Die Transformation experimenteller Verteilungen durch eine Self-Organizing Feature Map

Alfred Ultsch[1], Günter Halmans[1], Kira Schulz[2]

[1]Universität Dortmund, Abteilung Informatik, Postfach 500 500, 4600 Dortmund 50
[2]Gabelsbergerstr. 60, 8000 München 2

In vielen Fällen entspricht die Verteilung von empirisch erhobenen Daten nicht einer Normalverteilung. Um eine vergleichbare Skalierung der Daten zu erreichen, ist eine Transformation in eine Normalverteilung oder zumindest in eine symmetrische Verteilung notwendig. Desweiteren basieren viele statistische Verfahren auf der Annahme einer Normalverteilung. Die Bestimmung einer geeigneten Transformation beinhaltet typischerweise einen "trial and error" Prozeß oder benötigt die Erfahrung eines Experten. In diesem Bericht wird eine Methode beschrieben, mit der es möglich ist, durch den Einsatz einer Self-Organizing Feature Map den Auswahlprozeß zu automatisieren. Zur Prädiktion einer Transformation wurde der Lernalgorithmus der Feature Map modifiziert. Erste Ergebnisse haben gezeigt, daß die Feature Map in der Lage ist, die Verteilungen des Trainingsdatensatzes in eine Normalverteilung zu transformieren. Auch für neue Verteilungen, die das System nicht gelernt hat, prädiziert das Modell geeignete Transformationsparameter, wodurch seine Fähigkeit zur Generalisierung deutlich wird.

1. Einleitung

Die Verarbeitung von empirisch erhobenen Daten beinhaltet oftmals das Problem, daß diese Daten üblicherweise nicht einer statistischen Normalverteilung entsprechen. Dadurch sind sie nur schwer vergleichbar und darüberhinaus für parametrische Tests ungeeignet. In vielen Fällen ist es möglich, die Beobachtungen durch eine geeignete Transformation in eine Normalverteilung oder zumindest in eine symmetrische Verteilung zu überführen.

Die Statistik kennt eine Reihe solcher Transformationen [1]. Bei einer besonders ausgeprägten Schiefe findet die inverse Transformation ihre Anwendung. Ähneln die Daten einer Poissonverteilung, so kann diese durch die Wurzeltransformation in eine Normalverteilung überführt werden. Beobachtungen z.B. aus der Bevölkerungsstatistik werden oft mit der Funktion ln() transformiert. Daneben kennt die Statistik u.a. noch die

Box-Cox-Transformation, die Arcus-Sinus oder die Fishersche z-Transformation, die besonders in der Korrelationsrechnung ihre Anwendung findet [1].

Viele der in der Explorativen Datenanalyse angewandten Transformationen sind von der Form x^p und werden durch die sogenannte "ladder of power" charakterisiert [1]. Die Wahl eines geeigneten Exponenten ist dabei nicht trivial, vielmehr unterliegt sie dem "trial and error" Verfahren [5] oder der Erfahrung eines Experten. Es stellt sich daher die Aufgabe, die Wahl dieses Exponenten zu automatisieren.

Dieser Bericht zeigt eine Möglichkeit, wie durch den Einsatz einer Self-Organizing Feature Map [2] eine solche Transformation gefunden werden kann. Im folgenden wird davon ausgegangen, daß das Modell der Self-Organizing Feature Map in seinen Grundzügen bekannt ist und es werden nur eventuelle Abweichungen von diesem Modell erläutert (siehe Kapitel 3).

2. Datentransformation in der Explorativen Statistik

Um vorliegende Beobachtungen und deren Verteilung beurteilen und eine eventuelle Transformation bestimmen zu können, bedarf es Parameter zur Beschreibung dieser Verteilung. Empirische wie theoretische Verteilungen werden u.a. durch Parameter wie die Lage, die Streuung, die Quartile, die Perzentile, die Schiefe, den Exzeß oder den Variationskoeffizienten charakterisiert. Bei einer Normalverteilung sind beispielsweise die drei Lageparameter Median, arithmetischer Mittelwert wie auch der Modus aufgrund der Symmetrieeigenschaft identisch. Die Symmetrie um den Mittelwert impliziert auch, daß keine rechts- oder linksschiefe Verteilung vorliegt [1].

Exponent p	... 3 ...	2 ...	1 ...	0,5 ...	ln ...	-0,5 ...	-1 ...	-2 ...
Transformation	... x^3...	x^2...	x ...	$\sqrt{x}$ ...	ln(x) ...	$1/\sqrt{x}$ ...	1/x ...	$1/x^2$...

Verteilungsform: linksschief<---> symmetrisch <----> expon.<---------> rechtsschief

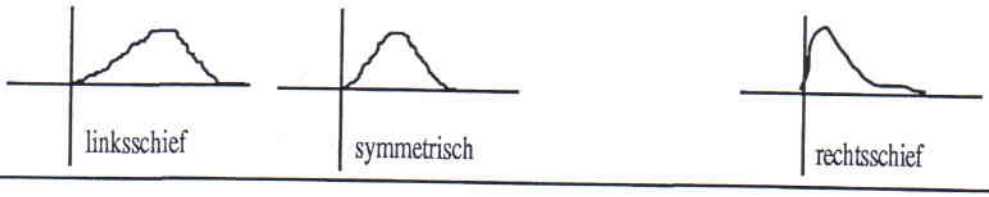

Abbildung1: Die "ladder of power"

Die Transformation zur Überführung eines Datensatzes in eine Normalverteilung ist vielfach von der Form x^p. Die sogenannte "ladder of power" kennzeichnet die Eigenschaften der Transformationen [1]. Abbildung 1 zeigt, welche Potenztransformationen auf die verschiedenen Formen der Verteilungen angewandt werden. Bei einer rechtsschiefen Verteilung muß der Exponent $p < 1$ sein, bei einer linksschiefen Verteilung sollte $p > 1$ gewählt werden. Die ln-Transformation kann in die Reihe der Transformationen an der

Stelle p = 0 eingefügt werden. Für negative Exponenten p wird die Ordnung der Daten umgekehrt, daher wird in diesem Falle oft die Transformation $-(x+c)^p$ gewählt. Die Schwierigkeit, negative Werte zu transformieren, wird durch Addition einer Konstanten umgangen.

Aus der "ladder of power" ist zwar ablesbar, welcher Exponent zu welcher Schiefe paßt, jedoch gibt sie keine Transformation für eine vorliegende Beobachtungsreihe direkt an. Die Auswahl eines geeigneten Exponenten ist ein "trial and error" Verfahren [5]. Oft ermöglicht erst die Erfahrung von vielen Verteilungen mit verschiedenen Formen und den dazugehörigen notwendigen Exponenten eine geeignete Auswahl.

Ein Schnellverfahren zur Suche der Transformation benutzt einen p-Quantilskoeffizienten der Schiefe [5]:

$$g_q^{(p)} = \frac{(x_{1-q}^p - \tilde{x}^p) - (\tilde{x}^p - x_q^p)}{x_{1-q}^p - x_q^p}$$

p: Transformation q : Quantile $\tilde{x}$: Median

Bei symmetrischen Verteilungen nehmen die Quantilskoeffizienten den Wert Null an. Bei rechtsschiefen Verteilungen sind sie größer, bei linksschiefen Verteilungen kleiner als Null.

Bei einer gegebenen Verteilung wird der p-Quantilskoeffizient für verschiedene Exponenten berechnet. Ist er gleich Null, so ist der optimale Exponent gefunden. Eine Überprüfung der Verteilung der transformierten Werte kann durch die Analyse der Q/Q-Plots erfolgen [1]. Obwohl der Rechenaufwand für die Berechnung des p-Quantilskoeffizienten gering ist, bedeutet er vor allem bei der Betrachtung vieler verschiedener Verteilungen einen erheblichen Zeitverlust.

3. Das modifizierte Modell der Self-Organizing Feature Map

Durch den Einsatz eines konnektionistischen Modells, der Self-Organizing Feature Map [2], haben wir versucht, eine Automatisierung zur Bestimmung eines geeigneten Exponenten zu erreichen. Konnektionistische Modelle wie die Self-Organizing Feature Map zeichnen sich durch ihre Generalisierungsfähigkeit sowie durch ihre Fähigkeit zur "graceful degradation" aus, d.h. sie sind in der Lage, auch mit unvollständigen Daten umzugehen und die passendste Ausgabe zu einer Eingabe zu generieren [4][7][3]. Vor allem letztere Eigenschaft macht sich die hier vorgestellte Methode zunutze. Bei den in diesem Bericht vorgestellten Experimenten haben wir eine zweidimensionale Feature Map mit einer Größe von 32 x 32 Units verwandt. Die Eingabevektoren waren 13-dimensional: Die 9 Perzentile einer Verteilung sowie die Schiefe, der Exzeß, der Variationskoeffizient sowie der für die Transformation notwendige Exponent (siehe Abbildung 2).

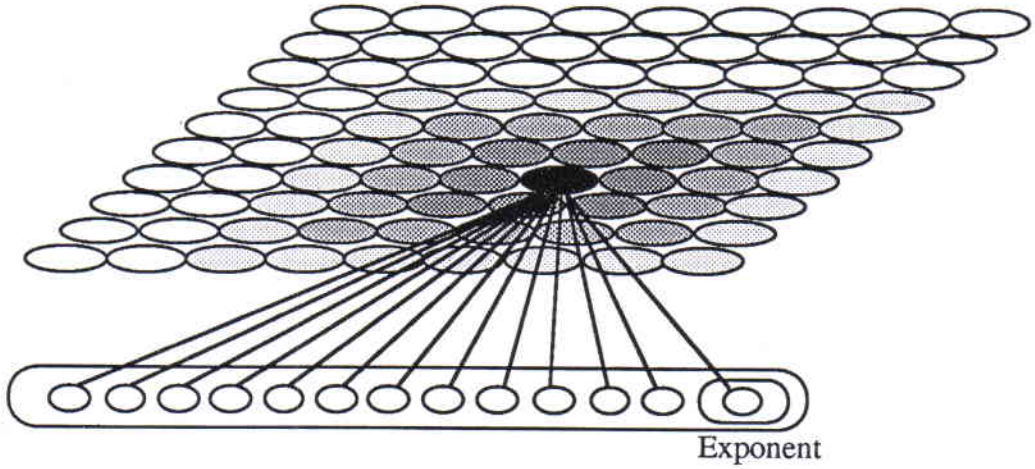

Abbildung 2: Das modifizierte Kohonen Modell

Die Lernphase des hier angewendeten Modells differiert von Kohonens Modell der Self-Organizing Feature Map [2]. In dem hier vorgestellten Modell wird für den Ordnungsprozeß der Feature Map die letzte Komponente ausgespart. D.h. zur Bestimmung der Unit, deren Gewichtsvektor dem Eingabevektor am ähnlichsten ist, werden nur die ersten zwölf Komponenten zur Berechnung der Euklidischen Distanz herangezogen. Damit erfolgt die Ordnung der Map ausschließlich nach den Charakteristika der Verteilungen. In der Adaptionsphase werden alle Komponenten der Gewichtsvektoren - einschließlich des Exponenten - in der Nachbarschaft der Gewinner-Unit dem Eingabevektor angepaßt. Mit dieser Modifikation wird ein überwachtes Lernen der Feature Map ermöglicht.

In der Arbeitsphase wird eine Verteilung, beschrieben durch die 9 Perzentile, der Schiefe, den Exzeß und den Variationskoeffizienten angelegt. Das Netz ermittelt aufgrund der zwölf Komponenten die Unit mit dem im Sinne des Euklidischen Abstands nächsten Gewichtsvektor. Ist diese Unit gefunden, kann das Netz die dreizehnte Komponente vervollständigen und somit eine plausible Aussage über den für diese Verteilung notwendigen Exponenten machen. Aufgrund der Eigenschaft zur Generalisierung ist das Netz in der Lage, Eingabevektoren mit der Beschreibung einer Verteilung, die sich nicht im Trainingsdatensatz befinden, zu klassifizieren und zu vervollständigen.

4. Die Trainingsdaten

Um ein überwachtes Lernen durchführen zu können, ist es notwendig, eine Menge von "Trainingsverteilungen" zu generieren, bei denen der für eine Transformation in eine Normalverteilung notwendige Exponent bekannt ist. Mit Hilfe der Approximationsfunktion von Hastings [1] wurde eine möglichst optimale Normalverteilung bestehend aus 100 Werten als Basis für die Trainingsverteilungen generiert. Die Verteilungen wurden durch Potenzieren mit 42 Exponenten, deren reziproken Werte im Bereich von 0.1 bis 14.5 lagen, gemäß der "ladder of power" erzielt. So entstand ein Trainingsset mit 42 verschiedenen Verteilungen unterschiedlicher Schiefe auf der Basis einer Normalverteilung.

Zur Beschreibung der Verteilungen wurden die Werte derselben zunächst z-transformiert. Infolgedessen erhalten alle Verteilungen eine Standardabweichung von 1 und den Mittelwert 0. Die Verteilungen wurden durch 13 Komponenten beschrieben. Die ersten neun Komponenten repräsentieren die Perzentile; die zehnte, elfte und zwölfte Komponente beschreiben die Schiefe, den Exzeß und den Variationskoeffizienten. Die dreizehnte Komponente wiederum beinhaltet den Exponenten.

5. Ergebnisse

Das beschriebene Modell wurde auf einem Transputersystem implementiert [8]. Die Self-Organizing Feature Map wurde in 300000 Lernschritten mit den 42 Trainingsverteilungen angelernt.

Zur Verifikation der Methode haben wir drei Testdatensätze unterschiedlicher Güte generiert. Der erste Testsatz (A) ist identisch mit dem Trainingsdatensatz. Mit der Überprüfung der Prädiktion dieses Datensatzes konnte kontrolliert werden, wie gut das Netz gelernt hatte. Der zweite Testdatensatz (B) besteht aus 30 neuen Verteilungen, die ebenfalls auf der N(0,1) Verteilung aus dem Trainingssatz basieren. Jedoch wurden diese Verteilungen durch die Verwendung nicht trainierter Exponenten generiert. Testsatz (C) schließlich beinhaltet 50 verschiedene Verteilungen, die auf Normalverteilungen basieren, welche mit Hilfe eines Pseudozufallszahlen-Generators erstellt wurden. Diese wurden mit verschiedenen Exponenten transformiert, um unterschiedliche Schiefen und Exzesse zu erhalten. Testsatz (B) und besonders Testsatz (C) überprüfen die Fähigkeit des Netzes zur Generalisierung.

Die transformierten Verteilungen wurden mit den 100 Quantilen der mit der Approximationsfunktion von Hastings generierten Standardnormalverteilung verglichen. Der Wert '100' repräsentiert den Vergleich dieser Verteilung mit sich selbst und stellt damit die maximal erreichbare Ähnlichkeit dar. Die Werte "z" wurden durch die folgenden Gleichungen berechnet:

$$z = \bar{a}_k \; ; \qquad a_k = \left(1 - \frac{x_k - nv_{min}}{\bar{utr} - nv_{min}}\right) * 100 \; ; \qquad x_k = \frac{\Sigma \, | \, diff_i \, |}{90} \; ;$$

$i = 1,...,100$, $k = 1,..., n$, n = Anzahl der Verteilungen,

$\bar{a}_k$ = Mittelwert der a_k aus einem Datensatz,
nv_{min} = Minimum der x_k der 50 Normalverteilungen des Pseudozufallszahlen-Generators,
$\bar{utr}$ = Mittelwert der x_k der untransformierten Verteilungen,
$diff_i$ = Differenz der Quantile der Hastings Normalverteilung und der betrachteten, z-transformierten Verteilung.

Die durchgezogene Linie in Abbildung 3 gibt den Grad der Ähnlichkeit eines Datensatzes von 50 verschiedenen, mit der Hilfe eines Pseudozufallszahlen-Generators erstellten Normalverteilungen an. Eine Genauigkeit im Vergleich zu der Hastings Verteilung von

ca. 60% darf somit als gut angesehen werden. Die gestrichelte Linie zeigt die Ähnlichkeit der untransformierten 50 Verteilungen aus dem Testdatensatz (C) und weist aufgrund der starken Schiefe einiger dieser Verteilungen natürlich einen schlechten Wert auf.

Zum Vergleich der von der Feature Map gelieferten Ergebnisse wurden die Verteilungen der drei Testsätze unter Verwendung des p-Quantilskoeffizienten (siehe Kapitel 2) transformiert. Abbildung 3 macht deutlich, daß diese Transformationen entweder besser oder im Bereich der durch den Zufallszahlen Generator generierten Normalverteilungen liegen.

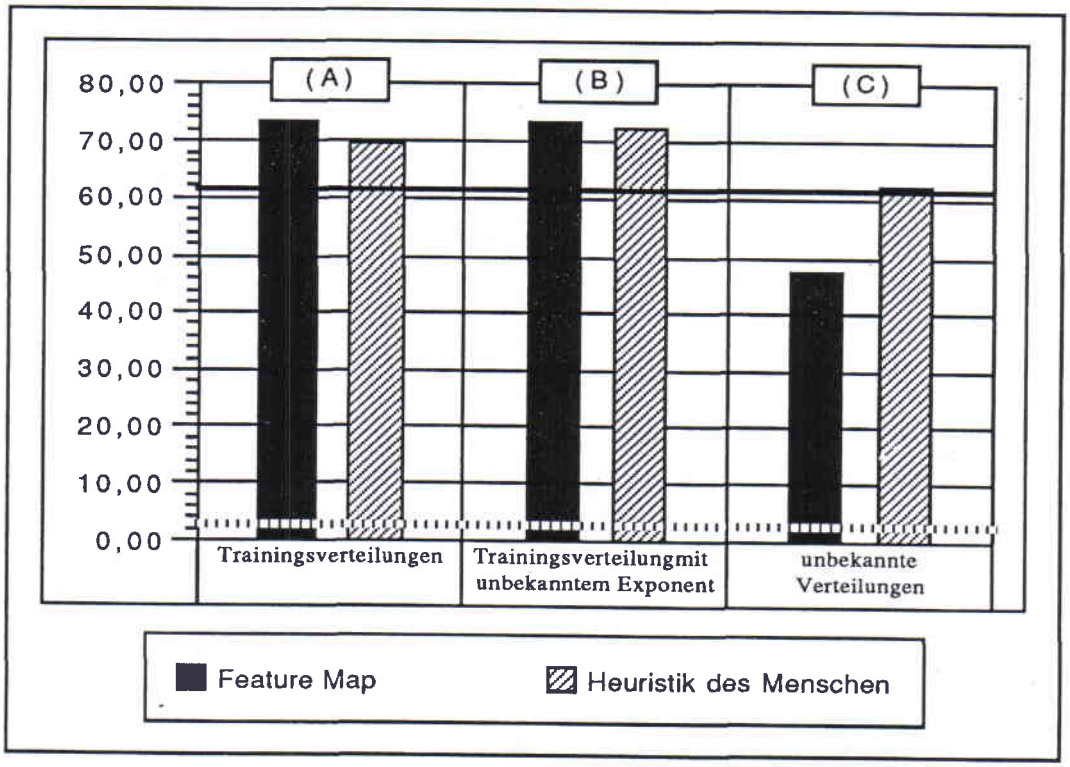

Abbildung 3: Die Self-Organizing Feature Map im Vergleich zu einer Heuristik

Testsatz (A) und Testsatz (B) zeigen, daß die erreichte Ähnlichkeit durch eine von der Feature Map vorgeschlagenen Transformation bei diesen Testverteilungen im Durchschnitt etwas besser als die der p-Quantilskoeffizienten ist. Zudem liegen die Werte oberhalb der von den Normalverteilungen erreichten Ähnlichkeit. Testsatz (B) macht die exzellente Fähigkeit der Feature Map zur Generalisierung deutlich, wenn die zu transformierenden Verteilungen auf die Standardnormalverteilung von Hastings basieren. Bei dem dritten Testdatensatz (C) erreicht auch die Methode des p-Quantilskoeffizenten keine bessere Ähnlichkeit als sie von den Normalverteilungen des Zufallszahlen-Generators vorgegeben wird. Das deutet darauf hin, daß hier nur sehr schwer eine genauere Anpassung an die Hastings Standardnormalverteilung erreicht werden kann. In diesem Zusammenhang ist der Wert der Feature Map von 47.3 als ein gutes Ergebnis zu bewerten. Die Kontrolle mit Hilfe der Q/Q-Plots unterstützt diese Annahme.

6. Abschließender Überblick

In diesem Bericht wird die prinzipielle Möglichkeit aufgezeigt, wie durch den Einsatz einer Self-Organizing Feature Map der "trial and error" Prozeß zur Bestimmung einer geeigneten Datentransformation ersetzt werden kann. Dazu wurden verschiedene Verteilungen unterschiedlichster Formen generiert. Ein gegenüber Kohonens Self-Organizing Feature Map modifiziertes Netz wurde mit diesen durch eine geeignete Beschreibung charakterisierten Verteilungen angelernt.

Die in Kapitel 5 beschriebenen Experimente zeigen, daß die Präzision der Anpassung an eine Normalverteilung, die durch den Einsatz der Feature Map erreicht wird, ähnlich gut oder besser als die des p-Quantilskoeffizienten ist. Diese traditionelle Methode unterliegt jedoch einem "trial and error" Prozeß und erfordert einigen Zeitaufwand oder die Erfahrung eines Statistik Experten. Die in diesem Bericht beschriebene Methode macht sich die Eigenschaften der Self-Organizing Feature Maps zunutze. Neue Verteilungen werden nicht nur mit den gelernten Verteilungen, sondern auch mit ihren Generalisierungen verglichen. So wird eine ähnlichste Verteilung gefunden und der zu dieser generalisierten Verteilung gehörende Exponent ausgegeben.

Erste Ergebnisse haben gezeigt, daß die Feature Map in der Lage ist, die notwendigen Exponenten für die Trainingsverteilungen exakt wiederzugeben. Experimente mit unbekannten Verteilungen machen deutlich, daß das Modell die Fähigkeit zur Generalisierung besitzt und Exponenten mit einer vielversprechenden Präzision schätzen kann. Der Vergleich mit einer vom Menschen eingesetzten Heuristik zeigt, daß die durch die Feature Map erreichte Güte der Anpassung an eine Normalverteilung in etwa 80% der durch die Heuristik gewonnenen Anpassung beträgt.

Acknowledgement

Diese Arbeit ist in Teilen durch Mittel des Landes Nordrhein-Westfalen im Rahmen des Benningsen-Foerder Forschungsprogrammes gefördert worden.

Literatur

[1] Hartung, J. Statistik. Lehr- und Handbuch der angewandten Statistik. 7. Auflage, Oldenbourg, München 1989

[2] Kohonen, T. Self-Organisation and Associative Memory. Springer Verlag, Berlin 1984

[3] Fanihagh, F.; Lütgendorf, A.; Mempel, M.; Rossbach, P.; Schneider, B.; Wegmann, F. Wissensakquisition für wissensbasierte Systeme mit konnektionistischen Modellen, in [6]

[4] Rumelhardt, D.E.; McClelland J.L. Parallel Distributed Processing: Exploration in the Microstructure of Cognition, Volume 1: Foundations, MIT Press, Cambridge (Massachusetts) 1986

[5] Schlittgen, R. Einführung in die Statistik, 2. Auflage, Oldenbourg, München 1990

[6] Ultsch, A. (Ed.) Kopplung deklarativer und konnektionistischer Wissensrepräsentation. Endbericht der Projektgruppe PANDA, Berichtnummer: 352, Universität Dortmund 1990

[7] Ultsch, A., Halmans, G., Mantyk, R. CONKAT: A Connectionist Knowledge Acquisition Tool. in: Proceedings of the Twenty-Fourth Annual Hawaii International Conference on System Sciences, IEEE Computer Society Press, Los Alamitos, California 1991

[8] Ultsch, A., Siemon, H.P. Kohonen Networks on Transputers: Implementation and Animation, in: Proceedings of the International Neural Network Conference (INNC), Volume 2, Paris 1990