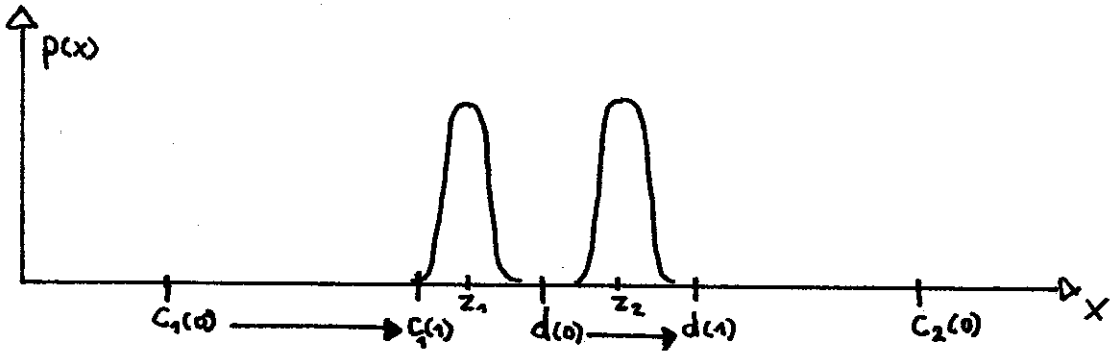
$$\boxed{d(n) = \frac{1}{2}\left(c_1(n)+c_2(n)\right)}$$

gegeben.

Die Startwerte $c_1(0)$ und $c_2(0)$ bestimmen die Klassengrenze $d(0)$, von deren Lage die Korrektur von $c_1(0)$ und $c_2(0)$ bei der ersten Iteration sehr stark abhängt:
Angenommen, $c_1(0)$ und $c_2(0)$ liegen sehr unterschiedlich von den Zentren entfernt.



Dann ist mit $p(x/\omega_2) \approx 0$ auch $P(\omega_2) \approx 0$, so daß immer $c_2(n+1) = c_2(0)$ sein wird. $c_1(n+1)$ konvergiert dann zu $E(p(x/\omega_1)) \approx E(p(x))$ mit $P(\omega_1) \approx 1$. Zweifellos ist damit aber das Ziel der Mustererkennung, die Trennung der beiden Cluster um z_1 und z_2, nicht erreicht.
Der gleiche Fall tritt auf, wenn $c_1(0)$ und $c_2(0)$ zwar gleich weit von $E(p(x))$ entfernt sind, aber der Gesamtabstand zu groß ist.



Schon die erste Iteration bewirkt eine derartige Verschiebung von $d(0)$, daß wieder obiger Fall eintritt.

Für das Systemverhalten bei den Startwerten $c_1(o)$ und $c_2(o)$ sind also zwei Größen wichtig:
Die Lage der Klassengrenze $d(o)$ relativ zu $E(p(x))$ (Symmetrie der Startwerte) und der absolute Abstand der Startwerte von $E(p(x))$.

Kapitel 7 Bifurkation des Algorithmus

Im vorigen Abschnitt wurde das Konvergenzverhalten des Algorithmus in Abhängigkeit von den Startwerten $c_1(0), c_2(0)$ untersucht.

Im folgenden Abschnitt möchte ich der Frage nachgehen: Kann sich das Konvergenzverhalten sprunghaft verändern, wenn sich ein Parameter von p(x) *(Kontrollparameter)* kontinuierlich ändert?

Dieses Verhalten läßt sich bei folgendem einfachen Beispiel beobachten:

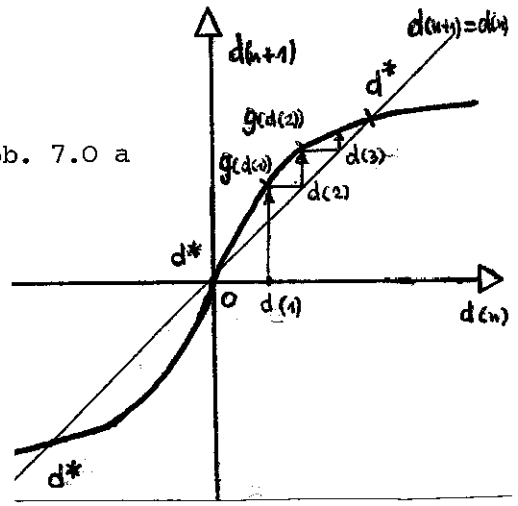


Abb. 7.0 a

Die Grafik zeigt die Abbildung
$d(n+1)=g(d(n))$
für die Schritte n=1,2,3,...
bei einem System mit einem labilen ($d^*=0$) und zwei stabilen Fixpunkten.
Geometrisch bedeutet die Fixpunktbedingung, daß die Funktion $g(d(n))$ einen Schnittpunkt mit der Geraden $d(n+1)=d(n)$ hat.

Sei für ein System
$d(n+1) := d(n)+f(d(n)) := g(d(n))$
eine Abbildungsvorschrift.
Im kontinuierlichen Fall ist die Änderung von d(n)

$$\dot{d}(t) = f(d(t)) := -d(t)\cdot(d^2(t)-m)$$

Sei m = constant.
Der Fixpunkt d^* des Systems für $t \to \infty$ ist mit der Fixpunktbedingung

$$\dot{d}(t) \stackrel{!}{=} 0$$

$$-d^*\cdot(d^{*2}-m) = 0$$

Dabei sind drei Fälle zu unterscheiden:

1) $m < 0$

Da $d^{*2}-m > 0$ ist die Fixpunktbedingung nur für d(t) =0 erfüllt.
Da der Zuwachs für d(t) dabei abnimmt

$$\ddot{d}(t)\Big|_{d^*=0} = m < 0$$

ist $d^* = 0$ auch ein stabiler Fixpunkt.

2) $m > 0$

Die Fixpunktbedingung ist erfüllt, wenn sowohl $d^* = 0$ als auch $d^{*2} - m = 0$. Also existieren drei Lösungen:

$$d_1^* = 0 \qquad d_2^* = +\sqrt{m} \qquad d_3^* = -\sqrt{m}$$

Da

$$\ddot{d}(t) = m - 2\,d(t)^2$$

ist $\ddot{d}(t)\big|_{d(t)=0} > 0$ und damit $d^*=0$ ein labiler Fixpunkt

und $\ddot{d}(t)\big|_{d(t)=\pm\sqrt{m}} < 0 \Rightarrow$ $d^* = \pm\sqrt{m}$ stabile Fixpunkte.

3) $m=0$

Aus Stetigkeitsgründen ist $d^*=0$ der einzige stabile Fixpunkt.

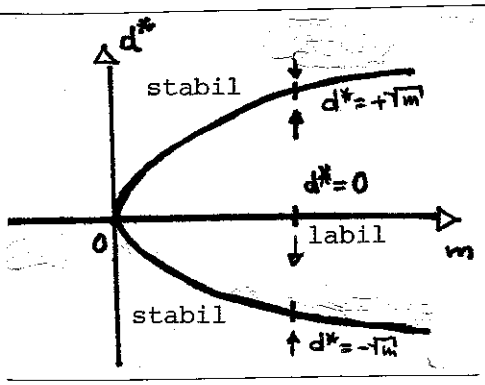


Abb. 7.0 b

Für das Verhalten des Systems ergibt sich damit (siehe nebenstehende Abbildung) :

Bei $m \leq 0$ ist $d^*=0$ einziger stabiler Fixpunkt einer Abbildung, die bei beliebigen $d \in \mathbb{R}$ den Abstand $|d-d^*|$ verkleinert.
Bei $m > 0$ ist diese Lösung nicht mehr stabil, es ergeben sich zwei weitere zusätzliche, stabile Lösungen
Das "Verzweigen" der Lösungsmenge der Fixpunktgleichung beim Nullpunkt m=0 wird als *Bifurkation* bezeichnet.
Es stellt sich nun die Frage, ob man ähnliches Verhalten auch bei der Abbildung des stochastischen Algorithmus beobachten kann.
Die Abbildung der Klassengrenze ist

$$d(n+1) = \tfrac{1}{2}\left(c_1(n+1, d(n)) + c_2(n+1, d(n))\right) := \chi(d_n)$$

Die Fixpunkte der Parameter sind

$$c_1^* = E(x/\omega_1) \qquad d^* := \tfrac{1}{2}(c_1^* + c_2^*)$$
$$c_2^* = E(x/\omega_2)$$

Daher ist

$$d^* = \frac{1}{2}\left(E(x/\omega_1) + E(x/\omega_2)\right), \quad \text{wobei} \quad \begin{matrix} x \in \Omega_1 \,\&\, x < d^* \\ x \in \Omega_2 \,\&\, x > d^* \end{matrix}$$

Im endlichen Fall $n < \infty$ gilt für den Erwartungswert der Klassengrenze

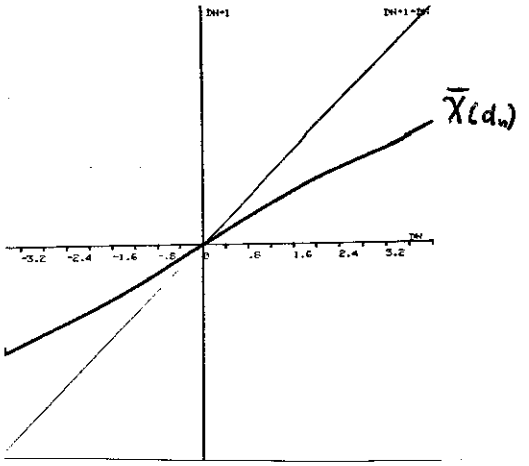
$$d_{(n+1)} = \frac{1}{2}\left(E(x/x<d(n)) + E(x/x>d(n))\right) = \overline{\chi}(d_n)$$

Die Fixpunktbedingung lautet hier

$$d_{(n+1)} = d_{(n)} \qquad d_n := d_{(n)}$$

Um eine allgemeine Aussage darüber zu machen, wann die Fixpunktbedingung erfüllt sein kann, muß ich $\overline{\chi}(d_n)$ näher untersuchen.

Für die Funktion $\overline{\chi}(d_n)$ gilt:



Bei einer symmetrischen Verteilungsfunktion mit $p(x)=p(-x)$ ist

1) $\lim_{n\to\infty} \overline{\chi}(d_n) - \frac{d_n}{2} = 0$

Die Gerade $d(n+1)=d(n)/2$ ist Asymptote.

2) $\overline{\chi}(0) = 0$

Beweis:

1) Es ist $$\lim_{d_n\to\infty} \overline{\chi}(d_n) = \frac{1}{2}\left[\lim_{d_n\to\infty} \frac{\int_{-\infty}^{d_n} x\,p(x)\,dx}{\int_{-\infty}^{d_n} p(x)\,dx} + \lim_{d_n\to\infty} \frac{\int_{d_n}^{\infty} x\,p(x)\,dx}{\int_{d_n}^{\infty} p(x)\,dx}\right]$$

Da $\lim_{d_n\to\infty} \int_{-\infty}^{d_n} p(x)\,dx = 1$ und $\lim_{d_n\to\infty} \int_{-\infty}^{d_n} x\,p(x)\,dx = 0$ (da $p(x)=p(-x)$)

ist

$$\lim_{n\to\infty} \overline{\chi} - \frac{d_n}{2} = \lim_{n\to\infty} \frac{1}{2} \frac{\int_{d_n}^{\infty} x\,p(x)\,dx}{\int_{d_n}^{\infty} p(x)\,dx} - \frac{d_n}{2} \overset{\text{l'Hopital}}{=} \frac{1}{2} \lim_{d_n\to\infty} \lim_{M\to\infty} \frac{\frac{\partial}{\partial d_n}\int_{d_n}^{M} x\,p(x)\,dx}{\frac{\partial}{\partial d_n}\int_{d_n}^{M} p(x)\,dx} - \frac{d_n}{2}$$

$$= \lim_{M\to\infty}\left(\lim_{d_n\to\infty} \frac{1}{2} \frac{d_n \cdot p(d_n)}{p(d_n)}\right) - \frac{d_n}{2} = \frac{d_n}{2} - \frac{d_n}{2} = 0$$

2) Da $p(x)=p(-x)$ ist $E(x/x<0)=-E(x/x>0)$.

Also ist auch $\overline{\chi}(0)=\frac{1}{2}\left(E(x/x<0)+E(x/x>0)\right)=0$

Aus den Feststellungen 1) und 2) folgt für den Verlauf von $\overline{\chi}(d_n)$, daß $\overline{\chi}(d_n)$ durch den Nullpunkt geht und sich asymptotisch an $d_n/2$ annähert.
Für $d_n=0$ ist auch $\overline{\chi}(d_n)=0$ und damit Fixpunkt der Abbildung.
Weitere Fixpunkte existieren nur dann, wenn $\overline{\chi}(d_n)$ die Fixpunktbedingung $\overline{\chi}(d_n)=d_n$ erfüllt, also zwischen dem Nullpunkt und dem Annähern an die Gerade mit $d_{n+1}=d_n/2$ vorher die Gerade $d_{n+1}=d_n$ schneidet.
Dies ist zweifelsohne dann der Fall, wenn $\overline{\chi}(d_n)$ in $d_n=0$ eine größere Steigung hat als die Fixpunkt-Gerade $d_{n+1}=d_n$.
Damit existieren mindestens drei Fixpunkte, wenn

$$\frac{\partial}{\partial d_n}\overline{\chi}(d_n)>1$$

gilt. Für eine Verteilung mit $p(x)=p(-x)$ bedeutet dies

$$\boxed{\frac{\partial}{\partial d_n}\overline{\chi}(d_n)=2\cdot p(0)\cdot E(x/x>0)>1} \qquad \text{Formel (7.0)}$$

Beweis:

Es ist $\overline{\chi}(d_n)=\frac{1}{2}\left[\frac{\int_{d_n}^{\infty} x\,p(x)\,dx}{\int_{d_n}^{\infty} p(x)\,dx}+\frac{\int_{-\infty}^{d_n} x\,p(x)\,dx}{\int_{-\infty}^{d_n} p(x)\,dx}\right]$

und damit $\frac{\partial}{\partial d_n}\overline{\chi}(d_n)=\frac{1}{2}\left[\frac{\partial}{\partial d_n}\frac{\int_{d_n}^{\infty} x\,p(x)\,dx}{\int_{d_n}^{\infty} p(x)\,dx}+\frac{\partial}{\partial d_n}\frac{\int_{-\infty}^{d_n} x\,p(x)\,dx}{\int_{-\infty}^{d_n} p(x)\,dx}\right]$

Mit der Formel $\left(\frac{a}{b}\right)'=\frac{a'b-ab'}{b^2}$

ist

$$\frac{\partial}{\partial d_n}\overline{\chi}(d_n)=\frac{1}{2}\left[\frac{-d_n\cdot p(d_n)\cdot\int_{d_n}^{\infty} p(x)\,dx+\int_{d_n}^{\infty} x\,p(x)\,dx\cdot p(d_n)}{\left(\int_{d_n}^{\infty} p(x)\,dx\right)^2}\right.$$

$$\left.+\frac{-d_n\cdot p(d_n)\cdot\int_{d_n}^{-\infty} p(x)\,dx+\int_{d_n}^{-\infty} x\,p(x)\,dx\cdot p(d_n)}{\left(\int_{d_n}^{\infty} p(x)\,dx\right)^2}\right]$$

Mit $$\int_0^{\infty} p(x)\,dx = \int_{-\infty}^{0} p(x) = \frac{1}{2}$$

und $$\int_0^{-\infty} x\,p(x)\,dx = -\int_0^{-\infty} -x\,p(-x)\,dx \overset{y=-x}{=} \int_0^{\infty} y\,p(y)\,dy$$

ist

$$\frac{\partial}{\partial d_w}\overline{\chi(d_w)}\Big|_{d_w=0} = \frac{1}{2}\left[\frac{0 + p(0)\cdot\int_0^{\infty} x p(x)\,dx}{1/4} + \frac{p(0)\int_0^{\infty} y p(y)\,dy}{1/4}\right] = 2\cdot p(0)\cdot\frac{\int_0^{\infty} x p(x)\,dx}{\int_0^{\infty} p(x)\,dx}$$

$$\frac{\partial}{\partial d_w}\overline{\chi(0)} = 2.p(0).E(x/x>0) \quad \text{q.e.d.}$$

Da auch $E(|x|/x<0)=E(x/x>0)$ gilt[+], so ist ebenfalls

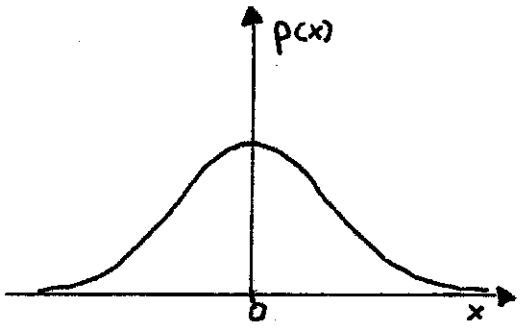
$$\frac{\partial}{\partial d_w}\overline{\chi(d_w)} = p(0).E(|x|) \qquad x\in\mathbb{R}$$

[+] $$\int_{-\infty}^{0} |x|\,p(x)\,dx \overset{Y=-X}{=} \int_0^{\infty} |Y|\,p(Y)\,dY$$

In den folgenden Abschnitten möchte ich verschiedene Verteilungen p(x) darauf untersuchen,ob die obige Ungleichung bei ihnen erfüllt sein kann.
Wenn diese Ungleichung erfüllt ist,bedeutet dies ,daß mindestens drei Fixpunkte der erwähnten Abbildung existieren müssen.Andererseits dagegen ist eine Nichterfüllung der Ungleichung keineswegs der Beweis dafür,daß eine Bifurkation nicht existieren kann.Es hängt bei jedem p(x) von der Form von $\overline{\chi(d_w)}$ ab,ob die Ungleichung das einzige Bifurkationskriterium darstellt oder nicht.

7.1 Untersuchung auf Bifurkation

7.1.1 Einfache Normalverteilung



Sei $p(x) = N(o,\sigma) = \frac{1}{\sqrt{2\pi\sigma}} \cdot e^{-\frac{x^2}{2\sigma}}$

Dann ist

$$p(o) = \frac{1}{\sqrt{2\pi\sigma}}$$

und

$$E(x/x>O) = \frac{1}{\sqrt{2\pi\sigma}} \cdot \underbrace{\int_0^\infty x \cdot e^{-\frac{x^2}{2\sigma}}\, dx}_{1/2} \overset{\text{Subst.: } y=\frac{x}{\sqrt{2\sigma}}}{=} \frac{2}{\sqrt{2\pi\sigma}} \cdot \int_0^\infty \sqrt{2\sigma} \cdot y \cdot e^{-y^2} dy \cdot \sqrt{2\sigma}$$

$$= \frac{4\sigma}{\sqrt{2\pi\sigma}} \cdot \frac{1}{2} e^{-y^2}\Big|_0^\infty = \frac{2\sigma}{\sqrt{2\pi\sigma}}$$

Also gilt für die Ungleichung

$$2 \cdot p(o) \cdot E(x/x>o) = \frac{1}{\sqrt{2\pi\sigma}} \cdot \frac{2\sigma}{\sqrt{2\pi\sigma}} \cdot 2 = \frac{2}{\pi} < 1$$

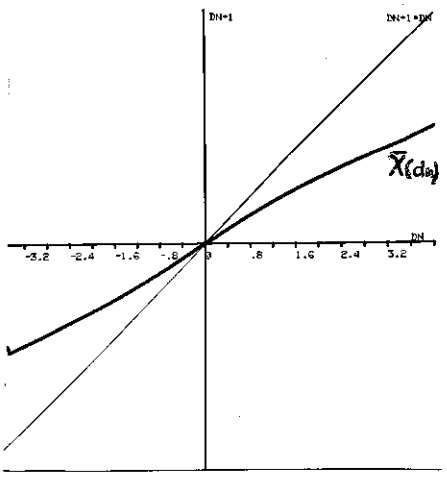


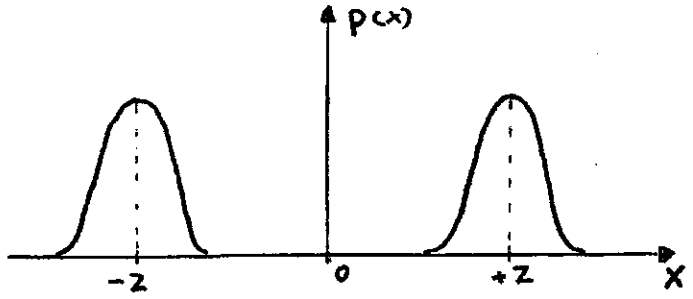
Abb. 7.1.1

Abb.7.1.1 zeigt die Funktion $\bar{X}(d_N)$ im Bereich der Normalverteilung. Die Nichterfüllung der Bedingungsgleichung bedeutet hier also auch, daß keine Bifurkation existiert.

7.1.2 Doppelte Normalverteilung

Sei $p(x)=\frac{1}{2}\left(N(-z,\sigma)+N(z,\sigma)\right)$

$$=\frac{1}{2}\cdot\frac{1}{\sqrt{2\pi\sigma}}\left(e^{\frac{-(x-z)^2}{2\sigma}}+e^{\frac{-(x+z)^2}{2\sigma}}\right)$$



Dann ist

$$p(o)=\frac{1}{\sqrt{2\pi\sigma}}\cdot e^{-\frac{z^2}{2\sigma}}$$

und für den Erwartungswert gilt mit Anhang A ,A= ½ ,B=O

$$E(x/x>o)=\frac{-\frac{1}{2}\left(\phi(-y)\cdot z-\phi(y)\cdot z-2\psi(y)\right)}{1/2}$$

$$=\left(2\phi(y)\cdot z+2\psi(y)-z\right)$$

$y:=\frac{z}{\sqrt{\sigma}}$

$\phi(-x)=1-\phi(x)$

$\psi(-x)=\psi(x)$

mit den Funktionen

$$\phi(a)=\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{a}e^{-t^2/2}\,dt$$ (Gaußsches Fehlerintegral)

und

$$\psi(a)=\sqrt{\frac{\sigma}{2\pi}}\cdot e^{-a^2/2}$$

Also gilt

$$m(z):=2p(o).E(x/x>o)=\frac{2}{\sqrt{2\pi\sigma}}\cdot e^{-\frac{z^2}{2\sigma}}\cdot\left(2\cdot\phi(y)\cdot z+2\cdot\psi(y)-z\right)$$

und für z=O

$$m(o)=\frac{2}{\sqrt{2\pi\sigma}}\cdot 2\cdot\sqrt{\frac{\sigma}{2\pi}}=\frac{2}{\pi}$$

Dieses Ergebnis war zu erwarten: Wenn beide Verteilungen "übereinander geschoben" werden,so ergibt sich eine einfache Normalverteilung und damit auch $\frac{\partial\overline{\chi}}{\partial d_m}(d_m)$ einer einfachen Normalverteilung.

Behauptung: Auch bei zwei überlagerten Normalverteilungen als p(x) ist die Ungleichung 7.0 niemals erfüllt.

Beweis:

Sei m(z) wie oben gegeben.

Da $\phi(\omega) \in [0;1]$ ist $2 \geq 2\phi(x) \quad \forall x \in \mathbb{R}$ $\qquad p(0) = \frac{1}{\sigma}\psi(y)$

$$\tilde{m}(z) =: \frac{2}{\sigma}\psi(y)\cdot\left(z\cdot(2\cdot\Phi(z)-1)+2\psi(y)\right) > m(z)$$

Die Funktion

$$\tilde{\tilde{m}}(z) = \frac{2}{\sigma}\psi(y)\cdot z + \frac{4}{\sigma}\psi^2(y) := T_1 + T_2$$

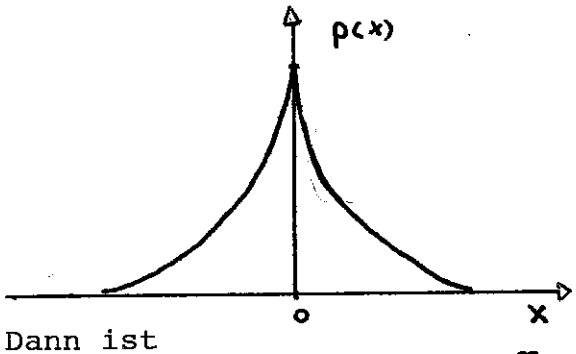
besteht aus zwei Termen. Der erste Term nimmt zwar mit z linear zu, aber mit $\exp(-z^2)$ noch stärker ab, also insgesamt monoton ab, ebenso wie der zweite Term.

Also gilt

$$m(z) \leqslant \tilde{m}(z) \leqslant \tilde{m}(0) = \frac{2}{\pi} < 1 \qquad \forall z \in \mathbb{R}^+$$

Dies läßt sich genauso auch für beliebiges σ zeigen.

7.1.3 Einfache Exponentialverteilung



Sei

$$p(x) = \frac{1}{2k} \cdot e^{-\left|\frac{x}{k}\right|}$$

also

$$p(o) = \frac{1}{2k} \qquad k > 0$$

Dann ist

$$E(x/x>0) = \frac{1}{2k} \cdot \underbrace{\int_0^\infty x \cdot e^{-\left|\frac{x}{k}\right|} dx}_{1/2} = \frac{1}{k} \int_0^\infty x \cdot e^{-\frac{x}{k}} dx$$

Partielle Integr.

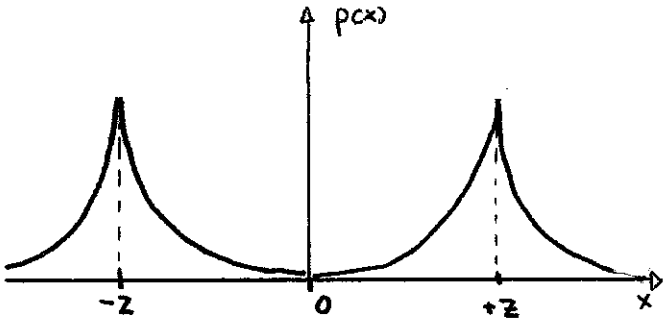
$$= \int_0^\infty e^{-\frac{x}{k}} dx = -k \cdot e^{-\frac{x}{k}} \Big|_0^\infty = k$$

Für die Ungleichung 7.0 lautet

$$2p(o).E(x/x>0) = 2 \cdot \frac{k}{2k} = 1 \text{ , nicht aber größer eins}$$

Also ist auch hier diese Bifurkationsvoraussetzung nicht gegeben.

7.1.4 Doppelte Exponentialverteilung



Es sei

$$p(x)=\frac{1}{4k}\left(e^{-\left|\frac{x-z}{k}\right|}+e^{-\left|\frac{x+z}{k}\right|}\right)$$

so ist

$$p(o)=\frac{1}{2k}\cdot e^{-\left|\frac{z}{k}\right|}$$

Um den Erwartungswert von x auszurechnen, muß ich das Integral so aufteilen, daß die Argumente der exp-Funktionen im Integrationsgebiet positiv bleiben und damit die Betragsstriche wegfallen:

$$E(x/x\gamma o)=\frac{1}{4k}\left[\int_0^z x\cdot e^{-\left|\frac{x-z}{k}\right|}dx+\int_z^\infty x\cdot e^{-\left|\frac{x-z}{k}\right|}dx+\int_0^\infty x\cdot e^{-\left|\frac{x+z}{k}\right|}dx\right]$$

$$=\frac{1}{4k}\left[\underbrace{\int_0^z x\cdot e^{+\frac{(x-z)}{k}}dx}_{T1}+\underbrace{\int_z^\infty x\cdot e^{-\frac{(x-z)}{k}}}_{T2}+\underbrace{\int_0^\infty x\cdot e^{-\frac{(x+z)}{k}}dx}_{T3}\right]$$

$$=: \quad T1 \quad + \quad T2 \quad + \quad T3$$

Mit der Beziehung

$$\int_{a_1}^{a_2}(A_1+A_2\cdot x)\cdot e^x dx = e^x\cdot[A_1+A_2(x-1)]\Big|_{a_1}^{a_2}$$

lauten dann die Einzelterme:

$$T1=\int_0^z x\cdot e^{\frac{x-z}{k}}dx \qquad y=\frac{x-z}{k}\quad dy=dx/k$$

$$=\int_{-z/k}^{0}(z\cdot k+k^2\cdot y)e^y dy = e^y\cdot(z\cdot k+k^2(y-1))\Big|_{-z/k}^{0} \quad \text{s.o.}$$

$$=(z\cdot k-k^2)-e^{-z/k}\cdot(z\cdot k+k^2(z/k+1)) \qquad =\boxed{k^2(z/k-1+e^{-z/k})}$$

$$T2=\int_z^\infty x\,e^{-\frac{x-z}{k}}dx \qquad y=-\frac{x-z}{k}\quad dy=-\frac{dx}{k}$$

$$=\int_0^{-\infty}(-kz+k^2y)e^y dy = e^y\cdot(-z\cdot k+k^2(y-1))\Big|_0^{-\infty}$$

$$=-[-z\cdot k-k^2] \qquad =\boxed{k^2(1+z/k)}$$

$$T3 = \int_0^\infty x \cdot e^{-\frac{x+2}{k}}\, dx \qquad y = -\frac{x+2}{k} \quad dy = -\frac{dx}{k} \qquad = \int_{-2/k}^{-\infty} (2k + k^2 y)\, e^y\, dy$$

$$= e^y \left(2k + k^2(y-1)\right)\Big|_{-2/k}^{-\infty} = -k^2 \cdot e^{-2/k}\left(2/k + (-2/k - 1)\right) = k^2 \cdot e^{-2/k}$$

Also lautet die Ungleichung

$$\frac{\partial}{\partial d_n}\overline{\chi}(d_n)_{d_n=0} = 2p(o).E(x/x>o) = 2\,\frac{1}{2k}\cdot e^{-2/k}.\frac{1}{4k}(T1+T2+T3)$$

$$= \frac{1}{4k^2}e^{-2/k}\cdot\left(k^2(2/k - 1 + e^{-2/k}) + k^2(1+2/k) + k^2 \cdot e^{-2/k}\right) \qquad 2/k := y$$

$$= \frac{1}{2}\left(e^{-y}\cdot e^{-y} + e^{-y}\cdot y\right) = \frac{1}{2}e^{-2y} + \frac{1}{2}e^{-y}\cdot y$$

Behauptung: Die Funktion

$$m(y) := \frac{1}{2}\left(e^{-2y} + e^{-y}\cdot y\right) = \frac{\partial}{\partial d_n}\overline{\chi}(d_n)\Big|_{d_n=0}$$

kann nie größer Eins werden.

Beweis:

Da $e^y > y \quad \forall y>0$ ist auch $e^y \cdot e^{-y} > y \cdot e^{-y}$

Damit ist

$$\frac{1}{2}\left(e^{-2y} + 1\right) := \tilde{m}(y) > m(y) = \frac{1}{2}\left(e^{-2y} + e^{-y}y\right)$$

Die Funktion $\tilde{m}(z)$ fällt mit exp(-2) monoton bei steigendem ,so daß gilt:

$$m(y) \leqslant \tilde{m}(y) \leqslant \tilde{m}(o) = 1 \qquad \forall y \in \mathbb{R}^+$$

Die Bifurkationsbedingung (7.0) ist bei dieser Verteilung ebenfalls nicht erfüllt.

7.2 Eine Verteilung mit Bifurkation

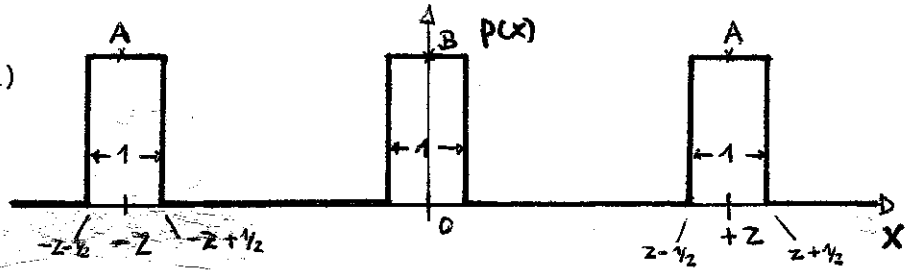
In den vorhergehenden Abschnitten wurden vier verschiedenen Wahrscheinlichkeitsverteilungen darauf untersucht,ob die Bifurkationsbedingung

$$\frac{\partial}{\partial d_w}\,\overline{\chi}(d_w)\Big|_{d_w=0} = 2\,p(o)\,E(x/x>o) > 1$$

bei ihnen erfüllt ist.Bei allen vier konnte nachgewiesen werden,daß dies nicht der Fall ist.

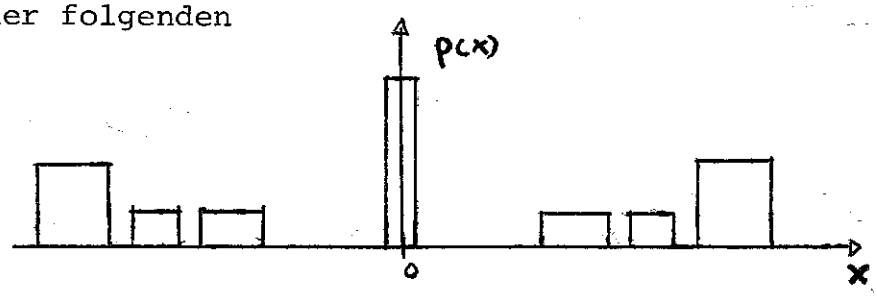
Im folgenden Abschnitt wird eine Verteilung angegeben, für die obenstehende Bifurkationsbedingung erfüllt sein kann.
Die Bifurkationsbedingung wird sicher dann erfüllt sein, wenn sowohl p(o) als auch p(x) bei großen x,und damit der Erwartungswert von x,besonders groß sein werden.
Dies ist z.B. bei folgender Verteilung der Fall.

a)



und der folgenden

b)



Die Verteilung a) läßt sich leicht durchrechnen:

$p(o) = B$ (Die Normierungsbedingung ist dabei $2A+B=1$) $\Rightarrow A = \frac{1-B}{2}$

$$E(x|x>0) = 2\cdot\left[\int_0^{z/2} x\cdot B\,dx + \int_{z-\frac{1}{2}}^{z+\frac{1}{2}} x\cdot A\,dx\right] = \left[x^2\Big|_0^{z/2}\cdot B + x^2\Big|_{z-\frac{1}{2}}^{z+\frac{1}{2}} A\right]$$

$$= \frac{B}{4} + A\left[(z+\tfrac{1}{2})^2 - (z-\tfrac{1}{2})^2\right] = \frac{B}{4} + \frac{(1-B)}{2}\cdot 2z = B\left(\tfrac{1}{4}-z\right) + z$$

$$2\cdot p(o)\cdot E(x|x>0) = 2B^2\left(\tfrac{1}{4}-z\right) + 2Bz$$

Wenn wir nun z.B. B=1/2 annehmen,so ist

$$2 \cdot p(0) \cdot E(x \mid x>0) = 2\left(\tfrac{1}{4}\left(\tfrac{1}{4}-2\right) + \tfrac{1}{2}z\right) = \tfrac{1}{2}\left(z+\tfrac{1}{4}\right)$$

und dieses größer eins,wenn

$$\tfrac{1}{2}\left(z+\tfrac{1}{4}\right) > 1 \quad \Leftrightarrow \quad z > 1{,}75$$

Die Bifurkationsbedingung ist also für alle $z > 1{,}75$ erfüllt! Daraus läßt sich schließen,daß mindestens drei Fixpunkte existieren müssen;wieviele es genau sind,ist aus der Bifurkationsungleichung nicht zu ersehen und müßte gesondert untersucht werden.

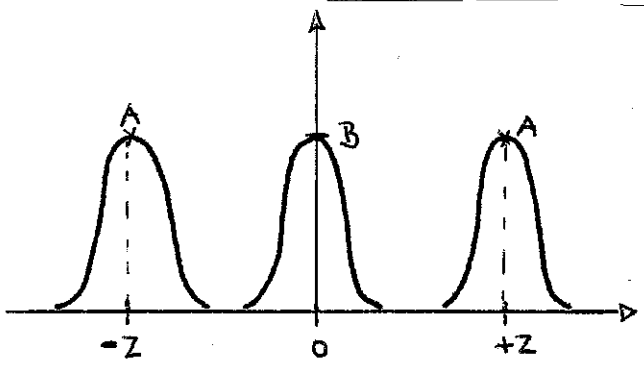
Wenn die in a) beschriebene Verteilung vorliegt, so verhält sich die stochastische Approximation als Abbildung so ähnlich wie die Abbildung mit der Differenzialgleichung

$$\dot{d}(t) = -d(t) \cdot \left(d^2(t) - m\right)$$

Bei einer kontinuierlichen Änderung des Parameters m (bei der stochastischen Iteration der Parameter $m(z,6) = \frac{2}{\partial dm}\overline{\chi}(0)$) ändert sich plötzlich das Ziel (Fixpunkt) der Abbildung, wenn ein bestimmter Wert überschritten wird;anstelle eines stabilen Fixpunkts tritt ein labiler,und,symmetrisch mit gleichem Abstand,zwei zusätzliche stabile Fixpunkte.
Dieses Verhalten des stochastischen Algorithmus soll in den folgenden Abschnitten rechnerisch und mit Computersimulation näher untersucht werden.
Um den Verhältnissen der Praxis näher zu kommen,wird an der Stelle der in a) beschriebenen Verteilung eine Verteilung benutzt,die durch Überlagerung von drei Normalverteilungen zustande kommt.

7.2.1 Bifurkationsbedingung bei drei Normalverteilungen

Es sei $p(x) = A.N(-z,\sigma)+B.N(o,\sigma)+A.N(z,\sigma)$



Dann ist die Verteilung symmetrisch und es gilt
$p(x)=p(-x)$

Die Konstanten A und B hängen durch die Normierungsbedingung zusammen:

$$\int_{-\infty}^{+\infty} A\cdot N(-z,\sigma)+B\cdot N(0,\sigma)+A\cdot N(z,\sigma)\,dx \stackrel{!}{=} 1 \qquad \Leftrightarrow A+B+A \stackrel{!}{=} 1$$
$$\Leftrightarrow A = \frac{1-B}{2}$$

Der Bifurkationsparameter $m(z,\sigma)$ ist mit

$$p(o) = \frac{1}{\sqrt{2\pi\sigma^2}}\left(2Ae^{-\frac{z^2}{2\sigma}}+B\right) = \frac{2}{\sigma}A\cdot\psi(y)+\frac{B}{\sqrt{2\pi\sigma^2}} \qquad y := \frac{z}{\sqrt{\sigma}}$$
$$\psi(-x)=\psi(x)$$
$$\Phi(-x)=1-\Phi(x)$$

und mit Anhang A

$$E(x/x>o) = -2A\left(\Phi(-y)\cdot z - z\cdot\Phi(y) - \psi(-y) - \psi(y)\right) + B\psi(o)$$
$$= +2A\left(2\Phi(y)\cdot z + 2\psi(y) - z\right) + B\psi(o)$$

also

$$m(z,\sigma) = 2\,p(o)\cdot E(x/x>o)$$
$$= 2\left(\frac{2}{\sigma}\cdot A\cdot\psi(y)+B\cdot\frac{1}{\sqrt{2\pi\sigma^2}}\right)\cdot\left(2A\left(2\Phi(y)\cdot z+2\psi(y)-z\right)+B\psi(o)\right)$$

mit den Funktionen

$$\Phi(y) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{y} e^{-t^2/2}\,dt \qquad \text{Gaußsche Fehlerfunktion}$$

und

$$\psi(y) = \frac{\sqrt{\sigma^2}}{\sqrt{2\pi}}\cdot e^{-y^2/2}$$

Als Polynom von B lautet $m(z,\sigma)$

$$m(z,\sigma) = 2\left(\left(1-B\right)\frac{1}{\sigma}\psi(y)+B\cdot\frac{1}{\sqrt{2\pi\sigma^2}}\right)\left((1-B)\left(2\Phi(y)\cdot z+2\psi(y)-z\right)+B\psi(o)\right)$$
$$= 2\left(B\left(\frac{1}{\sqrt{2\pi\sigma^2}}-\frac{1}{\sigma}\psi(y)\right)+\frac{1}{\sigma}\psi(y)\right)\left(B\left(\psi(o)-2\Phi(y)\cdot z-2\psi(y)+z\right)+\left(2\Phi(y)\cdot z+2\psi(y)-z\right)\right)$$
$$=: 2\left(B\cdot a_1+a_2\right)\left(B\cdot a_3+a_4\right)$$

$$m(B) = 2\left(B^2 \cdot a_1 a_3 + B\cdot(a_1 a_4 + a_2 a_3) + a_2 a_4\right)$$

$$= B^2 \cdot U_1 + B \cdot U_2 + U_3$$

mit den Konstanten

$$U_1 = 2\left(\frac{1}{\sqrt{2\pi}} - \frac{1}{c}\psi(y)\right)\left(\frac{\sqrt{6}}{\sqrt{2\pi}} - 2\phi(y)\cdot 2 - 2\psi(y) + 2\right)$$

$$U_2 = 2\left(\frac{1}{\sqrt{2\pi}} - \frac{1}{c}\psi(y)\right)\left(2\phi(y)2 + 2\psi(y) - 2\right) + \left(\frac{1}{c}\psi(y)\right)\cdot 2\cdot\left(\sqrt{\frac{6}{2\pi}} - 2\phi(y)\cdot 2 - 2\psi(y) + 2\right)$$

$$U_3 = \frac{1}{c}\psi(y)\left(2\phi(y)2 + 2\psi(y) - 2\right)$$

Dieses Polynom 2-ten Grades ist in Abb. 7.2 a dargestellt.

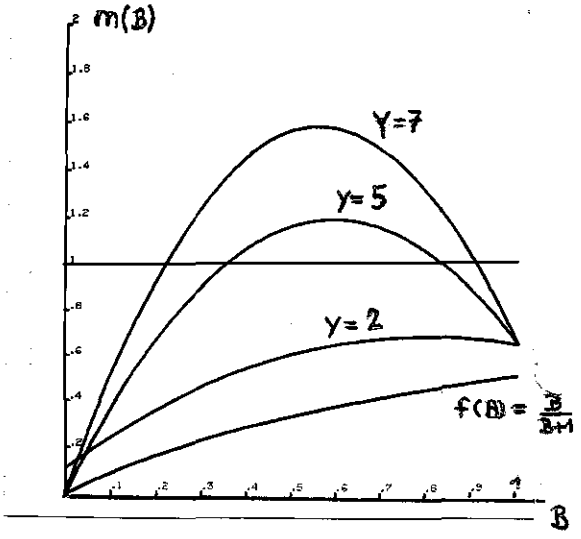


Abb. 7.2 a

Man erkennt, daß die Bifurkationsbedingung $m(B) > 1$ bei festem $y=\frac{z}{\sqrt{6}}$ nur für bestimmte B erfüllt ist.
Wenn y unterhalb eines bestimmten Wertes liegt, existiert kein B, das der Ungleichung $m(B) > 1$ genügt.

Das größte m(B), der Scheitelpunkt der Parabel, ist mit

$$\frac{\partial m(B)}{\partial B} = 2B\cdot U_1 + U_2 \stackrel{!}{=} 0$$

gegeben. Für $y \gg 3$ ist das B^+, für das $m(B^+)$ maximal wird, im limes mit

$$\lim_{y\to\infty} B^+ = \lim_{y\to\infty} \frac{-U_2}{2U_1} = \frac{-\left(2\frac{1}{\sqrt{2\pi}}\cdot(2+z)\right)}{2\left(2\frac{1}{\sqrt{2\pi}}\cdot\left(\sqrt{\frac{6}{2\pi}} - z\right)\right)}$$

$$\lim_{y\to\infty}\psi(y) = 0 \qquad \lim_{y\to\infty}\phi(y) = 1$$

$$= \lim_{y\to\infty} \frac{(z+2)}{2\left(z + \sqrt{\frac{6}{2\pi}}\right)} =$$

$$= \frac{1}{2}$$

Für "weit" ($y \gg 3$) auseinander liegende Zentren $-z, z$ ist die Bifurkationsbedingung somit (wenn überhaupt) bei $B = 1/2$ erfüllt.

Dies bedeutet aber nicht, daß für $B=1/2$ die stärkste Bifurkation existiert; wie wir später sehen werden, ist dies bei dem größten B der Fall, bei dem noch $m(B) > 1$ ist.

Die Funktion $m(Y)$ läßt sich für große y auch vereinfachen:

$$m(y) \underset{y \gg 3}{\approx} \lim_{y \to \infty} m(y) = \lim_{y \to \infty} B^2 \left(\frac{2}{\sqrt{12\pi\sigma}} \left(\sqrt{\frac{\sigma}{2\pi}} - 2z + z \right) \right) + B \left(\frac{2}{\sqrt{12\pi\sigma}} \cdot (2z - z) \right) + 0$$

$$= \lim_{y \to \infty} \frac{z}{\sqrt{\sigma}} \cdot \left(B \cdot \frac{2}{\sqrt{12\pi}} - B^2 \cdot \frac{2}{\sqrt{12\pi}} \right) + B^2 \cdot \frac{2}{\sqrt{12\pi\sigma}} \cdot \sqrt{\frac{\sigma}{2\pi}}$$

$$= \lim_{y \to \infty} y \cdot a + b$$

Bei großen y ist $m(y)$ also linear in y.

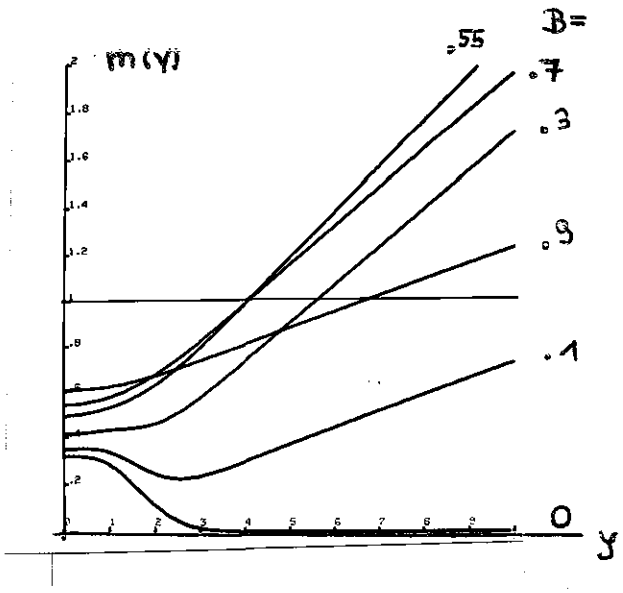


Abb. 7.2 b

In Abb. 7.2 b ist $m(y)$ für verschiedenen B aufgetragen.

Man sieht, wie für $B = 0$, den Fall zweier Normalverteilungen, die asymptotische Gerade mit der x-Achse zusammenfällt und damit nie $m(y) > 1$ werden kann.

Ebenso ist bei $B=1$, dem Fall nur einer Normalverteilung $N(0, \sigma)$,

$m(y)$ ein konstanter Wert < 1.

Für alle anderen Werte von B gibt es ein y^+, für das

$$m(y^+) > 1 \quad \forall y > y^+$$

gilt, also auch eine Bifurkation vorhanden ist.

Mit der folgenden Abbildung soll der Verlauf der Abbildung $d(n+1) = \overline{\chi}(d(n))$ näher erläutert werden.

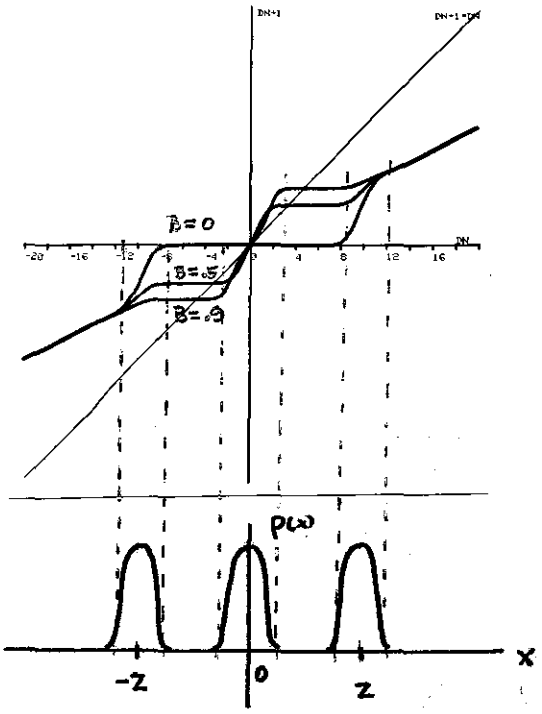


Abb. 7.2c

Bei konstanter Lage der Zentren (z=10) ist der Verlauf von $\overline{\chi}(d(n))$ für die Parameter B=0,.5,.9 gezeigt.
Wegen p(x)=p(-x) ist die Kurvenform symmetrisch.

Wie mit der darunter gezeichneten Verteilung p(x) erkennbar ist,ändern sich die Erwartungswerte der Klassen besonders,wenn die Klassengrenze d(n) in die Intervalle (-z-2,-z+2),(-2,2),(z-2,z+2) fällt,da dort jeweils mehr als 95% der normierten Normalverteilungen liegen.

In den Bereichen dazwischen ist d(n) fast konstant,da sich die Erwartungswerte mit $p(x) \approx 0$ fast nicht ändern.

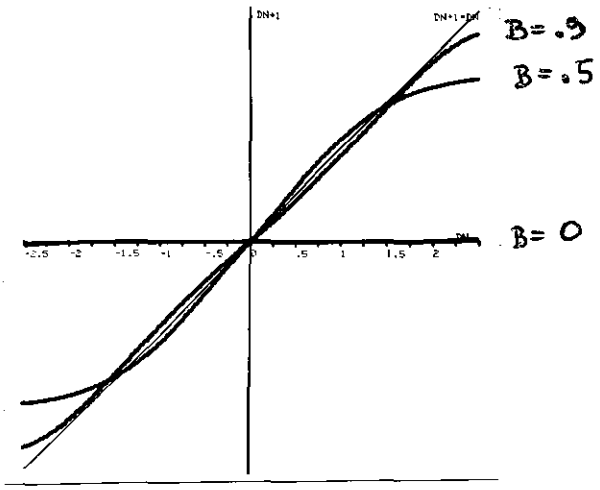


Abb. 7.2d

Den Übergang von einem Fixpunkt zu zwei stabilen und einem labilen zeigt Abb.7.2d Mit z=5 (vgl.Abb.7.2a) ist nur für B=.5,nicht aber für B =0 und B=.9 eine Bifurkation vorhanden.

7.2.2 Lage der Fixpunkte

Da p(x) symmetrisch um den Nullpunkt (p(x)=p(-x)) ist, genügt es, von den bei der Bifurkation vorhandenen beiden stabilen Fixpunkten ,die deshalb ebenfalls symmetrisch sind, nur einen zu betrachten.
Als Parameter m der Bifurkation wähle ich mir das Bifurkationskriterium

$$m := 2p(o).E(x/x>o) = 4 \cdot p(o) \cdot \int_0^\infty x p(x)\,dx$$

Wie groß ist nun $d^* = d^*(m)$?

Der Erwartungswert der Klassengrenze ist

$$\overline{d}(n+1) = \frac{1}{2}\left(E(x/x<d) + E(x/x>d)\right)$$

also

$$\overline{d}(n+1) = \frac{1}{2}\left[\frac{-\int_{-\infty}^{d_n} x\,p(x)\,dx}{\int_{-\infty}^{d_n} p(x)\,dx} + d_n \frac{\int_{d_n}^{\infty} x\,p(x)\,dx}{\int_{d_n}^{\infty} p(x)\,dx}\right]$$

$$= \frac{1}{2}\left[\frac{-\int_{-\infty}^{0} x\,p(x)\,dx + \int_{0}^{d_n} x\,p(x)\,dx}{\int_{-\infty}^{0} p(x)\,dx + \int_{0}^{d_n} p(x)\,dx} + \frac{\int_{d_n}^{0} x\,p(x)\,dx + \int_{0}^{\infty} x\,p(x)\,dx}{\int_{d_n}^{0} p(x)\,dx + \int_{0}^{\infty} p(x)\,dx}\right]$$

Mit

$$\int_{-\infty}^{0} p(x) = \int_{0}^{\infty} p(x) = \frac{1}{2} \quad , \quad \int_0^\infty x\,p(x)\,dx = \frac{m}{4\,p(o)}$$

und

$$\int_{-\infty}^{0} x\,p(x)\,dx = -\int_{0}^{\infty} x\,p(x)\,dx = -\frac{m}{4\,p(o)}$$

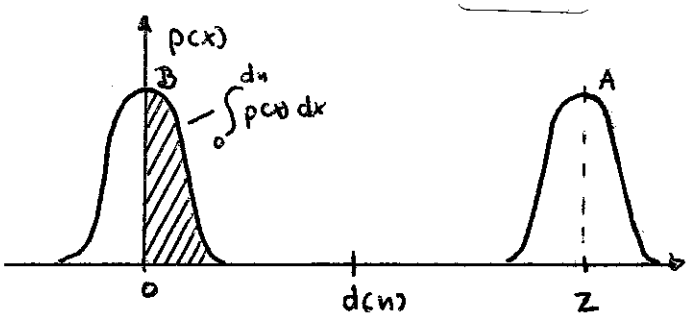
ist

$$\overline{d}(n+1) = \frac{1}{2}\left[\frac{-\frac{m}{4\,p(o)} + \int_0^{d_n} x\,p(x)\,dx}{1/2 + \int_0^{d_n} p(x)\,dx} + \frac{-\int_0^{d_n} x\,p(x)\,dx + \frac{m}{4\,p(o)}}{1/2 - \int_0^{d_n} p(x)\,dx}\right]$$

Da die Normalverteilungen sich nicht überlappen sollen($z > 4$) lassen sich folgende Näherungen vornehmen:

$$\int_0^{d_n} p(x)\,dx \approx \frac{B}{\sqrt{2\pi\sigma}} \cdot \int_0^\infty e^{-x^2/2\sigma}\,dx = \frac{B}{2}$$

(Integral über eine halbe Normalverteilung)



$$\int_0^{d_n} x\,p(x)\,dx \approx \frac{B}{\sqrt{2\pi\sigma}} \int_0^{\infty} x \cdot e^{-x^2/2\sigma}\,dx$$

$$= -\frac{B\cdot\sigma}{\sqrt{2\pi\sigma}} \cdot e^{-x^2/2\sigma}\Big|_0^{\infty} = \frac{B\cdot\sigma}{\sqrt{2\pi\sigma}} = p(0)\cdot\sigma$$

Damit ist

$$\bar{d}(n+1) = \frac{1}{2}\left[\frac{-\frac{m}{4p(0)} + p(0)\cdot\sigma}{\frac{1}{2}(1+B)} + \frac{\frac{m}{4p(0)} - p(0)\cdot\sigma}{\frac{1}{2}(1-B)}\right]$$

$$= \left(p(0)\cdot\sigma - \frac{m}{4p(0)}\right)\left(\frac{1}{1+B} + \frac{1}{B-1}\right)$$

$$= m \cdot \underbrace{\frac{1}{4p(0)}\left(\frac{2B}{1-B^2}\right)} + \underbrace{p(0)\cdot\sigma\cdot\left(\frac{2B}{B^2-1}\right)}$$

$$= m(z,\sigma,B)\ .\ a(B,\sigma) + b(B,\sigma)$$

Wenn d(n) also in den Bereich zwischen die Zentren fällt, in dem $p(x) \approx 0$, so ist d(n+1) nicht mehr von d(n) abhängig, sondern ein von n unabhängiger Wert, so daß gilt:

$$\bar{d}(n+1) = d(n) = d^* = m.a + b$$

d^* als Fixpunkt der Iteration ist dabei linear in m.

Wie ist dabei die Abhängigkeit des Fixpunkts vom Zentrenabstand z ?

$$d^* = 2p(0).E(x/x>0)\ .\ a + b$$
$$= E(x/x>0)\cdot\frac{1}{2}\cdot\left(\frac{2B}{1-B^2}\right) + p(0)\cdot\sigma\cdot\left(\frac{2B}{B^2-1}\right)$$

und mit Anhang A

$$E(x/x>0) = \frac{\frac{1-B}{2}\cdot\left(2\,\phi(y)\cdot z + 2\,\psi(y) - z\right) + B\cdot\frac{\sigma}{\sqrt{2\pi\sigma}}}{1/2}$$

$$= (1-B)(z) + p(0)\cdot\sigma\cdot 2$$

$\phi(y) \approx 1$
$y > 3$
$\psi(y) \approx \phi$

Damit ist

$$d^* = \frac{2B}{(B+1)(B-1)} \cdot \left(\rho(\omega) \cdot b - \frac{(1-B)}{2} \cdot z - \rho(\omega) \cdot b \right)$$

$$\boxed{d^* = z \cdot \frac{B}{B+1}}$$

d^* ist also auch linear in z !

Daraus läßt sich nun das B mit der größten Bifurkation, also größtem d^*, ablesen. Die Funktion $f(B) = \frac{B}{B+1}$ ist in Abb, 7.2 a dargestellt. Da sie monoton steigend ist, tritt also die Bifurkation mit größtem d bei dem B^+ auf, das größer 1/2 ist und für das $m(B^+) = 1$ gilt.

7.3 Computersimulation der Bifurkation

Es wurde das Konvergenzverhalten dreier verschiedener Algorithmen auf Bifurkationserscheinungen untersucht:

1) Iteration mit dem Erwartungswert

$$d(n+1)=\frac{1}{2}\left(E(x/x<d(n)) + E(x/x>d(n))\right)$$

p(x) sei dabei als Funktion bekannt.

2) Iteration mit dem Mittelwert

$$d(n+1)= \frac{1}{2}\left(\frac{1}{n_1} \sum_{\substack{i=1 \\ \forall x_i < d(n)}}^{n_1} x_i + \frac{1}{n_2} \sum_{\substack{i=1 \\ \forall x_i > d(n)}}^{n_2} x_i \right) \qquad x_i \in \{x(1),\dots,x(n)\}$$

p(x) wird dabei aus denen bis zum n-ten Schritt gespeicherten x(1),...,x(n) gebildet. x(i) ist eine stochastische Variable mit der Verteilung p(x(i)).

3) Stochastischer Algorithmus

$$d(n+1)= \frac{1}{2}(c_1(n)+c_2(n))$$

mit

$$c_i(n+1)=c_i(n) + \gamma_n(x(n+1)-c_i(n))$$

$$c_j(n+1)=c_j(n) \qquad i\neq j \text{, wenn } F_i(x(n+1),c_i) < F_j(x(n+1),c_j)$$

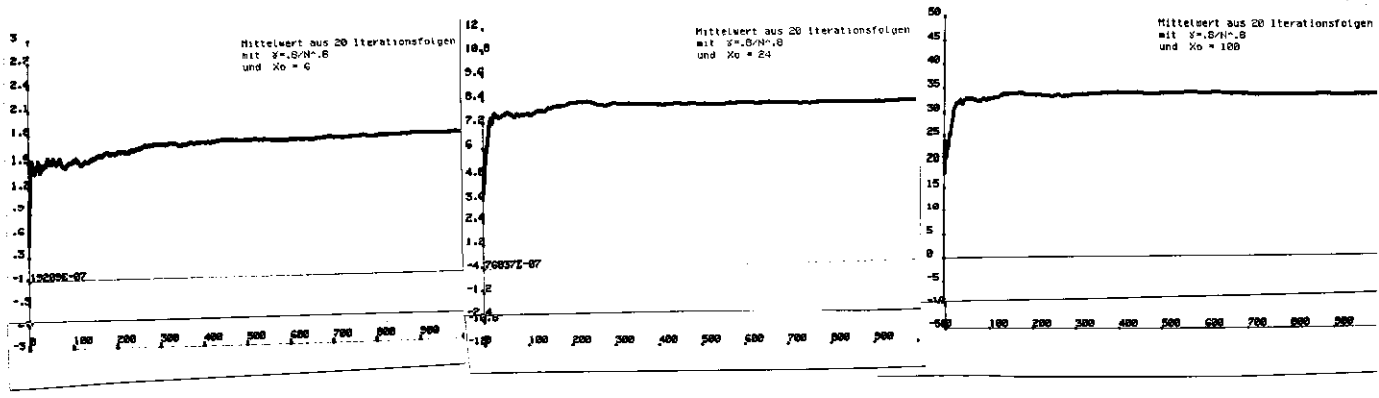
mit $F_i(x(n),c_i)= \frac{1}{2}(x(n)-c_i)^2$, $i,j \in \{1,2\}$

x(i) ist eine stochastische Variable mit der Verteilung p(x(i))

Jede Iteration wurde nach N Schritten abgebrochen, wenn eine "ausreichende" Konvergenz zum Fixpunkt vorlag. "Ausreichend" bedeutet für den Algorithmus in 1), daß sich der errechnete Wert bei der Iteration nicht mehr änderte, was sich innerhalb der ersten 10 Schritte vollzog. N wurde für 1) deshalb auf N:= 10 festgesetzt. p(x) wurde für alle drei Algorithmen mit

$$p(x)= A\cdot N(-z,1)+B\cdot N(o,1)+A\cdot N(z,1)$$ angenommen.

Für die Algorithmen in 2) und 3) bedeutet "ausreichend", daß die Abweichung in d(n) vom Mittelwert der letzten d(n) kleiner als ein Parameter sein soll,dessen günstigster Wert empirisch ermittelt wird.
Anstatt dieses Kriterium in die Computersimulation einzufügen(was zusätzlich Rechenzeit beansprucht), wurde der Algorithmus für verschiedene Werte von z simuliert. Die Ergebnisse sind in Abb. 7.3 a zu sehen.



Abbildungen 7.3 a

Für alle drei Simulationen war B=1/2,so daß mit Formel ein Fixpunkt von $\bar{d}^* = z/3$ zu erwarten war.
Man erkennt,daß der Algorithmus im Mittel diesen Wert bei N=100 gut erreicht.Für den stochastischen Algorithmus und den besser konvergierenden Algorithmus nach 2) wurde deshalb das Abbruchkriterium mit N:=200 fest gewählt.

Um das Bifurkationsverhalten ähnlich dem der Differenzialgleichung in Abb. 7.0b zu zeigen,wurden die Fixpunkte in Abhängigkeit der m ermittelt.Für den Algorithmus in 1) sind dies feste Punkte, für die anderen beiden Algorithmen eher Wahrscheinlichkeitsverteilungen um die $\bar{d}^*$ als Maximum.

In der Simulation wurde beiAbbruch der Iteration nach dem N-ten Schritt d(N) als Fixpunkt betrachtet.Bei festem m wurden die Iterationen 20 mal wiederholt,wobei die Startwerte d(o) gleichzufällig mit 0, +1,-1 variierten,also je d(o) $\approx$ 7 Iterationen . Dies wurde für 31 m - Werte wiederholt.

7.3.1 Simulationsmethoden

Für die grafische Darstellung der Funktion $d^*(m(z))$ ist es wichtig, daß die Variable $m(z)$ linear variiert wird. Da in $p(x)$ aber nicht $m(z)$, sondern z eingeht, muß z so variiert werden, daß $m(z)$ linear ist. Für $z \gg 3$ ist dies kein Problem, da $z \sim m(z)$. Im Bereich $z \approx 3$ aber treten Nichtlinearitäten durch die Gaußschen Fehlerfunktionen hinzu. Diese analytisch zu erfassen und damit $m^{-1}(m(z)) = z$ zu errechnen, ist recht schwierig. Stattdessen wurde mit einem interpolativen Algorithmus z solange iteriert, bis $m(z)$ dem gewünschten m_i bis auf 10^{-6} nahekam. Dies wurde für jedes i ($i=1,\ldots,31$) durchgeführt, so daß für jedes der linear ansteigenden m_i ein entsprechendes z_i ermittelt wurde.

Die für den Erwartungswert wichtige Gaußsche Fehlerfunktion $\Phi(x)$ wurde als punktweise definierte Funktion benutzt. Ihre Funktionswerte wurden für das Intervall $[0\ ,\ 3]$ der Literatur [6] entnommen. Für $-3 \leq x \leq 0$ wurde sie mit $\Phi(-x) = 1 - \Phi(x)$, für $x > 3$ mit $\Phi(x) := 1$ und für $x < -3$ mit $\Phi(x) := 0$ definiert.

Die Simulation von $p(x)$ wurde folgendermaßen durchgeführt: Zuerst wurden im Intervall $[0,1]$ gleichverteilte x_i erzeugt. Nach dem zentralen Grenzwertsatz konvergiert die Verteilung der standardisierten Summenvariablen x_s gegen die Normalverteilung $N(0,1)$ +)

$$\lim_{n\to\infty} x_s = N(0,1) \quad \text{mit} \quad x_s = \frac{\sum_{i=1}^{n} (x_i - \mu)}{\sigma(x_1 + \ldots + x_n)} \qquad \mu = E(x_i)$$

Setzt man die Voraussetzungen (Gleichverteilung) mit

$$\mu = \int_0^1 x\,dx = \frac{1}{2} \quad \text{und} \quad \sigma(x_1+\ldots+x_n) = \sqrt{\sigma_1^2+\sigma_2^2+\ldots\sigma_n^2} = \sqrt{n}\cdot\sigma(x_i) = \sqrt{n}\cdot\sqrt{E(x_i^2)-\mu^2}$$

$$E(x_i^2) = \int_0^1 x^2\,dx = \frac{1}{3}$$

so ist

$$x_s = \frac{1}{\sqrt{n}\cdot\sqrt{\frac{1}{12}}} \cdot \sum_{i=1}^{n} (x_i - \tfrac{1}{2})$$

Wir vernachlässigen den Fehler, bei endlichem n abzubrechen, und setzen der Einfachheit halber $n=12$. Man unterteilt

+) s. [1], S. 223

nun das $[0,1]$ Intervall in drei Intervalle I_1, I_2, I_3. Diese drei Intervalle haben jeweils eine solche Breite, daß $P(\text{Intervall}_i) = P(x \text{ aus } N_i(u_i,1)) = I_i$ mit $I_1 := I_2 := A$ und $I_3 := B$ und den Normalverteilungen $N_1(u_1,1) := N(-z,1)$, $N_2(u_2,1) := N(z,1)$ und $N_3(u_3,1) := N(o,1)$.

Die Verteilung $p(x) = p(x/N_i).P(N_i)$ ergibt sich dann daraus, daß für ein zu erzeugendes x mit $P(N_i)$ wie oben entschieden wird, aus welcher der drei Normalverteilungen es stammen wird und dann mit

$$x = u_i + \sum_{i=1}^{12} (x_i - \tfrac{1}{2}) \qquad x_i \text{ gleichverteilt aus } [0,1]$$

die Zufallsvariable x erzeugt wird.

7.3.2 Simulationsergebnisse

Die Lage der Fixpunkte $d^*(m)$ für den Algorithmus 1) der iterierten Erwartungswerte ist in Abb. 7.3 b wiedergegeben. Die Ergebnisse der Iteration mit dem Mittelwert und der stochastischen Iteration sind in Abb. 7.3 c und d aufgeführt. Beim Vergleichen der verschiedenen Ergebnisse fällt auf, daß in Abb. beim errechneten Erwartungswert der labile Fixpunkt bei $d^*=0$ existiert, während dies bei denen durch Zufallszahlen bestimmten Algorithmen nicht der Fall ist. Dies erklärt sich dadurch, daß für 1) bei angenommenen d(o):= 0 auch d(1)=d(n)=0 sein muß, während bei 2) und 3) die erzeugten Zufallszahlen die Klassengrenze d(n) nie exakt =0 werden lassen. Dadurch führt aber schon die nächste Iteration von $d^*=0$ weg, so daß $d^*=0$ nie als Fixpunkt erscheint.

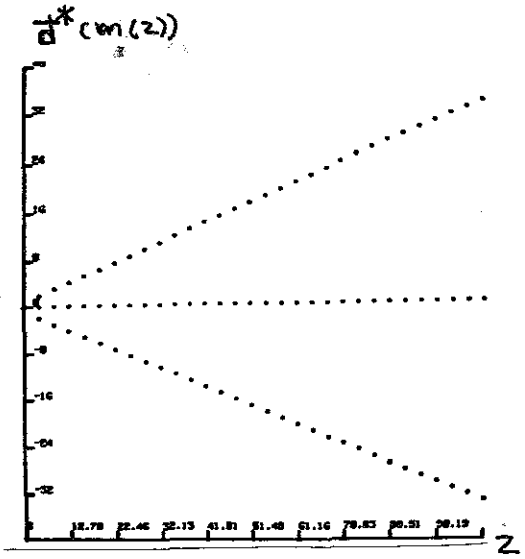


Abb. 7.3 b

Lage der Fixpunkte bei Iteration mit dem Erwartungswert

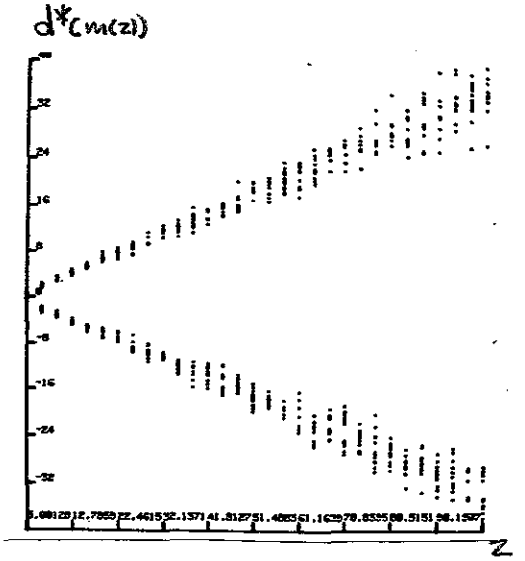


Abb. 7.3 c

Lage der Fixpunkte bei Iteration mit dem Mittelwert

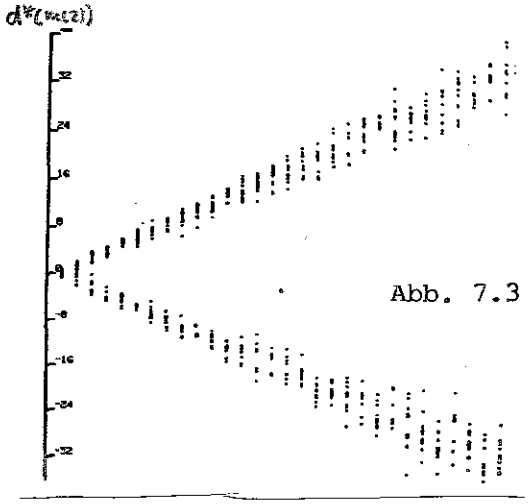


Abb. 7.3 d

Abb.7.3 d Stochast.Iteration

In Abb. 7.3 b und Abb.7.3c,d sind die Wahrscheinlichkeitsdichten der d^* nur als Punktdichten erkennbar.
Zur besseren Veranschaulichung sind in Abb. 7.3 e,f,g die Dichten als dreidimensional schräg versetzte Histogramme aufgetragen, bei denen nur jede 2.m-Wert der Überschaubarkeit wegen mit allen 20 $d^*(m(z))$ gezeichnet wurde.

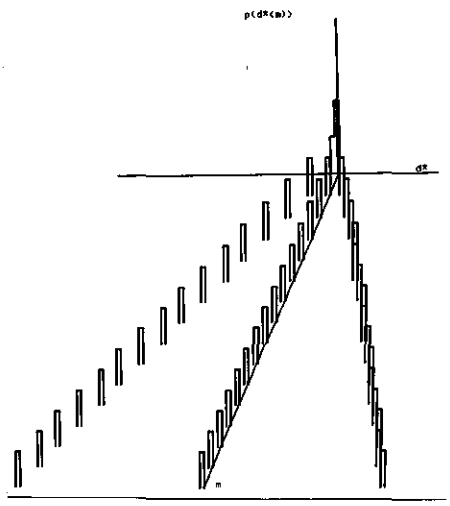


Abb. 7.3 e
Iteration mit dem Erwartungswert

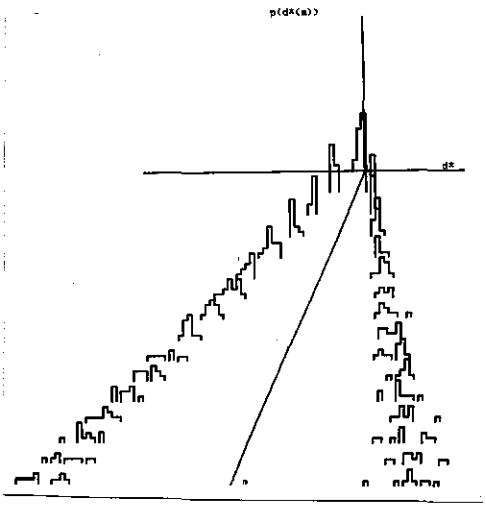


Abb. 7.3 f
Iteration mit dem Mittelwert

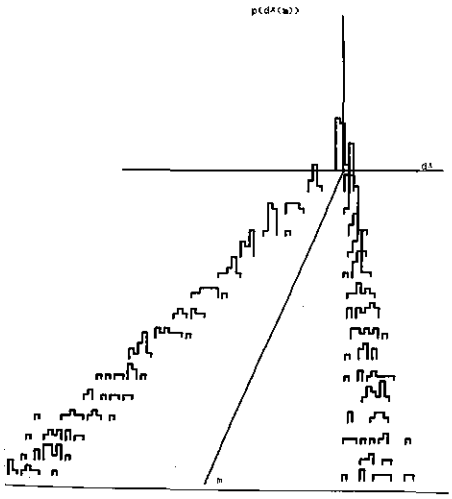
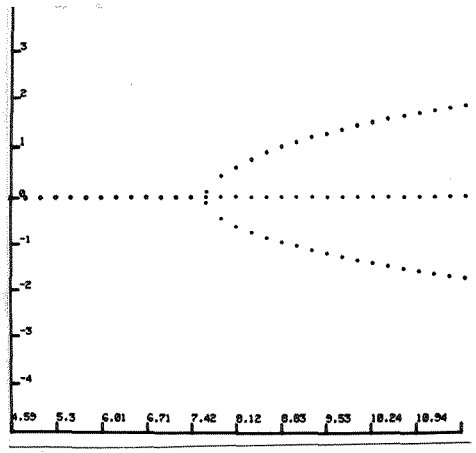


Abb. 7.3 g
Stochast. Iteration

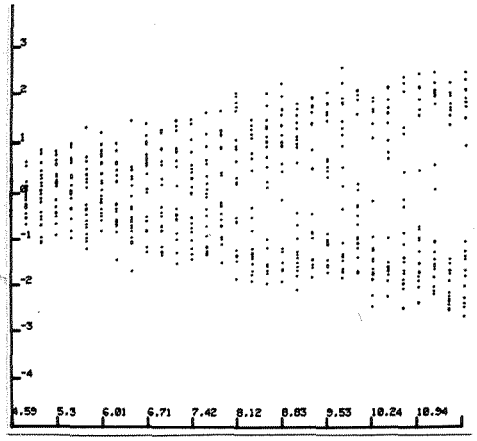
Man sieht deutlich, daß die Unterschiede zwischen dem stochastischen Algorithmus und der Iteration mit dem Mittelwert hinsichtlich der Streuung um den Fixpunkt nach 200 iterationen nicht allzu groß sind.

Bei genauer Betrachtung der Abbildungen stellt sich sofort die Frage: Die Bifurkation der Abbildungsgleichungen ist hier mit großer Variation von m bzw. von z gezeigt worden. Wie sieht aber bei geringer Variation von m um eins der Übergang von einem Fixpunkt der Abbildungen zu dreien bzw. zweien aus ?

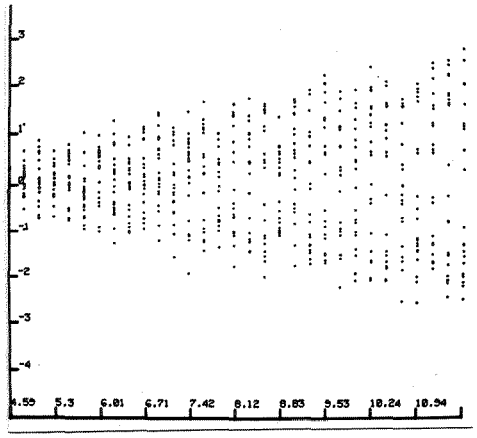
Diese Übergänge sind für alle drei Algorithmen in den folgenden Abbildungen dargestellt.
Auch hier fällt wieder, besonders in den Histogrammen, der geringe Unterschied der Algorithmen 2) und 3) auf, die Beide durch ihre stochastische Abhängigkeit einen Übergang von einem Fixpunkt zu zweien nur erahnen lassen.



Iteration mit dem Erwartungswert



Iteration mit dem Mittelwert



Stochastische Iteration

Kapitel 8 Anwendungen des Stochastischen Lernens

Auch gewisse biologische Prozesse lassen sich als Lernprozesse interpretieren.
Als Beispiel kann man die Vermehrung von Lebewesen in der Populationsgenetik betrachten.
Üblicherweise werden in der Populationsbilogie aus statistischen Grundannahmen Aussagen über Erwartungswerte von Populationen gemacht,ohne dabei etwas über Evolutionsziele auszusagen.
Bei der Interpretation der haploiden und diploiden Vermehrung als Lernprozeß ergibt sich als Lernziel,die Fitness der Gesamtpopulation so groß wie möglich zu machen,was für die Wahrscheinlichkeit des Überlebens der Art sicher entscheidend ist.

8.1 Haploide Vermehrung

Nach [8] ,S.74 ist der haploide Evolutionsfall dadurch ausgezeichnet,daß ein Allel A_1 etwas besser als ein zweites Allel A_2 ist. Die Überlebensrate oder relative Eignung(*Fitness*) jedes Allels in dem durch das Allel gekennzeichnete Lebewesen nach jedem Generationszyklus sei quantitativ mit

$$\frac{\text{Fitness von } A_1}{\text{Fitness von } A_2} = \frac{1+\varepsilon}{1} \quad \text{mit } 0 \lesssim \varepsilon \ll 1$$

gegeben.Für die Zahl N_i der Lebewesen mit Gen A_i *(Population N_i)* bedeutet dies nach der n-ten Generation (n-tem Zeitschritt)

$$N_1(n+1) = N_1(n) \cdot (1+\varepsilon)$$
$$N_2(n+1) = N_2(n) \cdot 1$$

Der relative Anteil der Lebewesen mit A_1 an der Gesamtpopulation ist damit

$$c_1(n+1) =: \frac{N_1(n+1)}{N_1(n+1)+N_2(n+1)} = \frac{(1+\varepsilon)\,N_1(n)}{(1+\varepsilon)\,N_1(n)+N_2(n)}$$

Mit

$$c_1(n) := \frac{N_1(n)}{N_1(n)+N_2(n)} \qquad \text{ist} \qquad N_1(n)+N_2(n) = \frac{N_1(n)}{c_1(n)}$$

so daß

$$c_1(n+1) = \frac{(1+\varepsilon)\,N_1(n)}{\left(\frac{1}{c_1(n)}+\varepsilon\right)N_1(n)} = \frac{(1+\varepsilon)\,c_1(n)}{1+\varepsilon\cdot c_1(n)}$$

$$= \frac{(1+\varepsilon)\cdot c_1(n) - \left(1+\varepsilon c_1(n)\right)\cdot c_1(n)}{1+\varepsilon\cdot c_1(n)} \qquad +c_1(n) = c_1(n) + \frac{c_1(n)(1-c_1(n))\cdot\varepsilon}{1+\varepsilon\cdot c_1(n)}$$

Angenommen, der Vermehrungszuwachs sei von zufälligen Umweltbedingungen x_i wie Wetter, Auftreten von Feinden usw. abhängig:

$$\varepsilon := \varepsilon(x_i) \qquad -1 \ll \varepsilon(x) \ll 1$$

Dann ist

$$c_1(n+1) = c_1(n) + \gamma_n \cdot f(x_n, c_1(n))$$

$$c_2(n+1) = 1 - c_1(n+1)$$

mit $\gamma_n = c_1(n)(1-c_1(n))$

$$f(x_n, c_1(n)) = \frac{\varepsilon(x_n)}{1+\varepsilon(x_n)\cdot c_1(n)}$$

ein stochastischer Algorithmus, der sich als Lernalgorithmus mit der Straffunktion

$$\nabla_{c_1} F(x_n, c_1(n)) = f(x_n, c_1(n))$$

$$F(c_1(n), x_n) := -\ln\,(1+\varepsilon(x_n)\,c_1)$$

$$= -\ln\left(\frac{(1+\varepsilon(x_n)\cdot N_1 + N_2}{N_1+N_2}\right)$$

$$=- \ln(\text{ Fitness der Gesamtpopulation})$$

interpretieren läßt.

Mit der Näherung $1+\varepsilon(x_1)c_1(n) \approx 1$ ist

$$c_1(n+1) = c_1(n) + \gamma_n \cdot \varepsilon(x_n)$$

mit der Straffunktion

$$F(c_1(n), x_n) = -\left(1+\varepsilon(x_n)\,c_1(n)\right)$$

Das lernende System hat hier also als Lernziel, die Strafe so klein wie möglich oder die negative Strafe, die Fitness der Gesamtpopulation, so groß wie möglich zu machen.

Was sind nun die Fixpunkte des Algorithmus ?

Die Strafe ist

$$J(c_1) = E(F(x,c_1)) = \int_{-\infty}^{\infty} \ln(1+\varepsilon(x)c_1)\, p(x)\,dx$$

so daß die Extremalbedingung lautet

$$\frac{\partial}{\partial c_1} J(c_1^*) = \int_{-\infty}^{+\infty} \frac{\varepsilon(x)}{1+c_1^*\cdot\varepsilon(x)}\, p(x)\,dx \stackrel{!}{=} 0$$

Angenommen, $\varepsilon(x)$ sei mit großer Warscheinlichkeit $\ll 1$ und $c_1^* < 1$, also auch $c_1^{*2}\cdot\varepsilon^2(x) \approx 0$.
Dann ist

$$\frac{\varepsilon(x)}{1+c_1^*\cdot\varepsilon(x)} = \frac{\varepsilon(x)(1-c_1^*\cdot\varepsilon(x))}{1-c_1^{*2}\varepsilon^2(x)} \approx \varepsilon(x) - c_1^*\varepsilon^2(x)$$

und damit gilt

$$\int_{-\infty}^{+\infty} (\varepsilon(x) - c_1^*\varepsilon^2(x))\, p(x)\,dx = 0$$

$$\bar{\varepsilon}(x) - c_1^*\bar{\varepsilon^2}(x) = 0$$

$$c_1^* = \frac{\bar{\varepsilon}(x)}{\bar{\varepsilon^2}(x)}$$

Mit $0 < |\varepsilon(x)| < 1$ ist

$$\varepsilon(x) - \varepsilon^2(x) > 0 \quad \text{bei } \varepsilon(x) > 0$$

$$\int_{-\infty}^{\infty} (\varepsilon(x) - \varepsilon^2(x))\, p(x)\,dx > 0$$

Also ist

$$\frac{\bar{\varepsilon}(x)}{\bar{\varepsilon^2}(x)} > 1 \quad \text{und} \quad \frac{\bar{\varepsilon}(x)}{\bar{\varepsilon^2}(x)} < 0 \quad \text{bei } \bar{\varepsilon}(x) < 0$$
$$\varepsilon(x) > 0$$

Da c_1^* als Konzentration nur zwischen 0 und 1 variieren kann, ist

$$c_1^* \begin{cases} 1 & \bar{\varepsilon}(x) > 0 \\ 0 & \bar{\varepsilon}(x) < 0 \end{cases}$$

Für das Überleben einer Genart ist also nur entscheidend, ob das von ihr geprägte Lebewesen im Mittel einen Selektionsvorteil gegenüber der anderen Art hat oder nicht.

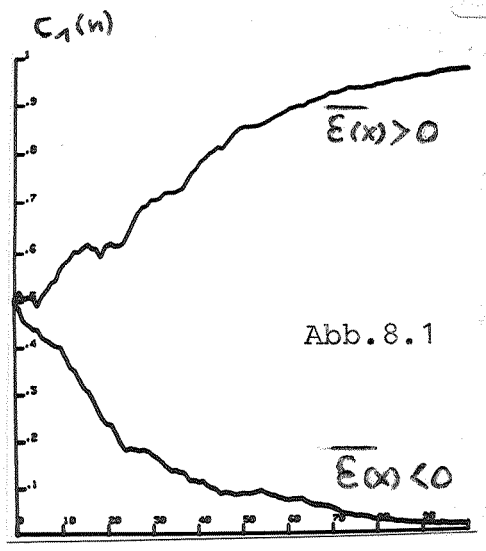


Abb.8.1

Abb. 8.1 zeigt eine Simulation dieses Prozesses. Die Konzentration c_1 ist aufgetragen als Funktion des Iterationsschritts. Parameter dafür sind $\overline{\varepsilon(n)} > 0$ für größere und $\overline{\varepsilon(n)} < 0$ für kleinere Fitness von A1 zu A_2 .

Die Wahl von $\gamma_n = c_1(n)(1-c_1(n))$ ist sinnvoll, da

$$\lim_{c_1(n)\to 1} \gamma_n = 0$$

$$\lim_{c_1(n)\to 0} \gamma_n = 0$$

8.2 Diploider Fall

Sei für die Fitness für ein Lebewesen mit bestimmter Merkmalsausprägung (*Phänotyp*) die Kombination der Allele A_1 und A_2 entscheidend.
Dann sei für die relative Fitness

Fitness A_1A_1 : Fitness A_1A_2 : Fitness A_2A_2

ist wie

$1+\varepsilon_1$: $1+\varepsilon_1\cdot\varepsilon_2$: 1

und die Iterationsgleichung ergibt sich nach [8] S.81 zu

$$c_1(n+1) = \frac{c_1(n) + \varepsilon_1 \cdot c_1(n)\left(c_1(n) + \varepsilon_2 \cdot c_2(n)\right)}{1 + \varepsilon_1 \cdot c_1(n)\left(c_1(n) + 2\varepsilon_2 \cdot c_2(n)\right)} \qquad \text{Formel (8.2)}$$

$$= c_1(n) + \frac{-\left(1+\varepsilon_1 \cdot c_1(n)\left(c_1(n) + 2\varepsilon_2 \cdot c_2(n)\right)\right) c_1(n) + c_1(n) + \varepsilon_1 \cdot c_1(n)\left(c_1(n) + \varepsilon_2 \cdot c_2(n)\right)}{1 + \varepsilon_1 c_1(n)\left(c_1(n) + 2\varepsilon_2 \cdot c_2(n)\right)}$$

$$= c_1(n) + \varepsilon_1 \cdot c_1(n) \cdot \frac{-c_1(n)\cdot\left(c_1(n) + 2\varepsilon_2 \cdot c_2(n)\right) + \left(c_1(n) + \varepsilon_2 \cdot c_2(n)\right)}{1 + \varepsilon_1 c_1(n)\left(c_1(n) + 2\varepsilon_2 \cdot c_2(n)\right)}$$

$c_2(n) = 1 - c_1(n)$

$$= c_1(n) + \varepsilon_1 \cdot c_1(n)\cdot(1-c_1(n)) \quad \frac{c_1(n) + \varepsilon_2 - c_1(n)\cdot 2\varepsilon_2}{1 + \varepsilon_1 c_1(n)\left(c_1(n) + 2\varepsilon_2 (1-c_1(n))\right)} \qquad \text{Formel (8.3)}$$

$$:= c_1(n) + \gamma_n(c_1(n)) \cdot f(\varepsilon_1, \varepsilon_2, c_1(n))$$

Mit der Verallgemeinerung

$$\varepsilon_1 := \varepsilon_1(x_n) \qquad -1 \ll \varepsilon_1(x_n) \ll 1$$
$$\varepsilon_2 := \varepsilon_2(x_n)$$

läßt sich dies wieder als stochastischer Algorithmus formulieren:

$$c_1(n+1) = c_1(n) + \gamma_n \cdot f(x_n, c_1(n))$$
$$c_2(n+1) = c_2(n)$$

Bei der Interpretation des Algorithmus als Lernprozeß ergibt sich als Straffunktion

$$F(x_n, c_1(n)) = -\frac{1}{2} \ln \left(\text{Gesamtpopulation}\right)$$

was sich durch Einsetzen zeigen läßt:

Mit [8] S81 ist die Zahl der Gammeten in

$A_1 A_1$	$A_1 A_2$	$A_2 A_2$
$c_1^2(n) \cdot (1+\varepsilon_1(x_n))$	$2 c_1 (1-c_1)(1+\varepsilon_1(x_n)\varepsilon_2(x_n))$	$(1-c_1)^2$

Da pro Phänotypus je 2 Gammeten existieren, ist Gesamtpopulation die Hälfte der Gesamtgammetenzahl.
Also ist

$$F(x_n, c_1(n)) = -\frac{1}{2} \ln \frac{1}{2}\left[c_1^2(n) \cdot (1+\varepsilon_1(x_n)) + 2 c_1(1-c_1)(1+\varepsilon_1\varepsilon_2) + (1-c_1)^2\right]$$

$$-\frac{\partial F(x_n, c_1(n))}{\partial c_1} = +\frac{1}{2}\left[c_1^2 \varepsilon_1(1-2\varepsilon_2) + 2\varepsilon_1\varepsilon_2 c_1 + 1\right]^{-1} \cdot \frac{\partial}{\partial c_1}[\ldots]$$

$$= \left[1 + c_1\varepsilon_1(c_1 + 2\varepsilon_2(1-c_1))\right]^{-1} \cdot \varepsilon_1(c_1(1-2\varepsilon_2) + \varepsilon_2) = f(x_n, c_1)$$

Auch hier läßt sich bei kleinem $\varepsilon_1(n)$ $f(x_n, c_1)$ nähern zu

$$f(x_n, c_1) \approx \varepsilon_1\left[c_1 + 2\varepsilon_2(1-c_1) - \varepsilon_2\right] \quad [8]\ \text{S.81 (17)}$$

so daß sich die Straffunktion zu

$$F(x_n, c_1(n)) = \frac{1}{2}\left(\text{Gesamtpopulation}\right)$$

ergibt.

8.2.1 Die Fixpunkte des Algorithmus

Die Fixpunkte der Iteration ergeben sich aus der Forderung, daß an den Fixpunkten

$$c(n+1)=c(n):=c^*$$

gelten soll. Dann ist

$$\frac{c(n+1)-c(n)}{1}:=\dot{c}(n)\stackrel{!}{=}0 \qquad \dot{} := \frac{\partial c(n)}{\partial n}$$

Die Ableitung $\frac{\partial c(n)}{\partial n}$ wird also als Differenzenquotient mit der Schrittweite Eins gebildet.

Für den Algorithmus der diploiden Vermehrung bedeutet dies mit Formel (8.3) $\left(c(n):=c\right)$

$$\dot{c}=\frac{\varepsilon_1\cdot c\cdot(1-c)\cdot\left(c(1-2\varepsilon_2)+\varepsilon_2\right)}{1+\varepsilon_1\cdot c\cdot\left(c(1-2\varepsilon_2)+2\varepsilon_2\right)}=\frac{\text{Zähler}(c,\varepsilon_1,\varepsilon_2)}{\text{Nenner}(c,\varepsilon_1,\varepsilon_2)}\stackrel{!}{=}0$$

Der Bruch ist NULL, wenn bei nichtverschwindendem Nenner der Zähler $Z(c,\varepsilon_1,\varepsilon_2)$ Null wird, was bei diesem Polynom 3. Grades an drei Werten von c geschieht:

1) $c_1^*=0$

2) $c_2^*=1$

3) $c_3^*=\frac{\varepsilon_2}{2\varepsilon_2-1}$

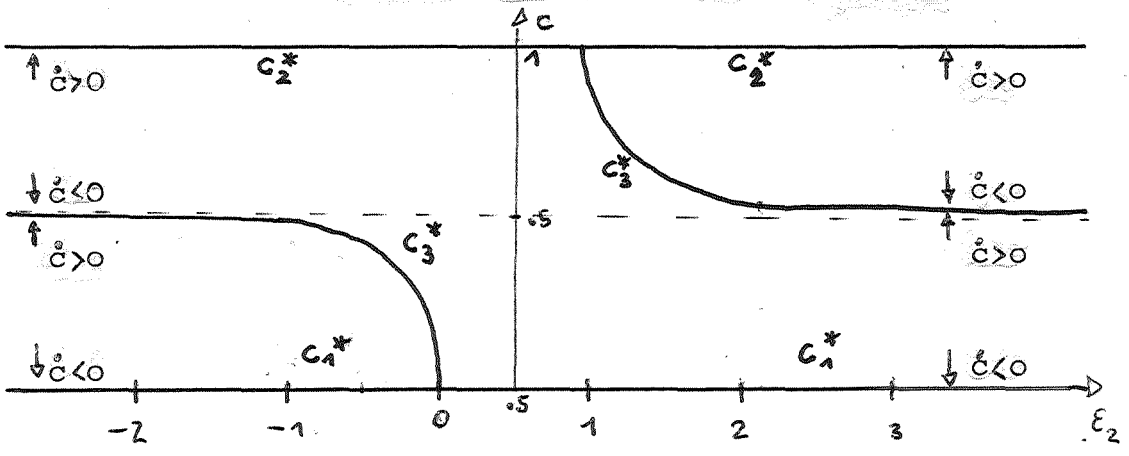
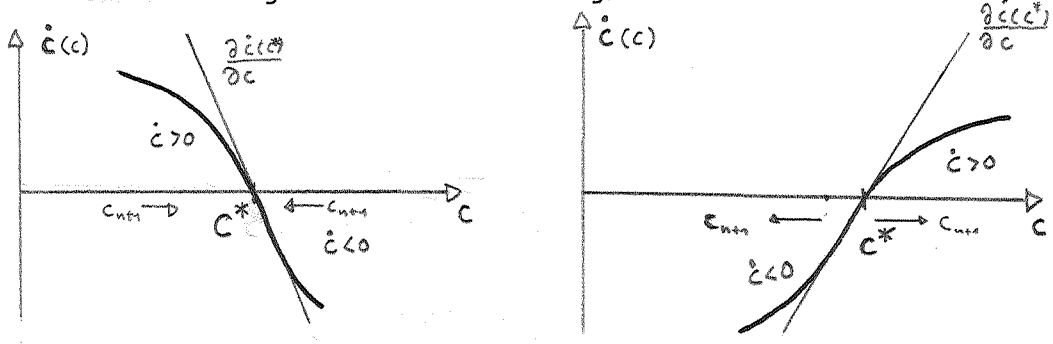


Abb. 8.2 Lage der Fixpunkte in Abhängigkeit von ε_2

Bei welchen Werten von und sind diese Fixpunkte *labil* und wann *stabil* ?

In Abb.8.2 sind die Bedingungen für c einer kleinen Umgebung um die Fixpunkte eingetragen, bei der diese stabil sind. Den allgemeinen Fall zeigt Abb. 8.3.



stabiler Fixpunkt Abb. 8.3 labiler Fixpunkt

Wie man bemerkt, ist bei einem stabilen Fixpunkt die Steigung der Tangente, also $\frac{\partial \dot{c}(c)}{\partial c}$, kleiner als Null, bei einem labilen Fixpunkt größer Null (Vgl. Kapitel 7.0).

Dieses Kriterium soll nun für den Algorithmus (8.2) angewendet werden.

Mit Formel (8.4) ist

$$\dot{c}(c, \varepsilon_1, \varepsilon_2) = \frac{Z(c, \varepsilon_1, \varepsilon_2)}{N(c, \varepsilon_1, \varepsilon_2)}$$

und somit

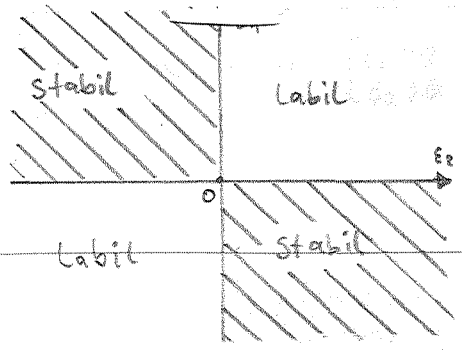
$$\frac{\partial}{\partial c} := ' \qquad \dot{c}'(c^*) = \frac{Z'(c^*)\cdot N(c^*) - Z(c^*)\cdot N(c^*)}{N^2(c^*)} = \frac{Z'(c^*)}{N(c^*)}, \text{ da } Z(c^*)=0 .$$

$$= \frac{\varepsilon_1 \cdot \left(3c^2(2\varepsilon_2-1)+2c(1-3\varepsilon_2)+\varepsilon_2\right)}{1+c^2\varepsilon_1(1-2\varepsilon_2)+c\,2\varepsilon_1\varepsilon_2}$$

An den Fixpunkten ist dies

1) $c_1^* = 0$

$$\left.\begin{array}{l} Z'(0) = \varepsilon_1\varepsilon_2 \\ N(0) = 1 \end{array}\right\} \quad \dot{c}'(0) = \varepsilon_1\varepsilon_2$$



Für $\begin{cases} \varepsilon_1 > 0 \\ \varepsilon_2 < 0 \end{cases}$ oder $\begin{cases} \varepsilon_1 < 0 \\ \varepsilon_2 > 0 \end{cases}$

ist $\dot{c}'(c_1^*) < 0$ und damit c_1^* stabil,

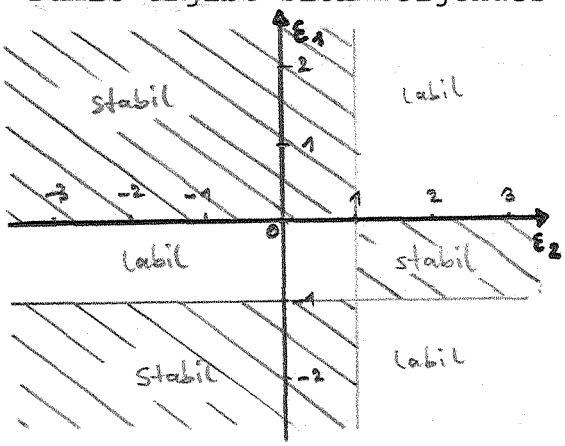
sonst Labil.

2) $c_2^* = 1$

$$Z'(c_2^*, \varepsilon_1, \varepsilon_2) = \varepsilon_1(3(2\varepsilon_1-1) + 2(1-3\varepsilon_2) + \varepsilon_2) = \varepsilon_1(\varepsilon_2-1)$$

$$N(c_2^*, \varepsilon_1, \varepsilon_2) = \varepsilon_1(1-2\varepsilon_2) + 2\varepsilon_1\varepsilon_2 + 1 = \varepsilon_1 + 1$$

Damit ergibt sich folgendes Schema:



ε_1	ε_2	$\mathring{c}^1(c_2^*, \varepsilon_1, \varepsilon_2) = \frac{\varepsilon_1(\varepsilon_2-1)}{\varepsilon_1+1}$
$-1 > \varepsilon_1$	> 1	> 0
	< 1	< 0
$-1 < \varepsilon_1 < 0$	> 1	< 0
	< 1	> 0
$\varepsilon_1 > 0$	> 1	> 0
	< 1	< 0

3) $c_3^* = \dfrac{\varepsilon_2}{2\varepsilon_2-1}$

$$z'(c_3^*, \varepsilon_1, \varepsilon_2) = \varepsilon_1\left(3\left(\frac{\varepsilon_2}{2\varepsilon_2-1}\right)^2 \cdot (2\varepsilon_2-1) + 2\frac{\varepsilon_2}{2\varepsilon_2-1} \cdot (1-3\varepsilon_2) + \varepsilon_2\right)$$

$$= \varepsilon_1\varepsilon_2 \cdot \left(\frac{3\varepsilon_2+2-6\varepsilon_2}{2\varepsilon_2-1} + 1\right)$$

$$= \varepsilon_1\varepsilon_2 \cdot \left(1 - \frac{3\varepsilon_2-2}{2\varepsilon_2-1}\right)$$

$$N(c_3^*, \varepsilon_1, \varepsilon_2) = 1 + \left(\frac{\varepsilon_2}{2\varepsilon_2-1}\right)^2 \varepsilon_1 \cdot (1-2\varepsilon_2) + \frac{\varepsilon_2}{2\varepsilon_2-1} 2\varepsilon_1\varepsilon_2$$

$$= 1 + \frac{-\varepsilon_1\varepsilon_2^2 + 2\varepsilon_1\varepsilon_2^2}{2\varepsilon_2-1} = 1 + \varepsilon_1\frac{\varepsilon_2^2}{2\varepsilon_2-1}$$

Polarität von $Z'(c_3)$ für verschiedene $\varepsilon_1, \varepsilon_2$:

a) $\varepsilon_1 > \emptyset$

$\varepsilon_2 > \emptyset$ wenn $\varepsilon_2 < 1$,

so $3\varepsilon_2 - 2 < 2\varepsilon_2 - 1$

$\dfrac{3\varepsilon_2-2}{2\varepsilon_2-1} < 1$, wenn auch $(2\varepsilon_2-1) > \emptyset$ $\not\Leftrightarrow$ $\varepsilon_2 > 1/2$

$\Rightarrow \varepsilon_1\varepsilon_2\left(1 - \dfrac{3\varepsilon_2-2}{2\varepsilon_2-1}\right) = z'(c_3^*) > \emptyset$

Wenn $2\varepsilon_2 - 1 < \emptyset$ $\not\Leftrightarrow$ $\varepsilon_2 < 1/2$, so ist $z'(c_3^*) < \emptyset$.

Wenn $\varepsilon_2 > 1$, so ist $z'(c_3^*) < \emptyset$ mit obiger Rechnung.

b) $\varepsilon_1 > \emptyset$, $\varepsilon_2 < \emptyset$

Wenn $\varepsilon_2 < \emptyset < 1$,

so ist $3\varepsilon_2 - 2 < 2\varepsilon_2 - 1$ $\qquad 2\varepsilon_2 - 1 < \emptyset$, da $\varepsilon_2 < 1/2$

$$\frac{3\varepsilon_2 - 2}{2\varepsilon_2 - 1} > 1$$

$$\emptyset > \left(1 - \frac{3\varepsilon_2 - 2}{2\varepsilon_2 - 1}\right)$$

$$\varepsilon_1 \varepsilon_2 \left(1 - \frac{3\varepsilon_2 - 2}{2\varepsilon_2 - 1}\right) = Z^i(c_3^*) \quad \emptyset$$

Es ergibt sich mithin folgendes Schema:

$\varepsilon_1 > \emptyset$ ε_2	$Z^i(c_3^*, \varepsilon_1, \varepsilon_2)$	Bei $\varepsilon_1 < \emptyset$	$Z^i(c_3^*, \varepsilon_1, \varepsilon_2)$
$\varepsilon_2 < \emptyset$	$> \emptyset$		$< \emptyset$
$\emptyset < \varepsilon_2 < 1/2$	$< \emptyset$		$> \emptyset$
$1/2 < \varepsilon_2 < 1$	$> \emptyset$		$< \emptyset$
$1 < \varepsilon_2$	$< \emptyset$		$> \emptyset$

Polarität des Nenners $N(c_3^*, \varepsilon_1, \varepsilon_2)$:

Sei $\varepsilon_2 > 1/2$. Dann ist bei

$$\varepsilon_1 > \frac{1 - 2\varepsilon_2}{\varepsilon_2^2}$$

$$\varepsilon_1 \varepsilon_2^2 > 1 - 2\varepsilon_2 \qquad 1 - 2\varepsilon_2 < \emptyset$$

$$\frac{\varepsilon_1 \varepsilon_2^2}{2\varepsilon_2 - 1} > -1$$

$$1 + \varepsilon_1 \frac{\varepsilon_2^2}{2\varepsilon_2 - 1} = N(c_3^*) > \emptyset$$

Also ist bei

$$\varepsilon_1 < \frac{1 - 2\varepsilon_2}{\varepsilon_2^2} \Rightarrow N(c_3^*) < \emptyset$$

Bei $\varepsilon_2 < 1/2$ ist die Polarität jeweils umgekehrt und es ergibt sich das Schema

$$f(\varepsilon_2) := \frac{1 - 2\varepsilon_2}{\varepsilon_2^2}$$

	ε_1	$N(c_3^*, \varepsilon_1, \varepsilon_2)$
$\varepsilon_2 < 1/2$	$> f(\varepsilon_2)$	$< \emptyset$
	$< f(\varepsilon_2)$	$> \emptyset$
$\varepsilon_2 > 1/2$	$> f(\varepsilon_2)$	$> \emptyset$
	$< f(\varepsilon_2)$	$< \emptyset$

Diese Bedingungen sind in Abb. 8.4 eingetragen; dazu ist mit dem sich ergebenden $\dot{c}'(c_3^*, \varepsilon_1, \varepsilon_2)$ das Stabilitätsgebiet markiert.

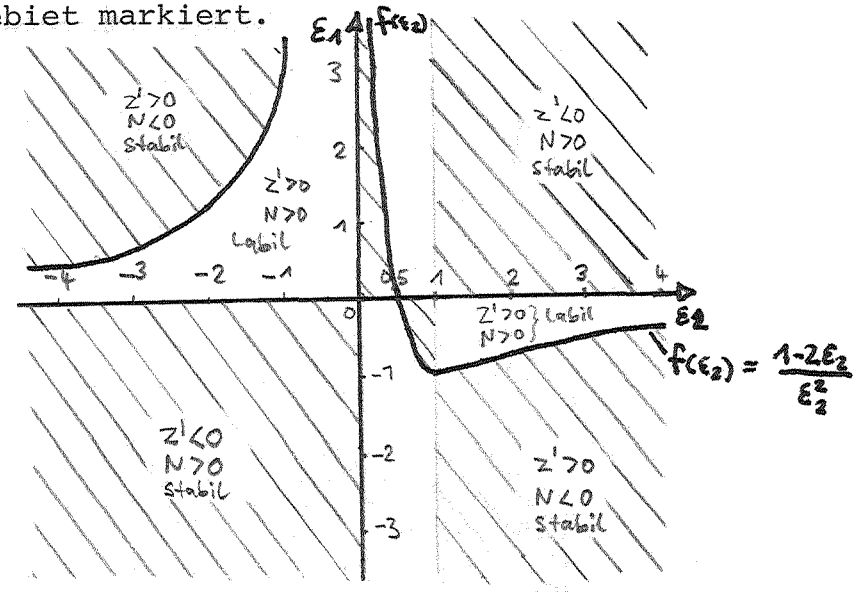


Abb. 8.4

Die stabilen $\varepsilon_1, \varepsilon_2$-Gebiete aller drei Fixpunkte sind in Abb. 8.5 zusammengetragen. In jedem Gebiet ist nur markiert, welche der Fixpunkte stabil sind; nichtgenannte sind also darin labil.

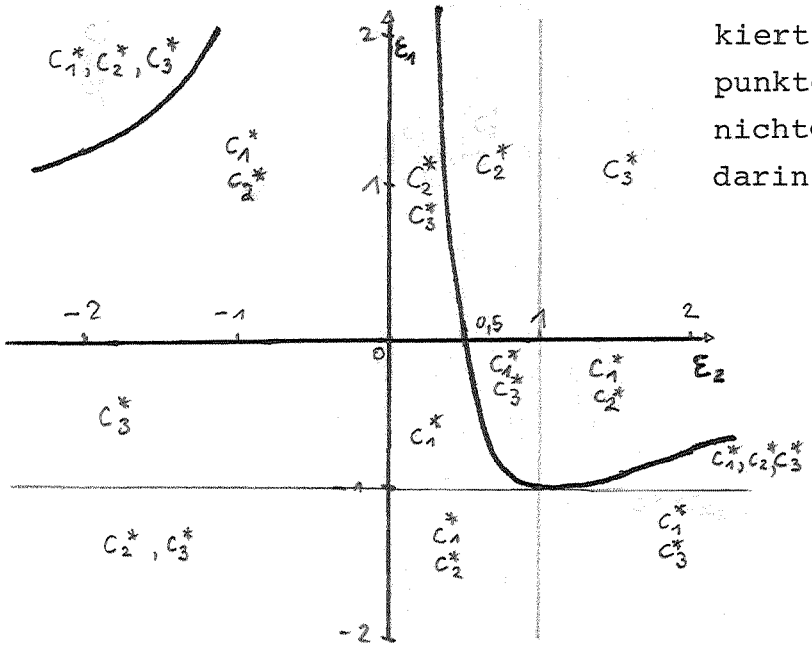


Abb. 8.5

8.2.2 Bifurkation der Fixpunkte

Man bezeichnet als Bifurkation ein Verzweigen der Lösungsmenge einer Differenzialgleichung bei Variation eines Parameters.(Vgl. Abb.7.0b)
Dieses Verhalten ist bei der Gleichung(8.2) der diploiden Vermehrung immer dann vorhanden, wenn als Parameter oder variiert wird und dabei ein Stabilitätswechsel der Fixpunkte auftritt.(vgl. Abb. 8.5 bzw. Abb. 8.8).
Ein Beispiel ($\varepsilon_1 > 0$, $\varepsilon_2 = 1$) ist in Abb. 8.6 zu sehen.

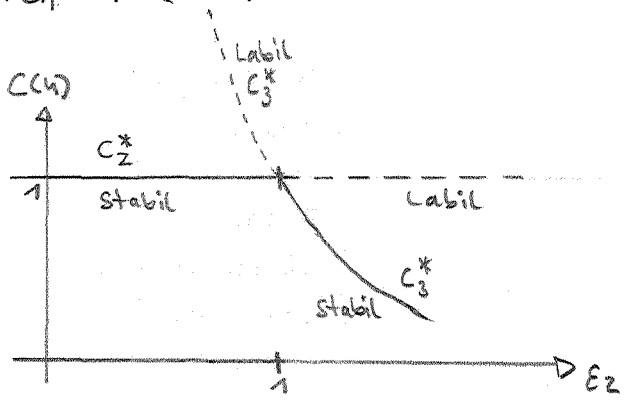


Abb. 8.6 Stabilitätsübertragung und Bifurkation der Fixpunkte.

Für $\varepsilon_1 < 0$ und $\varepsilon_2 = 0$ ist dies analog der Fall.

Zu der abgebildeten Bifurkation läßt sich auch eine Katastrophe konstruieren.
Dies wäre dann eine *Fold*-Katastrophe mit nichtverseller Entfaltung, worauf ich aber aus Platzgründen nicht näher eingehen möchte.

8.2.3 Der Geltungsbereich des Modells

In der biologischen Realität sind Konzentrationen, die kleiner Null oder größer Eins sind, ziemlich unsinnig. Genau dieser Fall tritt aber auf, wenn der Nenner in Formel (8.2) Null wird und damit der Quotient unendlich. Für den Geltungsbereich dieses Modells müssen wir also mindestens fordern, daß die Pole nicht im biologisch sinnvollen Bereich $[0,1]$ liegen. Eine weitere Möglichkeit ist, nur solche Parameter ε_1 und ε_2 zuzulassen, bei denen keine Polstellen existieren.
Wann existieren nun solche Pole ?
Die Abbildungsgleichung (8.2) hat dann Pole, wenn der Nenner Null wird.
Also ist

$$1 + \varepsilon_1 c(c + 2\varepsilon_2(1-c)) \stackrel{!}{=} 0$$

$$\Leftrightarrow \quad c^2 + c\,\frac{2\varepsilon_2}{1-2\varepsilon_2} + \frac{1}{\varepsilon_1(1-2\varepsilon_2)} \stackrel{!}{=} 0$$

$$c = -\frac{\varepsilon_2}{1-2\varepsilon_2} \pm \sqrt{\left(\frac{\varepsilon_2}{1-2\varepsilon_2}\right)^2 - \frac{1}{\varepsilon_1(1-2\varepsilon_2)}}$$

Wenn die Lösung der obigen quadratischen Gleichung komplex wird, so existieren keine reellen Nullstellen, also auch keine Pole.
Dies ist dann der Fall, wenn

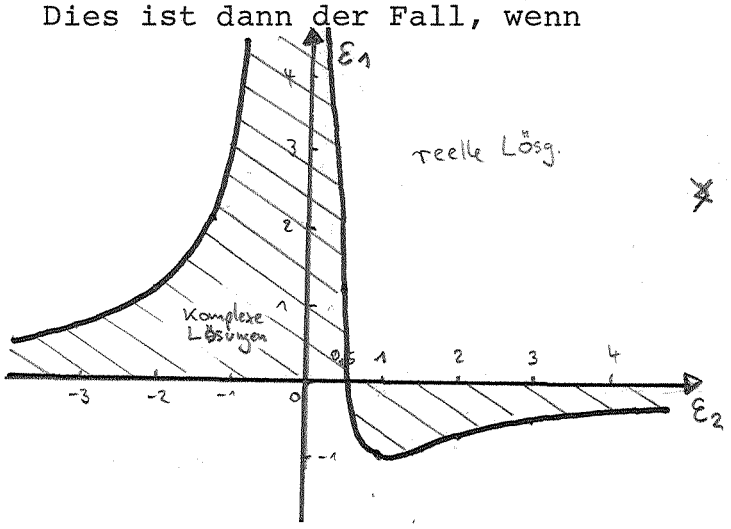


Abb. 8.7

$$\left(\frac{\varepsilon_2}{1-2\varepsilon_2}\right)^2 - \frac{1}{\varepsilon_1(1-2\varepsilon_2)} \stackrel{!}{<} 0$$

$$\Leftrightarrow \quad \frac{\varepsilon_1\varepsilon_2^2 - (1-2\varepsilon_2)}{\varepsilon_1(1-2\varepsilon_2)^2} \stackrel{!}{<} 0$$

Mit $\varepsilon_1 > 0$

ist $\varepsilon_1\varepsilon_2^2 \stackrel{!}{<} (1-2\varepsilon_2)$

$$\varepsilon_1 \stackrel{!}{<} \frac{1-2\varepsilon_2}{\varepsilon_2^2} := g(\varepsilon_2)$$

Mit $\varepsilon_1 < 0$

ist $\varepsilon_1 \stackrel{!}{>} g(\varepsilon_2)$.

Im Bereich $\varepsilon_2 > 1/2$ existiert kein positives ε_1 und im Bereich $\varepsilon_2 < 1/2$ kein negatives ε_1, die diese Bedingungen erfüllen.
In Abb. 8.7 sind die Punktmengen der $(\varepsilon_1, \varepsilon_2)$ Paare, die die

Ungleichungen erfüllen und damit komplexe Lösungen der Polgleichung erzeugen, als schraffierte Flächen eingezeichnet. Für diese Punktmengen existieren also keine reellen Nullstellen des Nenners der Formel (8.2).

Wie man bemerkt, ist die Funktion $f(\varepsilon_2)$ der Abb.8.4 mit $g(\varepsilon_2)$ von Abb.8.7 identisch.
An den Wertepaaren von ε_1 und ε_2, die die Bedingung $\varepsilon_1=f(\varepsilon_2)$ erfüllen, sind also zwei Veränderungen zu beobachten:
Zum Einen treten symmetrisch um $c=c_3^*$ zwei Pole auf, die innerhalb von $[0,1]$ liegen, zum Anderen wechselt die Stabilität der Fixpunkte. Da außerdem nach Voraussetzung $-1 \ll \varepsilon_1 \ll 1$ sein soll, ist es sinnvoll, für das Modell nur Werte von ε_1 zuzulassen, die wenig von Null verschieden sein sollen, so daß die im $\varepsilon_1,\varepsilon_2$-Bereich der reellen Lösungen auftretenden Pole der Funktion $c(n+1)=g(c(n),\varepsilon_1,\varepsilon_2)$ außerhalb des biologisch sinnvollen Bereichs $[0,1]$ fallen.
Auch muß der Bereich der ε_2 so gewählt werden, daß bei $\varepsilon_2 < 1/2$ $\varepsilon_1 < f(\varepsilon_2)$ und bei $\varepsilon_2 > 1/2$ $\varepsilon_1 > f(\varepsilon_2)$ gilt.
Damit ergibt sich für den Geltungsbereich des Modells ein $\varepsilon_1,\varepsilon_2$-Gebiet mit den folgenden stabilen Fixpunkten:

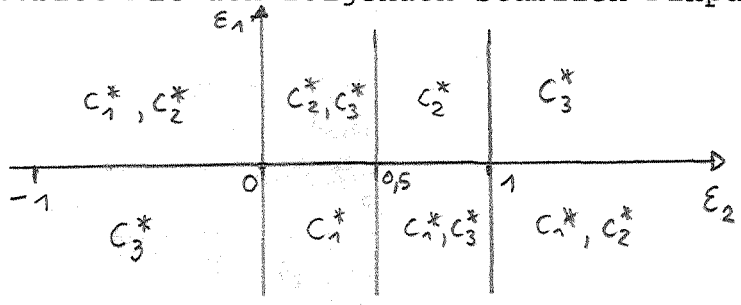
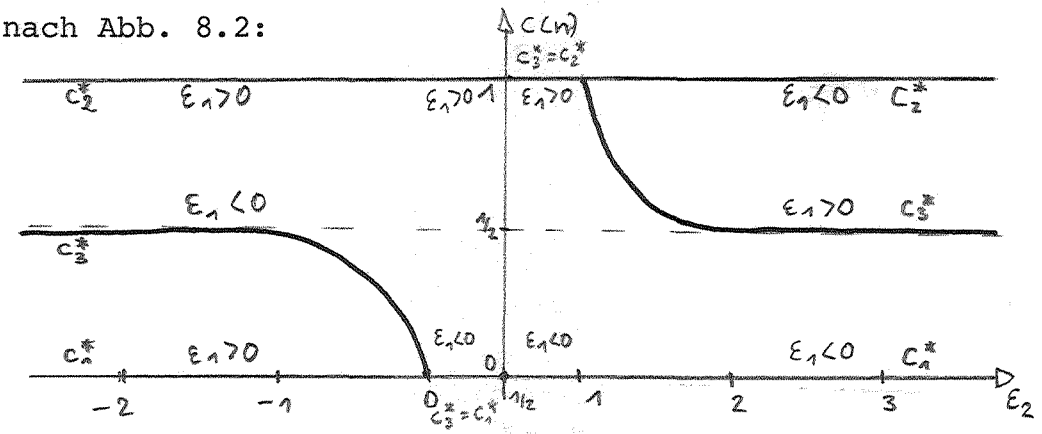


Abb. 8.8 Fixpunkte im Geltungsbereich des Modells

Dabei definiert man für $0 < \varepsilon_2 < 1/2$, wenn $c_3^* < 0$ ist, $c_3^* := 0 = c_1^*$.
Ebenso ist bei $1/2 < \varepsilon_2 < 1$, wenn $c_3^* > 1$ ist, $c_3^* := 1 = c_2^*$.

Damit ergibt sich das Diagramm der stabilen Fixpunkte nach Abb. 8.2:

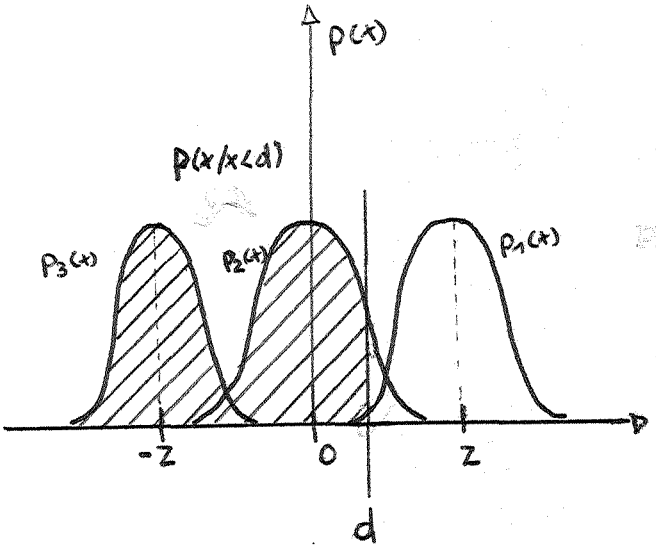


Für die ε_1 und ε_2, für die c_3^* nur labiler Fixpunkt ist, teilt c_3^* das Intervall $[0,1]$ in zwei Teile: Alle c(n), die kleiner als c_3^*, haben c_1^* als Fixpunkt; für $c(n) > c_3^*$ ist c_2^* Fixpunkt.

Anhang A

Der Erwartungswert einer abgeschnittenen Verteilung von drei überlagerten Normalverteilungen

---



$$p(x)=p_1(x)+p_2(x)+p_3(x)$$

$$p_1(x)=\frac{A}{\sqrt{2\pi\sigma}}\cdot\exp\left(-\frac{(x-z)^2}{2\sigma}\right)$$

$$p_2(x)=\frac{B}{\sqrt{2\pi\sigma}}\cdot\exp\left(-\frac{x^2}{2\sigma}\right)$$

$$p_3(x)=\frac{A}{\sqrt{2\pi\sigma}}\cdot\exp\left(-\frac{(x+z)^2}{2\sigma}\right)$$

$$E(x,x<d)=\frac{\int_{-\infty}^{d} x\cdot(p_1(x)+p_2(x)+p_3(x))\,dx}{\int_{-\infty}^{d} p_1(x)+p_2(x)+p_3(x)\,dx}$$

Zähler und Nenner des Bruchs rechne ich getrennt aus:

Nenner:

Das Integral wird in eine Summe von dreien zerlegt und das Argument der Exp.Funktion in jedem substituiert.

Mit $y=\frac{x-z}{\sqrt{\sigma}}$ $dy=\frac{dx}{\sqrt{\sigma}}$ obere Grenzen: $g_1=\frac{d-z}{\sqrt{\sigma}}$

$y=\frac{x}{\sqrt{\sigma}}$ $dy=\frac{dx}{\sqrt{\sigma}}$ $g_2=\frac{d}{\sqrt{\sigma}}$

$y=\frac{x+z}{\sqrt{\sigma}}$ $dy=\frac{dx}{\sqrt{\sigma}}$ $g_3=\frac{d+z}{\sqrt{\sigma}}$

ist $\frac{1}{\sqrt{2\pi}}\left\{A\int_{-\infty}^{g_1}\exp(-y^2/2)\,dy+B\int_{-\infty}^{g_2}\exp(-y^2/2)\,dy+A\int_{-\infty}^{g_3}\exp(-y^2+2)\,dy\right\}$

$$=A\,\Phi\left(\frac{d-z}{\sqrt{\sigma}}\right)+B\,\Phi\left(\frac{d}{\sqrt{\sigma}}\right)+A\,\Phi\left(\frac{d+z}{\sqrt{\sigma}}\right)$$

mit der Gausschen Fehlerfunktion $\Phi(x)=\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{x}\exp(-t^2/2)\,dt$

Beim Zähler wird ebenso verfahren:

$y = \frac{x-z}{\sqrt{\sigma}}$ $dx = dy\sqrt{\sigma}$ obere Grenzen: $g_1 = \frac{d-z}{\sqrt{\sigma}}$

$y = \frac{x}{\sqrt{\sigma}}$ $dx = dy\sqrt{\sigma}$ $g_2 = \frac{d}{\sqrt{\sigma}}$

$y = \frac{x+z}{\sqrt{\sigma}}$ $dx = dy\sqrt{\sigma}$ $g_3 = \frac{d+z}{\sqrt{\sigma}}$

so dass

$$\text{Zähler} = \frac{1}{\sqrt{2\pi}} \left[A\int_{-\infty}^{g_1} y\exp(-y^2/2)dy\,\sqrt{\sigma} + A\int_{-\infty}^{g_1} z.\exp(-y^2/2)dy + B\int_{-\infty}^{g_2} y.\exp(-y^2/2)dy\,\sqrt{\sigma} + A\int_{-\infty}^{g_3} y.\exp(-y^2/2)dy\,\sqrt{\sigma} - A\int_{-\infty}^{g_3} z.\exp(-y^2/2)dy \right]$$

$$= -A\frac{\sqrt{\sigma}}{\sqrt{2\pi}}\exp(-y^2/2)\Big|_{-\infty}^{g_1} + Az\,\Phi(g_1) - B\frac{\sqrt{\sigma}}{\sqrt{2\pi}}\exp(-y^2/2)\Big|_{-\infty}^{g_2} - A\frac{\sqrt{\sigma}}{\sqrt{2\pi}}\exp(-y^2/2)\Big|_{-\infty}^{g_3} - Az\,\Phi(g_3)$$

und mit $\psi(x) = \frac{\sqrt{\sigma}}{\sqrt{2\pi}}\exp(-x^2/2)$

ist der Zähler

$$= Az\,\Phi(g_1) - Az\,\Phi(g_3) - A\,\psi(g_1) - A\,\psi(g_3) - B\Phi(g_2)$$

und der Erwartungswert

$$E(x, x<d) = \frac{A\left(z.\Phi\left(\frac{d-z}{\sqrt{\sigma}}\right) - z.\Phi\left(\frac{d+z}{\sqrt{\sigma}}\right) - \psi\left(\frac{d-z}{\sqrt{\sigma}}\right) - \psi\left(\frac{d+z}{\sqrt{\sigma}}\right) \right) - B\,\Phi\left(\frac{d}{\sqrt{\sigma}}\right)}{A\left(\Phi\left(\frac{d-z}{\sqrt{\sigma}}\right) + \Phi\left(\frac{d+z}{\sqrt{\sigma}}\right) \right) + B\,\Phi\left(\frac{d}{\sqrt{\sigma}}\right)}$$

Mit der Beziehung

$$\int_{-\infty}^{d} p(x)dx + \int_{d}^{\infty} p(x)dx = 1 \text{ ,also } \int_{d}^{\infty} p(x)dx = 1 - \int_{-\infty}^{d} p(x)dx$$

und $$\int_{-\infty}^{d} xp(x)dx + \int_{d}^{\infty} xp(x)dx \overset{p(x)=p(-x)}{=} 0 \text{ ,also } \int_{d}^{\infty} xp(x)dx = -\int_{-\infty}^{d} xp(x)dx$$

ist

$$E(x, x>d) = \frac{-A.\left(\Phi(g_1).z - \Phi(g_3).z - \psi(g_1) - \psi(g_3)\right) + B.\Phi(g_2)}{1 - A.(\Phi(g_1) + \Phi(g_3)) - B.\Phi(g_2)}$$

Anhang B Auswahl der benutzten Computerprogramme

1) Der stochastische Algorithmus

```
10 REM  SIMULATION DES STOCHASTISCHEN LERNALGORHYTHMUS     K=#DIMENSIONEN
20 REM  mit Z1 Zentren P(k,z) und Z6 Startwerten C(k,z),   Z=#ZENTREN
30 REM  wobei das Zentrum P(1,1) in der Amplitude veraenderlich ist.
40 REM  (EINDIMENSIONAL)
50 REM
70 REM
90 DIM X0(31),P1(31)
100 Z1=3\Z6=2\N=100\REM>>>>>>>SIMULATION DER BIFURKATION <<<<<<<
110 K=1\D=1
200 G1=.8\G2=.8\REM ....KOEFFIZIENTEN ALFA UND BETA
250 REM
310 OPEN 'ZYPB2X.DTO' FOR OUTPUT AS FILE #1
320 OPEN 'XLIN.DAT' FOR INPUT AS FILE #2
330 FOR I=0 TO 30\INPUT #2:X0(I)\NEXT I
340 CLOSE #2
345 RANDOMIZE
350 P(1,1)=0
370 B=.5
380 B1=(1-B)/(Z1-1)
390 REM***************ES FOLGT DIE ITERATIONSSCHLEIFE
400 FOR I1=30 TO 0 STEP -1
405 PRINT I1
410 P(1,2)=-X0(I1)\P(1,3)=X0(I1)
420 FOR W=1 TO 20
430 FOR Z=1 TO Z6\J(Z)=1\NEXT Z
440 C(1,1)=0\C(1,2)=0
450 REM
460 FOR J=1 TO N
470 GOSUB 700
480 FOR Z=1 TO Z6\F(Z)=0
490 F(Z)=F(Z)+(X(K)-C(K,Z))^2/2
500 NEXT Z
510 REM-------GEBIETSSUCHE
520 H=1\FOR Z=2 TO Z6
530 IF F(H)<F(Z)GO TO 550
540 H=Z
550 NEXT Z
560 REM----------ITERATION
570 C(K,H)=C(K,H)+G1*(X(K)-C(K,H))/J(H)^G2
580 J(H)=J(H)+1
600 NEXT J
610 D1=(C(1,1)+C(1,2))/2
620 PRINT #1:D1
640 NEXT W
650 NEXT I1
660 CLOSE
670 REM*****************************************
680 END
690 REM-------GAUSSVARIABLE UM DIE ZENTREN
700 X=RND
710 FOR Z=1 TO Z1
720 IF X>((Z-1)*B1+B) THEN 730 \GO TO 740
730 NEXT Z
740 X(K)=0
745 FOR I=1 TO 12
750 X(K)=RND+X(K)-.5
760 NEXT I
770 X(K)=X(K)+P(K,Z)
780 RETURN
```

2) Iteration mit dem Mittelwert

```
50 REM  SIMULATION DES ZYPKIN'SCHEN LERNALGORHYTHMUS      K=#DIMENSIONEN
60 REM  mit Z1 Zentren P(k,Z1) und Z6 Startwerten C(k,Z6),
70 REM  wobei das Zentrum P(1,1) in der Amplitude veraenderlich ist.
80 REM  (EINDIMENSIONALE VERSION)
90 REM Es wird MIT DEM ERWARTUNGSWERT iteriert
100 REM  Ein Gaussches X(k) wird erzeugt und in eine Klasse eingeordnet.
110 REM  Aus den X der Klasse wird der neue Erwartungswert von C(k,z)
120 REM  und die neue Grenze Dn+1 ,daraus wiederum die
130 REM  korrigierten C(k,z) errechnet.
220 REM ------------------------------------------------------------
230 DIM Z(100)
234 DIM X0(30)
240 Z1=3\Z6=2\N=100
250 K=1\D=1
260 P(1,1)=0
270 RANDOMIZE
280 B=.5
290 B1=(1-B)/(Z1-1)
320 OPEN 'XLIN.DAT' AS FILE #7
340 FOR I=0 TO 30\INPUT #7:X0(I)\NEXT I
350 CLOSE #7
380 OPEN 'ZYPB2E.DT0' FOR OUTPUT AS FILE #2
400 REM***************ES FOLGT DIE ITERATIONSSCHLEIFE
420 FOR I1=30 TO 0 STEP -1
430 P(1,2)=-X0(I1)\P(1,3)=X0(I1)
440 FOR W=0 TO 20
450 FOR Z=1 TO Z6\J(Z)=1\NEXT Z
460 C(1,1)=0\C(1,2)=0
470 REM-----------------------------
480 FOR J=1 TO N
490 REM GRENZE ZWISCHEN DEN KLASSEN
500 D1=.5*(C(1,1)+C(1,2))
510 GOSUB 670
520 IF X>D1 THEN 530 \H=1\GO TO 540
530 H=0
540 REM
550 GOSUB 780
570 NEXT J
580 REM-----------------------------
590 PRINT #2:D1
600 NEXT W
620 NEXT I1
630 CLOSE
640 REM*************************************
650 END
660 REM--------------------GAUSSVARIABLE UM DIE ZENTREN
670 X=RND
680 FOR Z=1 TO Z1
690 IF X>(Z-1)*B1+B THEN 700 \GO TO 710
700 NEXT Z
710 X=0
720 FOR I=1 TO 12
730 X=RND+X-.5
740 NEXT I
750 X=X+P(K,Z)
760 Z(J)=X
770 RETURN
780 REM--------------------ERWARTUNGSWERT DER GAUSSVARIABLEN
790 REM NEUER PROTOTYP
795 N(H)=N(H)+1
800 C(1,H)=(C(1,H)*N(H)+X)/(N(H))
810 REM NEUE GRENZE
820 D1=.5*(C(1,1)+C(1,2))
830 REM NEUER ERWARTUNGSWERT BEIDER KLASSEN
840 N(1)=0\N(2)=0\S1=0\S2=0
850 FOR I=1 TO J
860 IF Z(I)>D1 THEN 870 \S1=S1+Z(I)\N(1)=N(1)+1\GO TO 880
870 S2=S2+Z(I)\N(2)=N(2)+1
880 NEXT I
890 IF N(1)=0 THEN 892 \C(1,1)=S1/N(1)
891 GO TO 893
892 C(1,1)=D1
893 IF N(2)=0 THEN 896 \C(1,2)=S2/N(2)\GO TO 900
896 C(1,2)=D1
900 RETURN
```

3) Iteration mit dem Erwartungswert

```
30 REM  SIMULATION DER BIFURKATION
40 REM
50 REM  ITERATION MIT DEM BERECHNETEN ERWARTUNGSWERT VON
60 REM  DER WAHRSCHEINLICHKEITSVERTEILUNG .
70 REM  Benutzt wird die in 'PHI.DAT'punktweise definierte Fehlerfkt PHI
80 Z1=3\Z6=2
82 N=100
85 DIM P1(31)
90 K=1
100 GOSUB 830 \REM EINLESEN VON PHI.DAT
105 P(1,1)=0
110 B=.5
120 B1=(1-B)/(Z1-1)
130 A=(1-B)/2
140 S=1\REM >>SIGMA=1<<
150 S0=1/SQR(S)
160 S1=SQR(S/(2*3.1416))
190 G1=.8\G2=.8\REM Koeff. alfa und bet
198 REM
200 OPEN 'XLIN' FOR INPUT AS FILE #1
210 DIM X0(30)
220 FOR I=0 TO 30\INPUT #1:X0(I)\NEXT I
230 CLOSE
290 OPEN 'ZYPB2T.DTO' FOR OUTPUT AS FILE #1
300 REM***************ES FOLGT DIE ITERATIONSSCHLEIFE
310 FOR I1=30 TO 0 STEP -1
312 P(1,2)=-X0(I1)\P(1,3)=X0(I1)
320 FOR W=0 TO 20
330 IF W>6 THEN 335 \C(1,1)=-1\C(1,2)=-1\GO TO 350
335 IF W>13 THEN 340 \C(1,1)=0\C(1,2)=0\GO TO 350
340 C(1,1)=3\C(1,2)=3
350 REM----------------
370 FOR J=1 TO N
380 GOSUB 580
500 NEXT J
505 REM----------------
510 D1=(C(1,1)+C(1,2))*.5
520 PRINT #1:D1
530 NEXT W
540 NEXT I1
550 CLOSE
560 REM*****************************************
570 END
580 REM-------------ERWARTUNGSWERT DER GAUSSVARIABLEN
590 D=S0*(C(1,1)+C(1,2))/2
600 C1=D-P(1,1)
610 C2=D-P(1,2)
620 C3=D-P(1,3)
630 C=C1\GOSUB 741 \F1=F
640 C=C2\GOSUB 741 \F2=F
650 C=C3\GOSUB 741 \F3=F
660 C0=C1\GOSUB 750 \P1=P
670 C0=C2\GOSUB 750 \P2=P
675 C0=C3\GOSUB 750 \P3=P
680 X1=B*(P(1,1)*P1-F1)+A*(P(1,2)*P2-F2+P(1,3)*P3-F3)
690 X2=B*P1+A*(F2+P3)
700 IF X2=1 THEN 720
710 X=-X1/(1-X2)\REM X IST AUS KLASSE 2
715 C(1,2)=X
720 IF X2=0 THEN 740
730 X=X1/X2\REM X IST AUS KLASSE 1
735 C(1,1)=X
740 RETURN
741 REM-------------FUNKTION F
742 IF ABS(C)<7 THEN 743 \F=0\GO TO 749
743 F=S1*EXP(-(C)^2/2)
749 RETURN
750 REM -------------BERECHNUNG DER FEHLERFUNKTION PHI(C0)
760 IF C0<3 THEN 770 \P=1\RETURN
770 IF C0>=0 THEN 810
780 IF C0>-3 THEN 790 \P=0\RETURN
790 C0=-C0\GOSUB 810
800 P=1-P\RETURN
810 P=P0(C0*50)
820 RETURN
830 OPEN "PHI.DAT" FOR INPUT AS FILE #1
835 DIM P0(150)
840 FOR I=0 TO 150
850 INPUT #1:P0(I)
860 NEXT I
870 CLOSE
880 RETURN
```

Liste der benutzten Literatur

1 Bauer , Wahrscheinlichkeitstheorie
Walter de Gruyter Verlag, Berlin 1968

2 Blomer et al.
A locally sensitive mapping
Medical Data Processing
Taylor&Francis Ltd., London

3 R.Duda,P.Hart
Pattern Classification and Scene Analysis
John Wiley & Sons, New York 1973

4 K.S.Fu
Sequential Methods in Pattern Recognition
Academic Press, New York 1968

5 K.Fukunaga
Introduction to statistical Pattern Recognition
Academic Press, New York 1972

6 Jahnke,Emde,Lösch
Tafeln höherer Funktionen
Teubner Verlagsgesellschaft ,Stuttgart 1966

7 Knopp
Infinite Sequentials and Series
Dover Publications, New York 1956

8 Ludwig
Stochastic Population Theories
Lecture Notes in Biomathematics,
Springer Verlag, Berlin 1974

9 Tou,Gonzales
Pattern Recognition Principles
Addison-Wesley Publishing Company, 1974

10 Tsypkin
Adaption and Learning in Automatic Systems
Academic Press, New York 1971

11 Tsypkin
Foundations of the Theorie of Learning Systems
Academic Press, New York 1973

12 Wilson-Bossert
Einführung in die Populationsbiologie
Springer Verlag, Berlin 1973

13 Zurmühl
Praktische Mathematik
Springer Verlag, Berlin 1965

Danksagung

An dieser Stelle möchte ich mit besonderer Dankbarkeit Herrn Prof.Dr. E.Pfaffelhuber erwähnen,der bis zu seinem tödlichen Verkehrsunfall im Juli 1976 dem von ihm gegebenen Arbeitsthema entscheidende Ideen gegeben hat. Von ihm stammen die Grundideen der Kapitel 1.3,7.0 und 8.0.

Ebenso möchte ich Herrn Prof. M.Dal Cin danken, der die weitere Betreuung dieser Arbeit übernommen hatte und in vielen fruchtbaren Diskussionen zusammen mit Herrn Dr. E.Dilger viel für das Zustandekommen dieser Arbeit beigetragen hat.

Auch Herrn Prof. Güttinger sei für die Unterstützung durch das Institut für Informationsverarbeitung gedankt.