Nichtlineare Hauptkomponentenanalyse auf Basis neuronaler Netze

Diplomarbeit

Matthias Scholz
Humboldt-Universität zu Berlin
Institut für Informatik

Betreuer

Prof. Dr. Klaus-Robert Müller
Fraunhofer Institut FIRST
Arbeitsgruppe IDA
(Intelligente Datenanalyse)

Prof. Dr. Hans-Dieter Burkhard
Humboldt-Universität zu Berlin
Institut für Informatik
Künstliche Intelligenz

Zusammenfassung

Die Hauptkomponentenanalyse, *Principal Component Analysis — PCA*, ist eine weit verbreitete und vielfältig anwendbare Methode der Dimensionsreduktion und der Merkmalsextraktion. Sie wird benutzt zur Komprimierung, zum Entrauschen von Daten oder allgemein als Vorverarbeitung bei Klassifikations-, Regressions- oder Quellentrennungsaufgaben.
Die PCA ist auf die Erkennung linearer Strukturen in Datenräumen beschränkt. Daher gibt es verschiedene Ansätze, eine mächtigere Methode zur Merkmalsextraktion zu entwickeln, welche auch nichtlineare Strukturen erkennen kann.
In dieser Arbeit wird eine nichtlineare PCA auf der Basis eines autoassoziativen neuronalen Netzes untersucht. Es werden die Möglichkeiten, aber auch die Grenzen dieser Netzarchitektur aufgezeigt. Darauf aufbauend wird versucht, eine nichtlineare PCA zu konstruieren, deren Eigenschaften mit denen der linearen PCA weitgehend übereinstimmen.
Anschließend wird diese nichtlineare PCA mit anderen Methoden der nichtlinearen Merkmalsextraktion anhand verschiedener Datensätze aus unterschiedlichen Anwendungsgebieten verglichen.

Abstract

Nonlinear principal component analysis (NLPCA) is known as a nonlinear generalization of the standard principal component analysis (PCA). Since NLPCA is a nonunique concept, it is discussed, how NLPCA can be defined as a nonlinear feature extraction technique most similar in spirit to PCA.
Not only the nonlinear reduction of a data set from its original dimension to the intrinsic dimension of the data is considered, but also the arrangement of the features spanning this intrinsic data space is requested to have an order similar to PCA. Thus, such NLPCA is a powerful preprocessing step. It can be used as nonlinear sphering (whitening) or it can be considered as a smoothing method which removes nonlinear correlations between variables. A suitable method to perform such NLPCA is to minimize a hierarchical error function. This error function can be applied to a multi-layer perceptron which is used in auto-associative mode to perform the identity mapping.

Danksagung

An dieser Stelle möchte ich mich herzlich bei all denen bedanken, die mich bei der Anfertigung dieser Arbeit unterstützt haben.
Besonderer Dank gilt Herrn Prof. Dr. Klaus-Robert Müller und allen anderen Mitgliedern der Arbeitsgruppe IDA (Intelligente Datenanalyse) am Fraunhofer Institut FIRST. Insbesondere bei Ricardo Vigário, Andreas Ziehe, Stefan Harmeling, Gunnar Rätsch, Sebastian Mika, Motoaki Kawanabe, Jens Kohlmorgen, Steven Lemm und Benjamin Blankertz möchte ich mich für zahlreiche Diskussionen und Hinweise bedanken.
Herrn Prof. Dr. Hans-Dieter Burkhard danke ich für die Betreuung von Seiten der Humboldt-Universität zu Berlin.
Auch danke ich dem Fraunhofer Institut FIRST für die Bereitstellung von Rechentechnik und Software und dafür, dass mir der Besuch der internationalen Konferenz ESANN 2002 ermöglicht wurde.
Für die zur Verfügung gestellten Datensätze und für die Hilfe bei der Analyse möchte ich mich bei Jürgen Stock, *Centro de Investigaciones de Astronomía (CIDA)*, Venezuela (Stern-Spektraldatensatz) und bei David T. Mewett, *Flinders University, Australia* (EMG-Daten) bedanken.

Selbständigkeitserklärung

Hiermit bestätige ich, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen Hilfsmittel als angegeben verwendet habe.

Berlin, 8. Mai 2002 Matthias Scholz

Einverständniserlärung

Ich erkläre mich damit einverstanden, dass ein Exemplar dieser Diplomarbeit in der Bibliothek des Instituts für Informatik verbleibt.

Berlin, 8. Mai 2002 Matthias Scholz

Thesen

1. Die klassische nichtlineare Hauptkomponentenanalyse (NLPCA) basierend auf dem Autoencoder [12] kann nur eingeschränkt als nichtlineare Erweiterung der linearen Hauptkomponentenanalyse (PCA) gelten. Die extrahierten Merkmale besitzen keine speziellen Eigenschaften und keine Ordnung. Es existiert nur ein Kriterium an den Unterraum, welcher durch die Merkmale aufgespannt wird. Der Algorithmus ist daher nur als reiner Dimensionsreduktions-Algorithmus einsetzbar.

2. Die Merkmale einer NLPCA sollten die gleichen Eigenschaften aufweisen wie die der PCA, mit der einzigen Ausnahme, dass sie nichtlinear, d.h. gekrümmt sein können. Die wesentlichste Eigenschaft ist, dass die extrahierten Merkmale nichtlinear unkorreliert sind. Dies kann durch eine hierarchische Ordnung der nichtlinearen Merkmale erreicht werden. Dadurch ist eine NLPCA auch im Sinne der Merkmalsextraktion vergleichbar mit der PCA.

3. Eine hierarchische Bedingung ist über den Rekonstruktionsfehler realisierbar. Eine Bedingung an die Varianz oder eine deflationäre Extraktion der Merkmale ist dagegen sehr schwierig, wenn nicht sogar unmöglich.

4. Im nichtlinearen Fall ist die Extraktion der Merkmale nicht immer als stetige Funktion approximierbar. Die Merkmale können aber durch eine Modellierung der inversen Generierungsfunktion extrahiert werden. Der zweite Teil des Autoencoders kann dazu allein optimiert werden, der Extraktionsteil wird dazu nicht benötigt.

Inhaltsverzeichnis

Kapitel 1

Einleitung

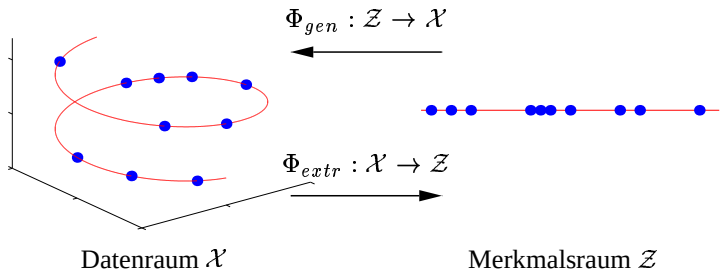


Abbildung 1.1: Die originalen Daten liegen im 3-dimensionalen Raum (links), ihre wahre intrinsische Dimension ist nur 1-dimensional. Die Lage der Daten ist mit einem Parameter bzw. mit einem Merkmalswert exakt beschreibbar. Aus den 1-dimensionalen Merkmalswerten (rechts) können die originalen Daten (links) erzeugt werden, vorausgesetzt, die Generierungsfunktion $\Phi_{gen} : \mathcal{Z} \rightarrow \mathcal{X}$ vom Merkmalsraum $\mathcal{Z}$ in den originalen Datenraum $\mathcal{X}$ ist bekannt.

1.1 Dimensionsreduktion und Merkmalsextraktion

In der Praxis liegen Daten oft in hochdimensionalen Datenräumen vor. So ist beispielsweise bei der Klassifikation handgeschriebener Ziffern auf Grauwertbildern der Auflösung 16 x 16 der Datenraum 256-dimensional.
Die wahre intrinsische Dimension der Daten ist aber häufig geringer als die Dimension der gegebenen originalen Daten. Die Daten können in diesen Fällen ohne großen Informationsverlust auf weniger Dimensionen komprimiert werden. Sie sind mit einer geringeren Anzahl von Parametern beschreibbar. Diese Parameter werden als Merkmale bezeichnet und können im originalen Datenraum eine Kurve beschreiben, siehe Abbildung 1.1. Der reduzierte Datenraum wird als Merkmalsraum bezeichnet.
Bei der Dimensionsreduktion werden die Daten eines d-dimensionalen Raumes

auf einen m-dimensionalen Unterraum, den Merkmalsraum, projiziert, $d > m$. Dies entspricht einer Abbildung $\Phi : \mathcal{X} \rightarrow \mathcal{Z}$ vom originalen Datenraum $\mathcal{X}$ in den Merkmalsraum $\mathcal{Z}$.

Eine Dimensionsreduktion kann als Vorverarbeitung bei Regressions- oder Klassifikationsalgorithmen zu einer verbesserten Leistung führen. Dies erscheint vielleicht widersinnig, da keine Informationen dazukommen, im Gegenteil, in den meisten Fällen sogar Informationen verlorengehen. Der Grund hierfür liegt im *curse of dimensionality*, wonach die Schwierigkeit eines Lernproblems bei gleichbleibend vielen Daten mit der Dimensionalität des Raumes wächst.

Was relevante Informationen sind, und wo sie zu finden sind, hängt entscheidend von dem eigentlich zu lösenden Problem ab, der Klassifikation oder Regression. Oft ist aber eine Dimensionsreduktion ohne zusätzliches Wissen gewünscht bzw. erforderlich. Dazu können verschiedene Annahmen gemacht werden.
Eine häufige Annahme bei der Dimensionsreduktion ist, dass der Informationsgehalt in direktem Zusammenhang mit der Varianz der Daten steht. Große Varianz ist gleichzusetzen mit viel Informationen, kleine Varianz entspricht vernachlässigbarer Information oder beschreibt nur das Rauschen der Daten. Die Projektion der Daten auf einen Unterraum, unter Erklärung größtmöglicher Varianz, kann zum Entrauschen der Daten benutzt werden. Der Unterraum bzw. Merkmalsraum stellt eine Komprimierung der Daten dar.
Eine der wichtigsten Methoden zur Dimensionsreduktion mit der Forderung maximaler Varianz im Merkmalsraum ist die in dieser Arbeit untersuchte Hauptkomponentenanalyse, die PCA — *Principal Component Analysis* [5].

Steht bei einer Methode der reduzierte Merkmalsraum im Vordergrund, wird sie als Methode der Dimensionsreduktion bezeichnet. Wie die Merkmale den Merkmalsraum beschreiben, ist dabei von geringerer Bedeutung. Steht die Suche nach Merkmalen mit bestimmten Eigenschaften im Vordergrund, wird die Methode der Merkmalsextraktion zugeordnet. Eine Dimensionsreduktion muss dabei nicht zwangsläufig stattfinden. Dimensionsreduktion und Merkmalsextraktion stehen in engem Zusammenhang, unterscheiden sich aber zum Teil in der Zielstellung.
Ziel der Merkmalsextraktion ist es, Merkmale mit bestimmten Eigenschaften zu bestimmen, aus denen wie bei der Dimensionsreduktion die originalen Daten generiert werden können. Eine der wichtigsten Eigenschaften ist die Unabhängigkeit der Merkmale: aus Merkmalswerten eines Merkmals lassen sich keine Merkmalswerte eines anderen Merkmals ableiten. Diese Eigenschaft spielt in der Quellentrennung eine große Rolle. Hierbei wird versucht, aus Mischungen von Signalquellen (z.B.: Sprache, EEG-Signale) die ursprünglichen Signalquellen zu bestimmen. Bei Annahme unabhängiger Signalquellen kann eine Bestimmung unabhängiger Merkmale der Mischungen zum Ziel führen [10].

Auch die PCA kann als eine Methode zur Merkmalsextraktion betrachtet werden. Die PCA-Merkmale haben die wesentliche Eigenschaft, dass sie linear unkorreliert sind. Unkorrelierte Merkmale sind die Grundlage einer erfolgreichen Vorverarbeitungsmethode — dem *sphering* oder *whitening*.
Sphering (Whitening) kann als Vorverarbeitung bei Klassifikations- und Regressionsalgorithmen und als Vorverarbeitung bei der Trennung von Signalquellen eingesetzt werden.

Die PCA ist beschränkt auf eine lineare Dimensionsreduktion bzw. lineare Merkmalsextraktion. Die vorliegende Arbeit konzentriert sich auf eine nichtlineare Verallgemeinerung der PCA zu einer NLPCA — *Nonlinear Principal Component Analysis* [25].
Dazu wird eine NLPCA-Methode zur nichtlinearen Dimensionsreduktion auf der Basis neuronaler Netze untersucht — der *Autoencoder* [12]. Diese Methode bestimmt ähnlich der PCA einen nichtlinearen Unterraum maximaler Varianz, stellt aber keine speziellen Anforderungen an die Merkmale, welche diesen Raum beschreiben. Ziel meiner Arbeit ist es, diese Methode weiterzuentwickeln, so dass unkorrelierte nichtlineare Merkmale extrahiert werden. Dadurch ist diese Methode auch im Sinne der Merkmalsextraktion vergleichbar mit der PCA.
Es existieren verschiedene andere Methoden der nichtlinearen Dimensionsreduktion und Merkmalsextraktion, mit denen die vorgeschlagene NLPCA verglichen wird, die *Principal Curves* [8], die *Kern PCA* [24] und *Locally Linear Embedding (LLE)* [19]. Eine Übersicht über verschiedene Methoden der Dimensionsreduktion ist in [4] zu finden.

1.2 Übersicht

Kapitel 2: Lineare PCA und Sphering
Die gut bekannte und weit verbreitete lineare Hauptkomponentenanalyse (PCA) wird hier vorgestellt. Es wird kurz auf die Anwendungsmöglichkeiten eingegangen, und welche Rolle die Eigenschaften der PCA-Merkmale dabei spielen. Diese Eigenschaften sind bei einer Erweiterung der PCA zu einer nichtlinearen PCA (NLPCA) von Bedeutung.

Kapitel 3: Der klassische Autoencoder (s-NLPCA)
Es wird eine Variante der NLPCA auf der Basis neuronaler Netze vorgestellt, welche den Schwerpunkt dieser Arbeit bildet — die s-NLPCA mit dem *Autoencoder*.

Kapitel 4: Eindimensionale Merkmalsextraktion
Hier wird zunächst die Extraktion eines nichtlinearen Merkmals, des ersten nichtlinearen Hauptmerkmals, betrachtet. Es werden die Grenzen des Autoencoders aufgezeigt und Vorschläge zu ihrer Überwindung gegeben.

Kapitel 5: Mehrdimensionale Merkmalsräume
In diesem Kapitel werden mit dem klassischen Autoencoder (s-NLPCA) mehrdimensionale nichtlineare Merkmalsräume bestimmt. Dabei zeigen sich Schwächen, die insbesondere auf den symmetrischen Trainingsalgorithmus zurückzuführen sind. Mit dem klassischen Autoencoder lassen sich keine unkorrelierten Merkmale wie mit der linearen PCA bestimmen.

Kapitel 6: Hierarchische NLPCA
Ziel ist die Entwicklung einer nichtlinearen PCA, welche nichtlineare Merkmale ähnlich der linearen PCA in hierarchischer Ordnung extrahiert und hierdurch zu nichtlinearen unkorrelierten Merkmalen führt.
Es werden die Schwierigkeiten erläutert und darauf aufbauend ein hierarchischer Algorithmus (h-NLPCA) in Form einer hierarchischen Fehlerfunktion entwickelt. Diese Fehlerfunktion wird auf den Trainingsalgorithmus des Autoencoders angewendet.

Kapitel 7: Experimente
Zum Abschluss wird diese hierarchische NLPCA (h-NLPCA) mit der symmetrischen NLPCA (s-NLPCA) des klassischen Autoencoders, mit der linearen PCA und mit anderen modernen Methoden der Dimensionsreduktion und der Merkmalsextraktion verglichen.

Verzeichnis der am häufigsten benutzten mathematischen Notationen

N	Anzahl der Daten
n	Index über die Daten $n = 1, ..., N$
d	Anzahl der Dimensionen im originalen Datenraum
m	Anzahl extrahierter Merkmale, Anzahl der Dimensionen im Merkmalsraum
i	Merkmalsindex $i = 1, ..., m$
$\mathcal{X}$	originaler Datenraum, der gegebenen Rohdaten
$\mathcal{Z}$	Merkmalsraum
x	originaler Datenvektor $x \in \mathcal{X}, x = (x_1, ..., x_d)^T$
z	Datenvektor des Merkmalsraumes $z \in \mathcal{Z}, z = (z_1, ..., z_m)^T$
Φ_{extr}	Extraktionsfunktion $\Phi_{extr} : \mathcal{X} \to \mathcal{Z}$
Φ_{gen}	Generierungsfunktion $\Phi_{gen} : \mathcal{Z} \to \mathcal{X}$
$\hat{x}$	rekonstruierter Datenvektor $\hat{x} = \Phi_{gen}(\Phi_{extr}(x))$
E	Rekonstruktionsfehler, mittlerer quadratische Fehler (MSE) $E = \frac{1}{dN} \sum_n^N \sum_k^d \lvert x_k^n - \hat{x}_k^n \rvert^2$
E_i	Rekonstruktionsfehler bei Verwendung des i-ten Merkmals
$E_{i,j}$	Rekonstruktionsfehler bei Verwendung der Merkmale i und j

Kapitel 2

Lineare PCA und Sphering

2.1 Hierarchische Ordnung der Merkmale

Die PCA ist eine der bekanntesten Methoden der Dimensionsreduktion und Merkmalsextraktion [5]. Ziel ist es, möglichst viel Information bei der Reduzierung der Dimension zu erhalten. Bei der PCA wird angenommen, dass relevante Informationen in den Richtungen enthalten sind, in denen die Daten die größte Varianz besitzen. Diese Richtungen werden als Merkmale der Daten bezeichnet. Die PCA liefert diese Merkmale geordnet:

> **Varianz:** Das erste Merkmal bezeichnet die Richtung maximaler Varianz der Daten. Das zweite Merkmal bezeichnet die Richtung maximaler Varianz vom restlichen orthogonalen Unterraum bezüglich des ersten Merkmals.
> Allgemein: Die ersten m Merkmale spannen den m-dimensionalen linearen Unterraum größter Varianz der Daten auf.

Diese Ordnung der Merkmale wird im Folgenden als *hierarchische Ordnung* bezeichnet und ist bei der Betrachtung nichtlinearer Merkmale von Bedeutung. Die Vektoren, welche die Richtungen der Merkmale beschreiben, bilden zusammen eine orthogonale Basis, die den Merkmalsraum aufspannt. Durch Reduzierung der Basis um Merkmale, in deren Richtung die Varianz am geringsten ist, wird die Dimension des Merkmalsraumes reduziert, was zu der gewünschten niedrigdimensionalen Darstellung der Daten führt. Dies ist vorteilhaft, wenn die wahre intrinsische Dimensionalität der Daten geringer ist als die Dimension des originalen Datenraumes, die Daten daher auf einem linearen Unterraum liegen.
Die PCA ist eine orthogonale Basistransformation $\mathcal{X} \rightarrow \mathcal{Z}$ vom originalen Datenraum $\mathcal{X}$ in den Merkmalsraum $\mathcal{Z}$, welche invertierbar ist. Die inverse Transformation $\mathcal{Z} \rightarrow \mathcal{X}$ rekonstruiert aus den Merkmalswerten z die Daten $\hat{x}$. Die rekonstruierten Daten $\hat{x}$ sind die Projektionen der originalen Daten x auf den linearen Unterraum, den Merkmalsraum. Der mittlere quadratische Fehler (MSE) von $\hat{x}$, $E = \frac{1}{dN} \sum_n^N \sum_k^d |x_k^n - \hat{x}_k^n|^2$, wird als Rekonstruktionsfehler bezeichnet. Der Rekonstruktionsfehler E steht in enger Beziehung zur Varianz und zur hierarchischen Ordnung der Merkmale:

> **Rekonstruktionsfehler:** Der mittlere quadratische Fehler (MSE) der Projektionen $\hat{x}$ auf den Merkmalsraum, gegeben durch die ersten m

Merkmale, ist minimal bezüglich Projektionen auf beliebige andere m-dimensionale lineare Unterräume.

Für die Bestimmung der Merkmale existieren folglich zwei verschiedene gleichwertige Bedingungen, zum einen die *Maximierung der Varianz* und zum anderen die *Minimierung des Rekonstruktionsfehlers*. Die richtige Wahl der Bedingung spielt bei der Entwicklung eines Algorithmus zur Bestimmung nichtlinearer Merkmale eine entscheidende Rolle, siehe Kapitel 6.

2.2 Analytische Ausführung der PCA

Die Merkmale der linearen PCA können exakt durch Lösung eines Eigenwertproblems bestimmt werden. Dazu wird die Kovarianzmatrix

$$\mathcal{C} = \frac{1}{N} \sum_{n=1}^{N} x_n x_n^T$$

des auf Mittelwert gleich Null, $\sum_n x_n = 0$, korrigierten Datensatzes $\{x_n \in R^d \mid n = 1, ..., N\}$ benötigt. Die Richtungen der Merkmale entsprechen genau den Eigenvektoren ν zu den Eigenwerten λ der Kovarianzmatrix $\mathcal{C}$, also der Lösung der Gleichung $\mathcal{C}\nu = \lambda\nu$.
Die Varianz der ersten m Merkmale entspricht der Summe über die m größten Eigenwerte $\sum_{i=1}^{m} \lambda_i$, der Rekonstruktionsfehler entspricht der Summe über die restlichen Eigenwerte $\mathrm{MSE} = \sum_{\mathrm{i=m+1}}^{\mathrm{N}} \lambda_{\mathrm{i}}$, siehe [5].
Die PCA ist eine lineare Funktion $\Phi_{extr} : \mathcal{X} \rightarrow \mathcal{Z}$ vom originalen Datenraum $\mathcal{X}$ in den Merkmalsraum $\mathcal{Z}$. Datenvektoren $x \in \mathcal{X}$ werden Merkmalsvektoren $z \in \mathcal{Z}$ zugeordnet. Der einzelne Merkmalswert $z_i = \nu_i^T x$ des Merkmalsvektors $z = (z_1, ...z_m)^T$ ist eine Projektion des Datenvektors x auf den Eigenvektor ν_i, welcher die Richtung des Merkmals i kennzeichnet. Der Vektor ν_i ist normiert, so dass $\nu_i^T \nu_i = \mathrm{I}$ gilt.
Aus den Merkmalsvektoren $z = (z_1, ...z_m)^T$ können die originalen Datenvektoren x rekonstruiert werden $\hat{x} = \sum_{i=1}^{m} z_i \nu_i = \sum_{i=1}^{m} (\nu_i^T x)\nu_i$. Dies entspricht einer Projektion $\hat{x}$ der Datenvektoren x auf den linearen Unterraum, gegeben durch die Merkmalsrichtungen ν_i.
Die einzelnen Merkmalswerte z_i werden auch als Komponenten bezeichnet. Die Komponenten zu den ersten m relevanten Merkmalen sind die Hauptkomponenten. Die Bestimmung dieser Werte ist daher die Hauptkomponentenanalyse, PCA — *Principal Component Analysis*. Eine sehr ausführliche Darstellung der PCA ist zum Beispiel in [5] zu finden.

2.3 Symmetrie und Hierarchie

Zwei klassische Anwendungen der PCA bestehen darin, Daten zu komprimieren und Daten zu entrauschen. Hierbei wird angenommen, dass der Informationsgehalt der Varianz der Daten entspricht. Der Merkmalsraum wird um die Merkmalsrichtungen geringer Varianz reduziert, wodurch weniger relevante Informationen bzw. das Rauschen der Daten entfernt werden. Der dimensionsreduzierte Merkmalsraum stellt einen Unterraum des originalen Datenraumes dar. Beim Komprimieren und beim Entrauschen ist nur dieser Unterraum von Bedeutung. Es handelt sich daher um reine

Dimensionsreduktions-Anwendungen. An die Basis bzw. die Merkmale, welche diesen Unterraum beschreiben, werden keine weiteren Anforderungen gestellt. Die Merkmale können in beliebiger Anordnung den Unterraum beschreiben. Weder die Orthogonalität noch die hierarchische Ordnung werden benötigt. Die hierarchische Ordnung ist vorteilhaft, aber nicht zwingend notwendig, wenn die optimale Dimension des Merkmalsraumes nicht bekannt ist. Ein Algorithmus, welcher eine Basis bzw. Merkmale ohne spezielle Ordnung bestimmt, wird im Folgenden als symmetrischer Algorithmus bezeichnet. Ein symmetrischer Algorithmus behandelt alle Merkmale gleich, es gibt daher keine Bevorzugung bestimmter Merkmale.
Die PCA ist ein hierarchischer Algorithmus. Zusätzlich zur Bestimmung des optimalen Unterraumes werden die Merkmale, die den Unterraum aufspannen, hierarchisch geordnet bestimmt. Diese hierarchische Ordnung der Merkmale steht in engem Zusammenhang mit unkorrelierten Merkmalen [11], wie auch später zu sehen sein wird. Die PCA ist folglich auch ein Algorithmus der Merkmalsextraktion. Das Kriterium unkorrelierter Merkmale spielt bei der Verwendung der PCA als Vorverarbeitungsmethode eine wesentliche Rolle.
Die PCA kann auf verschiedene Weise als Vorverarbeitung benutzt werden. Zum einen können die beiden klassischen Anwendungen Komprimieren und Entrauschen als Vorverarbeitung verwendet werden, zum anderen gibt es eine weitere erfolgreichere Vorverarbeitungsmethode, welche aus der PCA abgeleitet werden kann und auf die Eigenschaft unkorrelierter Merkmale aufbaut — das *sphering* oder *whitening*.

2.4 Lineares Sphering (Whitening)

Das Sphering [6] ist eine Normierung der Daten. Beim linearen Sphering werden:

- lineare Korrelationen zwischen den einzelnen Variablen entfernt,
- die Daten auf einheitliche Varianz skaliert und
- der Mittelwert auf Null gesetzt.

Ziel ist es, eine sphärische Normalverteilung der Daten zu erreichen. Beim linearen Sphering wird dafür vorausgesetzt, dass die Daten eine Gaußverteilung besitzen und nur lineare Korrelationen zwischen den einzelnen Variablen existieren, siehe Abbildung 2.1.
Gesucht wird eine *sphering* Matrix W, welche die Daten x in eine Darstellung z abbildet, mit der Kovarianzmatrix $\mathcal{C}$ von z, $\mathrm{cov}\{z\} = \frac{1}{N}\sum_{n=1}^{N} z_n z_n^T$, als Einheitsmatrix I:

$$z = Wx \qquad ; \qquad \mathrm{cov}\{z\} = \mathrm{I}$$

Das Sphering ist bis auf eine Rotation eindeutig.
Die PCA kann zum Sphering benutzt werden, indem die Merkmalswerte z_i der Merkmale i auf Varianz gleich 1 skaliert werden. Die Merkmale i der PCA sind bereits linear unkorreliert, $\mathrm{cov}\{z\}$ ist eine diagonale Matrix.
Sphering ist auch direkt durchführbar, indem aus der Kovarianzmatrix $\mathcal{C} = \mathrm{cov}\{x\}$ eine *sphering* Matrix $W = \mathcal{C}^{-\frac{1}{2}}$ bestimmt wird. Hierbei erfolgt keine Rotation der Daten.
In der Signalverarbeitung wird Sphering auch als *whitening* bezeichnet.

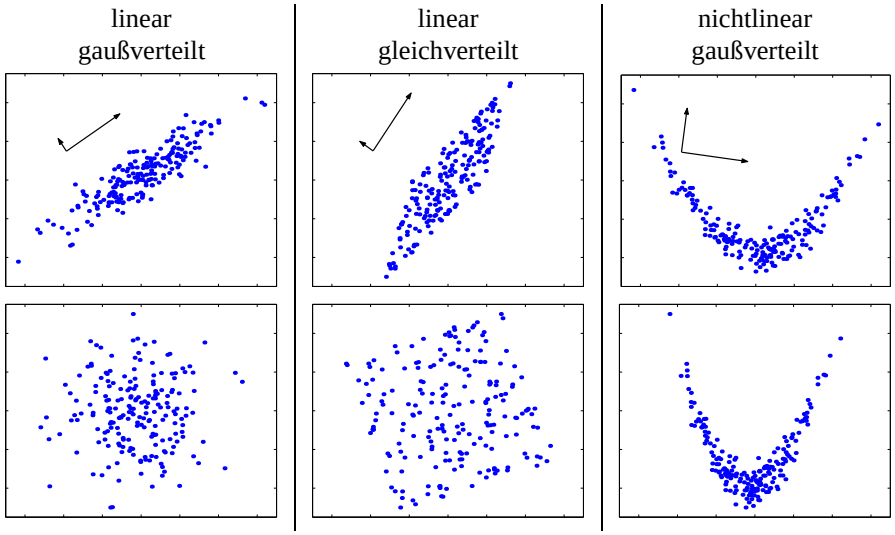


Abbildung 2.1: Sphering auf verschiedenen Datenverteilungen. Jeweils oben eine gegebene Datenverteilung und darunter das Ergebnis nach dem Sphering. Die PCA-Merkmalsrichtungen, welche auf einheitliche Varianz skaliert werden, sind durch Pfeile gekennzeichnet. Links: Sphering auf gaußverteilten Daten mit linearer Korrelation führt zu einer sphärischen Normalverteilung. Mitte: Gleichverteilte Daten, es fehlt noch eine Rotation für die Bestimmung unabhängiger Merkmale. Rechts: Bei Daten mit nichtlinearer Korrelation führt lineares Sphering nicht zu der gewünschten sphärischen Verteilung. In Kapitel 6 wird eine h-NLPCA vorgestellt, mit der nichtlineares Sphering möglich ist.

2.5 PCA auf Basis neuronaler Netze

Die lineare PCA ist eine vielfältig anwendbare Methode, aber beschränkt auf lineare Strukturen in Datenräumen. In der Praxis besitzen Datensätze aber häufig auch eine nichtlineare Struktur. Es ist daher naheliegend zu versuchen, die lineare PCA zu verallgemeinern für die Erkennung nichtlinearer Merkmale. Eine solche nichtlineare Erweiterung der PCA wird als NLPCA — *Nonlinear Principal Component Analysis* bezeichnet. Es gibt verschiedene Ansätze einer NLPCA. In dieser Arbeit wird die NLPCA auf Basis neuronaler Netze untersucht.

Für die lineare PCA existieren verschiedene Realisierungen mit neuronalen Netzen. Beispiele sind das *APEX-Netzwerk* [13] basierend auf *Ojas's Lernregel* [17], Sanger's *generalised hebbian algorithm* [22] oder der *lineare Autoencoder* [21]. Eine ausführliche Beschreibung dieser Methoden ist auch in [5] zu finden.

Der Autoencoder lässt sich relativ einfach nichtlinear erweitern [12] und ist daher eine Netzarchitektur, die sich sehr gut für eine NLPCA eignet. Die NLPCA auf der Basis eines Autoencoders steht im Mittelpunkt dieser Arbeit.

Kapitel 3

Der klassische Autoencoder (s-NLPCA)

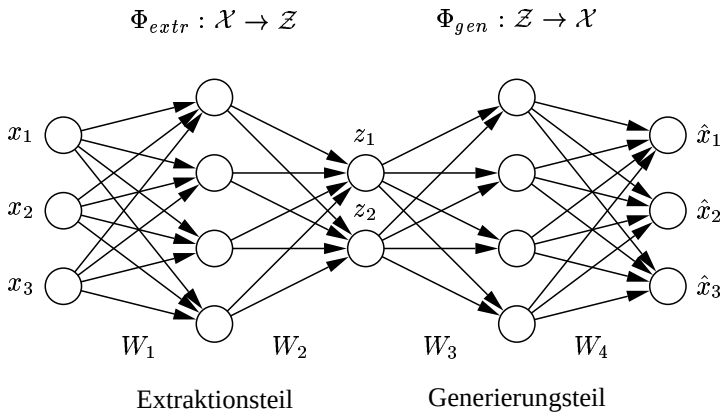


Abbildung 3.1: Autoencoder Netzwerk, [3-4-2-4-3]-Netz, mit nichtlinearen verdeckten Schichten zur Extraktion eines zweidimensionalen Merkmalsraumes aus einem dreidimensionalen Datenraum. Jede Schicht besitzt zusätzlich einen *bias*-Knoten, welcher aus Gründen der Übersicht nicht dargestellt ist.

3.1 Architektur

Der Autoencoder ist ein Multilagenperzeptron, welches die identische Abbildung lernt, die Ausgabe soll gleich der Eingabe sein. Das Netz besitzt aber eine mittlere Schicht mit weniger Knoten als in der Eingabe- oder Ausgabeschicht, siehe Abbildung 3.1. Hierdurch ist das Netz gezwungen, die Eingabe auf weniger Dimensionen abzubilden und von dieser Abbildung die selben Daten als Ausgabe wieder zu rekonstruieren. Die

mittlere Schicht repräsentiert den Merkmalsraum und wird im Folgenden als Merkmalsschicht bezeichnet. Die einzelnen Knoten der Merkmalsschicht repräsentieren die verschiedenen Merkmale i mit den Merkmalswerten z_i.
Der Autoencoder kann als aus zwei Teilen bestehend betrachtet werden. Der erste Teil, der Extraktionsteil, extrahiert die Merkmale, stellt daher eine Funktion $\Phi_{extr}(W_1, W_2) : \mathcal{X} \rightarrow \mathcal{Z}$ vom originalen Datenraum $\mathcal{X}$ in den Merkmalsraum $\mathcal{Z}$ dar. Der zweite Teil, der Generierungssteil, erzeugt aus den Merkmalswerten die originalen Daten. Er stellt folglich die inverse Funktion $\Phi_{gen}(W_3, W_4) : \mathcal{Z} \rightarrow \mathcal{X}$ vom Merkmalsraum in den originalen Datenraum dar.
Soll eine lineare PCA realisiert werden, sind nur zwei lineare Schichten notwendig, eine für die Extraktionsfunktion und eine weitere für die Generierungsfunktion. Für die Realisierung einer nichtlinearen PCA sind nichtlineare Funktionen notwendig, die beiden Teilnetze benötigen dafür zusätzlich jeweils mindestens eine nichtlineare verdeckte Schicht.
Eine der grundlegenden Arbeiten zum nichtlinearen Autoencoder ist die von Kramer [12]. Eine gute Einführung in neuronale Netze bietet Bishop [2].

3.2 Optimierung

Beim Trainieren des Autoencoders wird der Rekonstruktionsfehler $E = \frac{1}{dN} \sum_n^N \sum_k^d |x_k^n - \hat{x}_k^n|^2$ minimiert, wobei $\hat{x} = \Phi_{gen}(\Phi_{extr}(x))$ die Rekonstruktion (Netzausgabe) des Datenvektors $x = (x_1, ..., x_d)^T$ ist. Die Funktionen Φ_{extr} und Φ_{gen} lauten in Matrixschreibweise (der Einfachheit halber ohne *bias*) wie folgt:

$$\begin{aligned} z^n &= W_2 g_1(W_1 x^n) \\ \hat{x}^n &= W_4 g_2(W_3 z^n) \end{aligned}$$

W_1, W_2, W_3 und W_4 sind die Gewichtsmatrizen der einzelnen Netzschichten. $g_1(.)$ und $g_2(.)$ sind nichtlineare Transferfunktionen (z.B.: $g_1(.) = g_2(.) = tanh(.)$), welche elementweise auf die Matrizen/Vektoren angewendet werden. Die ausführliche Fehlerfunktion des gesamten Autoencoders lautet:

$$E(w) = \frac{1}{N} \sum_n^N ||x_k^n - W_4 g_2(W_3 W_2 g_1(W_1 x_k^n))||^2 \tag{3.1}$$

Der zu optimierende Gewichtsvektor $w = (w_1, ..., w_p)^T$ bezeichnet alle Gewichte der Matrizen $W_1, ..., W_4$. Die Optimierung erfolgt mit einem iterativen Gradientenverfahren. Dazu wird der Gradient der Fehlerfunktion $\nabla E(w) = (\frac{\partial E}{\partial w_1}, ..., \frac{\partial E}{\partial w_p})^T$ mit dem *backpropagation*-Algorithmus bestimmt. Die iterative Optimierung erfolgt im einfachsten Fall durch eine schrittweise Annäherung in Richtung des negativen Gradienten $w^{t+1} = w^t - \eta \nabla E(w)$ (t ist die Iteration, η ist die Schrittweite) oder durch ein effizienteres Gradientenverfahren wie dem konjugierten Gradientenabstieg, *conjugate gradient decent* [9, 18], welcher aufgrund besserer Resultate in dieser Arbeit verwendet wurde.
Verschiedene Regularisierungsvarianten zur Vermeidung eines übertrainierten Netzes (*overfitting*) werden in Kapitel 6.6 in Zusammmenhang mit einer hierarchische Extraktion von Merkmalen gesondert behandelt.

3.3 Anwendung

Es kann gezeigt werden [3, 1], dass der lineare Autoencoder eine Basis bzw. Merkmale findet, welche den Unterraum beschreiben, der durch die ersten m PCA-Merkmale gegeben ist. Die Richtungen dieser Merkmale sind aber nicht zwangsläufig identisch mit den Merkmalsrichtungen der PCA.
Für den nichtlinearen Fall gilt ebenfalls, dass der Autoencoder erfolgreich nichtlineare Unterräume (Merkmalsräume) extrahiert, deren kennzeichnende Merkmale aber in keiner besonderen Weise angeordnet sind.
Die Ursache liegt im symmetrischen Optimierungsalgorithmus. Alle Merkmale werden gleichwertig behandelt, es gibt keine Ordnung oder Bevorzugung bestimmter Merkmale. Die nichtlineare PCA auf der Basis des klassischen Autoencoders wird aufgrund dieser Symmetrie im Folgenden als s-NLPCA bezeichnet.
Die s-NLPCA ist beschränkt auf reine Dimensionsreduktions-Anwendungen, wie Entrauschen und Komprimierung, bei denen der Unterraum relevant ist, nicht aber die Merkmale selbst.
Der gesuchte Merkmalsraum muss bei der s-NLPCA eine geringere Dimension besitzen als der originale Datenraum. Die Daten werden sonst einfach kopiert. Bei der PCA ist dies nicht der Fall, eine Basistransformation findet auch bei gleicher Dimensionalität statt.
Der Autoencoder soll eine Extraktionsfunktion Φ_{extr} und eine Generierungsfunktion Φ_{gen} modellieren. Das bedeutet, diese Funktionen müssen auch existieren, was im nichtlinearen Fall nicht immer gegeben ist.
Die Beschränkungen des Autoencoders werden in den nächsten Kapiteln näher erläutert und es werden Lösungen vorgestellt, um diese Beschränkungen zu überwinden.

Kapitel 4

Eindimensionale Merkmalsextraktion

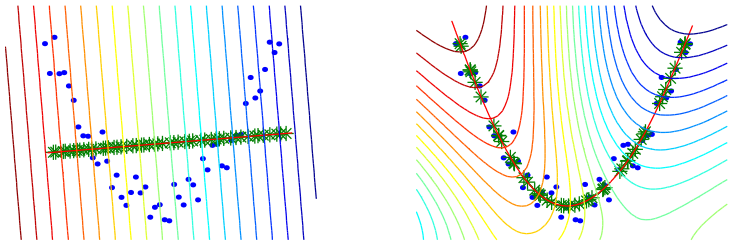


Abbildung 4.1: Extraktion eines Merkmals aus verrauschten Daten quadratischer Struktur. Der Autoencoder ([2-4-1-4-2]-Netz) als NLPCA beschreibt die Struktur der Daten '$\cdot$' besser als die lineare PCA. Der mittlere quadratische Fehler der Projektionen '$*$' auf das erste Merkmal (Linie) ist bei der NLPCA geringer als bei der PCA. Konturlinien zeigen die Richtung der Projektion. Daten auf einer Konturlinie werden dem gleichen Merkmalswert zugeordnet.

Mit nichtlinearen Merkmalen lassen sich Datenverteilungen häufig besser beschreiben als mit linearen Merkmalen, siehe Abbildung 4.1. Zunächst wird die Extraktion eines nichtlinearen Merkmals betrachtet — das Merkmal größter Varianz.
Der Autoencoder benötigt zur Extraktion eines Merkmals entsprechend einen Knoten in der Merkmalsschicht. Nichtlineare Korrelationen geringen Grades können hiermit sehr gut beschrieben werden. Bei komplexen nichtlinearen Korrelationen ist die Fähigkeit des Autoencoders jedoch begrenzt.
Das ist einerseits auf die lineare Initialisierung zurückzuführen, andererseits modelliert der Autoencoder zwei zueinander inverse Funktionen $\Phi_{extr} : \mathcal{X} \to \mathcal{Z}$ und $\Phi_{gen} : \mathcal{Z} \to \mathcal{X}$, wobei speziell die Abbildung $\mathcal{X} \to \mathcal{Z}$ nicht immer eindeutig gegeben ist. Anhand einer verrauschten Kreisstruktur werden die Beschränkungen genauer erläutert, und es werden Lösungen vorgeschlagen.

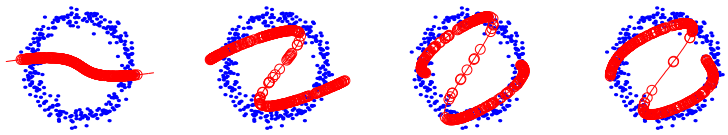


Abbildung 4.2: Gewöhnliche Optimierung. Alle Daten haben im Optimierungsalgorithmus den gleichen Einfluss. Jeweils vier der ersten 100 Lernschritte sind dargestellt, Schritte: 20, 60, 80 und 100.

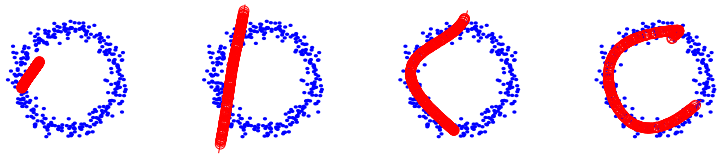


Abbildung 4.3: Algorithmus mit gewichteten Daten. Dargestellt sind die Schritte: 8, 20, 40 und 100.

4.1 Initialisierungsproblem

Die Gewichte des Autoencoders werden gewöhnlich mit kleinen Zufallswerten initialisiert. Der Autoencoder startet daher im linearen Bereich und bestimmt zuerst lineare Merkmale. Diese können ein lokales Minimum darstellen, von dem der Optimierungsalgorithmus nicht mehr zum optimalen nichtlinearen Merkmal findet.
Im Falle einer Kreisstruktur ist das Merkmal im Zentrum des Kreises gefangen. Eine Korrektur in Richtung einer Hälfte würde zur Erhöhung des Fehlers bezüglich der anderen Hälfte führen, siehe Abbildung 4.2. Was im Lokalen erwünscht ist, dass das Merkmal im Mittel der Daten liegt, stört global, wenn sich die Daten auf einer Struktur außerhalb des Mittelpunktes befinden.

4.1.1 Gewichtung der Daten

Eine mögliche Lösung besteht darin, den Autoencoder zuerst auf einer einfachen Teilstruktur der Daten zu trainieren und danach das erkannte Merkmal zur Initialisierung für das Trainieren auf der gesamten Struktur zu benutzen. Um zum Beispiel die Struktur eines Kreises zu lernen, kann das Netz zuerst mit einem Halbkreis und danach mit einem Vollkreis trainiert werden.
Da gewöhnlich die Struktur nicht bekannt ist, wird eine Methode zur Bestimmung einer Teilstruktur benötigt. Als wichtige Teilstruktur kann zum Beispiel eine hohe Datendichte betrachtet werden. Statt nur einen bestimmten Datenbereich auszuwählen, können alle Daten einbezogen werden, gewichtet nach dem Abstand zum Punkt größter Dichte. Hierdurch wird erreicht, dass andere Strukturen der Daten nicht total vernachlässigt werden.
Ein Datum auf dem Punkt größter Dichte kann mit 2 gewichtet werden, es wird da-

durch vom Lernalgorithmus doppelt gewertet. Das am weitesten entfernt liegende Datum kann mit Null gewichtet werden und hat daher keinen Einfluss auf das Ergebnis. Alle anderen Daten werden entsprechend ihrem Abstand mit Werten zwischen 0 und 2 gewichtet. Der Punkt größter Dichte kann mit einem Dichteschätzer bestimmt werden. Die Methode mit gewichteten Daten führte zu einem robusteren Lernverhalten bei komplizierten Merkmalen, siehe Abbildung 4.3. Auch bei einfacheren Merkmalen hat sich diese Methode in Experimenten nicht negativ ausgewirkt. Sie kann daher immer angewendet werden.

4.2 Inverse Abbildung

Der Autoencoder modelliert zwei zueinander inverse Funktionen $\Phi_{extr} : \mathcal{X} \to \mathcal{Z}$ und $\Phi_{gen} : \mathcal{Z} \to \mathcal{X}$, dabei ist er auf stetige Funktionen beschränkt. Speziell die Abbildung $\mathcal{X} \to \mathcal{Z}$ ist aber nicht immer eindeutig gegeben und daher mit dem Autoencoder nicht modellierbar. Es wird eine inverse Trainingsmethode vorgestellt, mit der die Generierungsfunktion Φ_{gen} allein modelliert werden kann.

4.2.1 Überschneidende Merkmale

Bei der Extraktion von geschlossenen oder zyklischen Merkmalen, beispielsweise einer Kreisstruktur, wird eine Funktion mit einer Unstetigkeitsstelle, einer Stufe, benötigt. Verschiedenen Punkten des originalen Datenraumes entlang eines geschlossenen Merkmales werden, z.B. kontinuierlich steigend, verschiedene Merkmalswerte zugeordnet. Wenn der Anfangspunkt wieder erreicht wird, ist der Merkmalswert verschieden vom Anfangsmerkmalswert, es ist ein Sprung bzw. eine Stufe nötig. Eine Stufenfunktion kann mit einem Autoencoder nicht dargestellt werden. Die Stufe kann nur mit einer stetigen Funktion angenähert werden, siehe Abbildung 4.4 oben.
Bei sich selbst überschneidenden Merkmalen ist die Abbildung $\mathcal{X} \to \mathcal{Z}$ nicht eindeutig. Dem Überschneidungspunkt im originalen Datenraum werden zwei verschiedene Punkte im Merkmalsraum zugeordnet. Solch eine mehrdeutige Abbildung ist mit dem Autoencoder nicht darstellbar.
Die inverse Abbildung $\mathcal{Z} \to \mathcal{X}$ dagegen ist in vielen Anwendungen eine stetige Funktion. Die Beschränkung des Autoencoders ist nur auf den Extraktionsteil zurückzuführen, der Generierungsteil ist durchaus in der Lage, geschlossene oder sich selbst überschneidende Merkmale zu rekonstruieren, vorausgesetzt die zugehörigen Merkmalswerte sind gegeben. Es ist daher naheliegend, den Autoencoder zu trennen und den Generierungsteil allein zu trainieren.

4.2.2 Inverses Training

Zum Trainieren des Generierungsteiles $\mathcal{Z} \to \mathcal{X}$ werden als Eingabewerte die eigentlich gesuchten Merkmalswerte $z^n = (z_1^n, ..., z_m^n)^T$ benötigt. Da nur die Ausgabewerte $x^n = (x_1^n, ..., x_d^n)^T$ vorgegeben sind, ist eine Methode nötig, die nicht nur die optimalen Gewichte liefert, sondern auch die optimalen Eingabewerte z^n. Solch eine Methode wird im Folgenden als inverses Training bezeichnet, da die inverse Funktion zur eigentlich gesuchten Extraktionsfunktion modelliert wird.
Aus neuronaler Sicht kann dazu vor den Generierungsteil, d. h. direkt vor die Merkmalsschicht eine weitere Schicht gesetzt werden, deren Anzahl Knoten der Anzahl der Datenvektoren entspricht. Diese neue Schicht dient als Eingabeschicht und

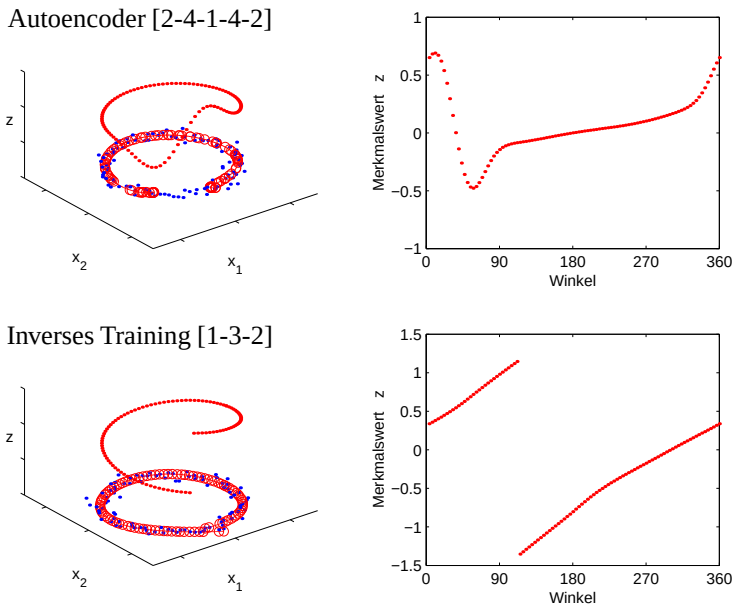


Abbildung 4.4: Vergleich eines invers trainierten [1-3-2] Generierungsteils (unten) mit dem normalen Autoencoder [2-4-1-4-2] Netz (oben). Links ist die Extraktionsfunktion $\Phi_{extr} : \mathcal{X} \rightarrow \mathcal{Z}$ dargestellt, die Merkmalswerte z^n sind über den originalen Daten x^n dargestellt. Das Hauptmerkmal beim Kreis ist der Winkel. Rechts werden die extrahierten Merkmalswerte dem originalen Winkel gegenübergestellt. Der normale Autoencoder approximiert grob die Stufe mit einer sinusähnlichen Funktion. Beim inversen Training ist die Stufe exakt darstellbar, der Merkmalswert korreliert annähernd linear mit dem Winkel.

bekommt als Eingabe die Einheitsmatrix I. Für das n-te zu rekonstruierende Datum bedeutet dies, dass als Eingabe ein Indikatorvektor benutzt wird, bei dem nur an der n-ten Stelle eine 1 und sonst 0 steht. Hierdurch wird erreicht, dass ein Gewicht der ersten Schicht direkt einem Datum zugeordnet wird. Bei einem mehrdimensionalen Merkmalsraum sind es entsprechend mehr Gewichte. Die Gewichte entsprechen den gesuchten Werten z des Merkmalsraumes und können direkt abgelesen werden. Bei der Implementierung dieser Methode kann bei großen Datensätzen ein Problem mit der Größe der Einheitsmatrix auftreten, für die daher eine *sparse* Matrix verwendet werden sollte.

Effizienter ist die direkte Optimierung ohne vorgesetzte Eingabeschicht. Die Optimierung der Gewichte und die Optimierung der Merkmalswerte als Eingaben können dazu als ein gemeinsames Optimierungsproblem betrachtet werden, wozu nur zusätzlich zu den Gradienten der Gewichte auch die Gradienten der Eingaben benötigt werden. Diese Gradienten erhält man durch Fortführung des Backpropagation-Algorithmus bis auf die Eingabeschicht, siehe auch [7].

Gesucht sind eine von Gewichten w abhängige Generierungsfunktion $\Phi_{gen}(w) : \mathcal{Z} \rightarrow \mathcal{X}$ und Merkmalsvektoren $z \in \mathcal{Z}$, so dass der Rekonstruktionsfehler minimal ist: $\min_{w,z} ||x - \Phi_{gen}(w, z)||^2$. Dazu kann folgende Fehlerfunktion

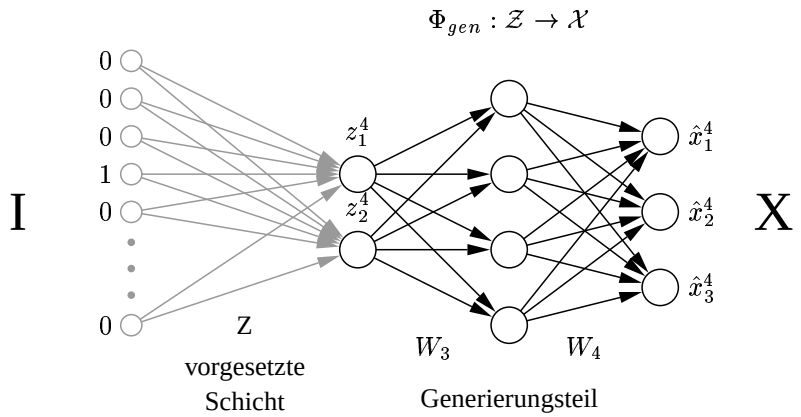


Abbildung 4.5: Inverse Trainingsmethode. Nur die Generierungsfunktion wird approximiert. Zusätzlich zu den Gewichten w werden auch die Eingaben z bestimmt. Trainiert wird entweder mit zusätzlicher Eingabeschicht (grau) oder effizienter nur mit dem Generierungsteil (schwarz) und Backpropagation bis zur Merkmalsschicht. Die Knoten sind mit dem 4-ten Datum ($n = 4$) gekennzeichnet.

minimiert werden:

$$E(w,z) = \frac{1}{dN} \sum_n^N \sum_k^d \left[x_k^n - \sum_j^h w_{kj} g \left(\sum_i^m w_{ji} z_i^n \right) \right]^2 \tag{4.1}$$

Aus der Fehlerfunktion ergeben sich die zur Optimierung benötigten partiellen Ableitungen, die Gradienten der Gewichte w_{kj}, w_{ji} und der Eingaben z_i^n :

$$\begin{array}{lcll} \frac{\partial E}{\partial w_{kj}} & = & \sum_n \sigma_k^n g(a_j^n) & ; \quad \sigma_k^n = \hat{x}_k^n - x_k^n \\ \frac{\partial E}{\partial w_{ji}} & = & \sum_n \sigma_j^n z_i^n & ; \quad \sigma_j^n = g'(a_j^n) \sum_k w_{kj} \sigma_k^n \\ \frac{\partial E}{\partial z_i^n} & = & \sigma_i^n & ; \quad \sigma_i^n = \sum_j w_{ij} \sigma_j^n \end{array}$$

σ_k, σ_j und σ_i sind die durch Backpropagation erhaltenen partiellen Fehler der einzelnen Knoten jeweils der Ausgabeschicht, der verdeckten Schicht und der Eingabeschicht. $\hat{x} = \Phi_{gen}(w,z)$ ist die Netzausgabe, die Rekonstruktion des originalen Datums x. $g(a_j^n)$ ist die Ausgabe des j-ten Knotens in der verdeckten Schicht beim n-ten Datum, mit $a_j^n = \sum_i w_{ji} z_i^n$ und $g(.)$ als nichtlineare Transferfunktion, z.B. $g(.) = tanh(.)$. Die *bias* Gewichte wurden nicht extra betrachtet. Sie können in den Summen als zusätzliche Elemente w_{k0} und w_{j0} mit zugehörigen konstanten Eingaben $z_0 = 1$ und $g(a_0)$ = 1 einbezogen werden.
Im Gegensatz zu [7], wo die Gewichte und die Eingabe abwechselnd und mit jeweils unterschiedlicher Lernschrittweite optimiert wurden, werden hier die Gewichte und Eingaben mit dem *konjugierten Gradientenabstieg* [9, 18] gemeinsam optimiert, da eine getrennte Optimierung nicht nötig ist und ein Oszillieren des Algorithmus hiermit verhindert wird.
Abbildung 4.6 zeigt die Extraktion eines Merkmals mit Überschneidungspunkt. Der Unterschied zum normalen Autoencoder ist in Abbildung 4.4 dargestellt.

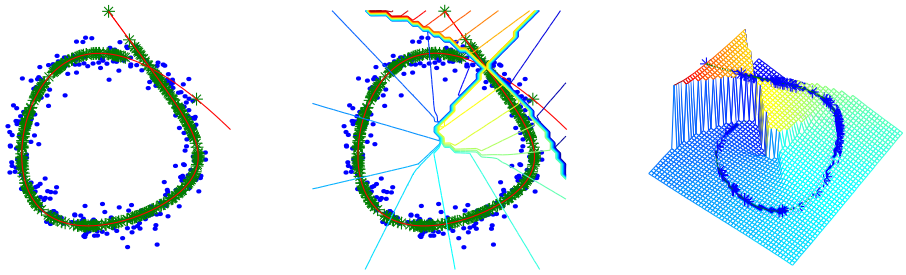


Abbildung 4.6: Extraktion eines Merkmals mit Überschneidungspunkt. Der Generierungsteil [1-3-2] des Autoencoders wurde dazu invers trainiert. Die gegebenen Daten x ('$\cdot$') besitzen eine verrauschte Kreisstruktur.
Links ist der extrahierte Merkmalsraum (Linie) mit den darauf projizierten Daten $\hat{x}$ ('$*$') dargestellt. In der mittleren Abbildung sind zusätzlich die Höhenlinien der geschätzten Extraktionsfunktion $\Phi_{extr} : \mathcal{X} \rightarrow \mathcal{Z}$ eingezeichnet.
Die Schätzung der Extraktionsfunktion selbst ist rechts als Gitternetz dargestellt. Zu einem beliebigen Wert des originalen Datenraumes $\mathcal{X}$ (x-y-Ebene) wird der optimale Wert z des Merkmalsraumes $\mathcal{Z}$ (z-Achse) bestimmt. Zusätzlich ist die Generierungsfunktion $\Phi_{gen} : \mathcal{Z} \rightarrow \mathcal{X}$ eingezeichnet (Linie mit '$*$').

Schätzung der Extraktionsfunktion

Da der Extraktionsteil des Autoencoders fehlt, existiert keine Funktion $\Phi_{extr} : \mathcal{X} \rightarrow \mathcal{Z}$, die neue Daten in den Merkmalsraum $\mathcal{Z}$ abbildet. In [7] wird vorgeschlagen, einen Extraktionsteil mit Hilfe der Merkmalswerte der Trainingsdaten nachträglich zu trainieren, dies führt aber bei Merkmalen wie den hier betrachteten zu keinem Erfolg, da der Extraktionsteil die geforderte Abbildung nicht darstellen kann. Bei einfachen nichtlinearen Merkmalen wiederum ist eine Trennung des Netzes nicht nötig.
Neue Daten können aber ähnlich wie Trainingsdaten behandelt werden. Es werden zu gegebenen neuen Ausgabedaten x optimale Eingabedaten z gesucht. Die Generierungsfunktion, gegeben durch die Gewichte des Generierungsteiles, bleibt dabei konstant.
Die Fehlerfunktion 4.1 ist jetzt nur von den Eingaben z abhängig, $E(w, z)$ wird zu $E(z)$. Zu den Eingaben können auch wieder die Gradienten berechnet werden $\frac{\partial E}{\partial z_i^n} = \sigma_i^n$. Bei der zuerst vorgestellten Variante mit einer vorgesetzten Eingabeschicht wird entsprechend nur diese Schicht optimiert. Die Bestimmung der Eingaben zu neuen Daten ist somit ein wesentlich geringeres Optimierungsproblem, welches wieder mit einem Gradientenverfahren gelöst werden kann.
Ein optimaler Merkmalswert z ist nicht immer garantiert, daher sollte mehrmals mit unterschiedlicher Initialisierung optimiert werden. In Abbildung 4.6 rechts wurde die Extraktionsfunktion auf diese Weise erzeugt. Nach 20 Optimierungsläufen waren alle 900 eindimensionalen Merkmalswerte des Gitternetzes optimal bezüglich des Rekonstruktionsfehlers.
Es wäre zu vermuten, dass an Überschneidungspunkten die Daten nicht eindeutig zugeordnet werden, sondern zufällig zwischen den zwei Möglichkeiten des Merkmalswertes wechseln. Dies war aber in diesem Experiment nicht der Fall. Es hat sich ein Merkmalswert gegenüber dem anderen durchgesetzt.

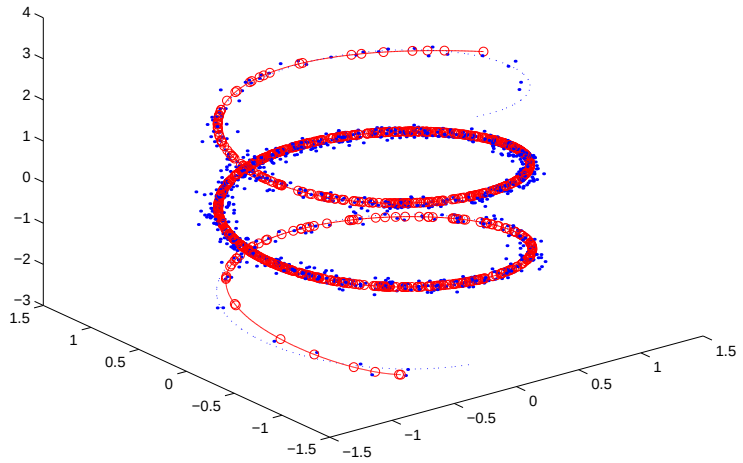


Abbildung 4.7: Approximation der Spiral-Struktur mit dem Generierungsteil des Autoencoders. Optimiert wurde ein [1-8-3] Netz mit der inversen Trainingsmethode. Die Projektion $\hat{x}$ ('$\circ$') auf den eindimensionalen Merkmalsraum (Linie) stellt eine rauschfreie Rekonstruktion der Spiral-Daten x ('$\cdot$') dar.

Auch haben Experimente gezeigt, dass es hier kein Initialisierungsproblem wie im Kapitel 4.1 gibt. Die Kreisstruktur ist ohne gewichtete Initialisierung bestimmbar.
Die Komplexität des Problems ist beim einzeln trainierten Generierungsteil geringer. Es sind beim kreisförmigen Merkmal nur drei Knoten in der verdeckten Schicht nötig ([1-3-2]-Netz) gegenüber jeweils vier Knoten beim klassischen Autoencoder ([2-4-1-4-2]-Netz).

4.2.3 Spiral-Struktur

Mit dem inversen Training kann eine komplexe Spiral-Struktur wie aus [14] approximiert werden, siehe Abbildung 4.7. Valpola und Honkela benutzen in [14] eine *nonlinear independent factor analysis*. Die Ergebnisse sind weitgehend identisch, inklusive der Ungenauigkeit an den Enden der Spirale. Dies ist auch nicht verwunderlich, da beiden Methoden die gleiche Netzarchitektur [1-hid-3] zugrunde liegt. Das Training unterscheidet sich darin, dass in [14] wieder die Eingaben und die Gewichte jeweils getrennt bestimmt werden. Die Eingaben werden dabei durch *mixtures of Gaussians* modelliert.
Der Datensatz besteht aus 1000 Daten gaußverteilt entlang eines Merkmals z. Die Abbildung $\Phi_{gen} : \mathcal{Z} \rightarrow \mathcal{X}$ in dem dreidimensionalen Datenraum $\mathcal{X}$ erfolgt mittels Sinus und Cosinus:

$$x_1 = \sin(\pi z) \quad ; \quad x_2 = \cos(\pi z) \quad ; \quad x_3 = z$$

Zusätzlich wurden die Daten x mit additivem gaußschen Rauschen η (std $\sigma = 0.5$) verfremdet.

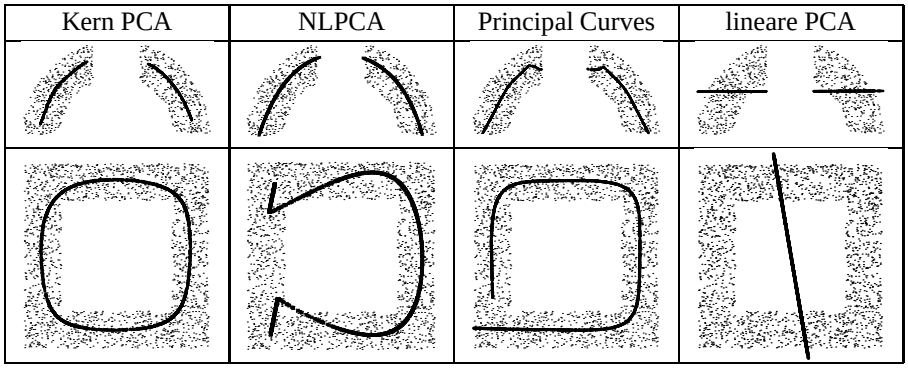


Abbildung 4.8: Vergleich verschiedener Methoden zum Entrauschen von Datensätzen, aus [16]. Es wird eine eindimensionale Struktur als Rekonstruktion der verrauschten Daten gesucht.

Mit dem klassischen Autoencoder konnte diese Spiral-Struktur nicht gefunden werden. Die Ursache liegt wieder in einem starken lokalen Minimum, gegeben durch ein zuerst bestimmtes lineares Merkmal, analog zur Kreisstruktur in Kapitel 4.1 .

4.2.4 Missing Data

Die inverse Trainingsmethode ist geeignet zur Extraktion von Merkmalen aus stark unvollständigen Datensätzen. Bei fehlender Variable x_k^n eines Datenvektors $x^n = (x_1^n, ..., x_d^n)^T$ wird einfach der partielle Fehler $\sigma_k^n = \hat{x}_k^n - x_k^n$ auf Null gesetzt. Hierdurch können alle vorhandenen Variablen optimal genutzt werden. Es werden keine vollständigen Datenvektoren x^n benötigt. Lineare und nichtlineare Korrelationen zwischen den Variablen x_k werden beachtet. Aus den extrahierten Merkmalswerten z^n sind vollständige rauschfreie Daten $\hat{x}^n$ bestimmbar.

4.3 Vergleich verschiedener Methoden

Es existieren verschiedene Methoden zur Extraktion nichtlinearer Merkmale und zur nichtlinearen Dimensionsreduktion. Eine sehr erfolgreiche Methode ist die *Kern PCA* [24, 23]. Weitere Methoden sind der *principal curves* Algorithmus [8] und die in dieser Arbeit untersuchte Methode der NLPCA mit dem klassischen nichtlinearen *Autoencoder* [12].
In [16] wird ein Vergleich dieser Ansätze beim Entrauschen von Daten gezeigt. Es werden eindimensionale Strukturen im zweidimensionalen Datenraum gesucht, siehe Abbildung 4.8. Die Kern PCA beschreibt in diesem Experiment die eindimensionale Struktur am besten. Dabei werden aber 4 Merkmale der Kern PCA benötigt. Es findet daher keine Dimensionsreduktion statt. Bei den anderen Methoden wird jeweils nur ein Merkmal benutzt. Jedes Datum $x^n = (x_1^n, ..., x_d^n)^T$ wird dabei einem skalaren Merkmalswert z^n zugeordnet.

Kapitel 5

Mehrdimensionale Merkmalsräume

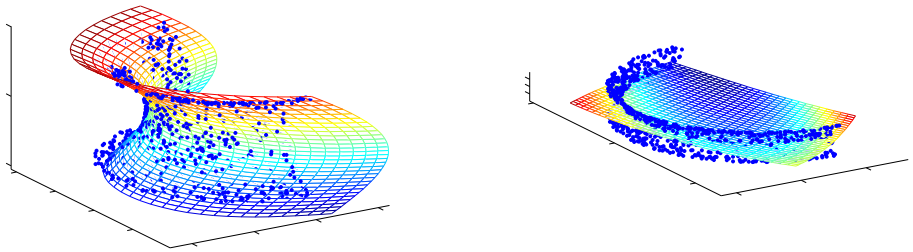


Abbildung 5.1: Daten liegen im zweidimensionalen nichtlinearen Unterraum (Torus-Ausschnitt). Dargestellt ist der vom Autoencoder extrahierte zweidimensionale Merkmalsraum als Gitternetz. Sind die Varianzen in Richtung der beiden Merkmale sehr unterschiedlich, wird vom Autoencoder nicht mehr die optimale Lösung bestimmt (rechts). Die Lösung ist annähernd linear.

Bisher wurde nur die Extraktion eines Merkmals betrachtet, wesentlich interessanter ist aber die Extraktion mehrerer nichtlinearer Merkmale. Da beim Autoencoder die Merkmalsschicht den Merkmalsraum repräsentiert, muss die Anzahl der Knoten der Anzahl gewünschter Merkmale entsprechen.
Zwei wesentliche Beschränkungen besitzt der klassische Autoencoder. Zum einen können Merkmale mit deutlich unterschiedlicher Varianz nicht erkannt werden, und zum anderen sind nichtlineare Merkmale einer niedrigdimensionalen Lösung nicht in einer höherdimensionalen Lösung enthalten. In beiden Fällen liegt der Grund im Wesentlichen in der Symmetrie des Lernalgorithmus, das heißt, in der gleichwertigen Behandlung der Merkmale.

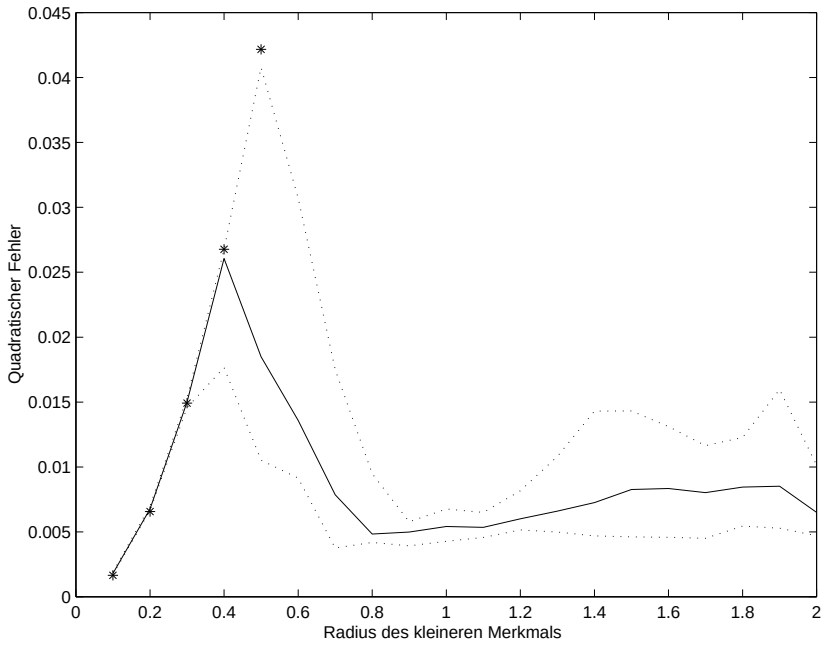


Abbildung 5.2: Fehler bei verschiedenen Größenverhältnissen zweier Merkmale (Halbkreise), nur der Radius des kleineren Merkmals wird verändert.
Dargestellt ist der Median über 100 Durchläufe und der Median der oberen und der unteren Abweichung (Punktlinien).
Außerdem ist zum Vergleich der Fehler bei linearer PCA mit $*$ dargestellt. Dies entspricht im unteren Bereich einem linearen Merkmalsraum in der x-y-Ebene.

5.1 Merkmale unterschiedlicher Varianz

Wenn die Varianzen in Richtung der Merkmalsdimensionen wesentlich unterschiedlich sind, hat ein Autoencoder Probleme, auch nur ein einziges nichtlineares Merkmal zu finden. Das Netz benutzt dabei die Möglichkeit mehrerer Merkmale, um ein nichtlineares Merkmal mit mehreren annähernd linearen Merkmalen zu beschreiben, siehe Abbildung 5.1 rechts. Dies entspricht einer linearen PCA Lösung, mit dem Unterschied, dass die linearen Merkmale nicht hierarchisch geordnet sind, aber den gleichen Unterraum beschreiben.
In Abbildung 5.2 wird gezeigt, ab wann die Größenverhältnisse kritisch sind. Es wurde ein Datenraum wie in Abbildung 5.1 benutzt, bestehend aus einem großen halbkreisförmigen Merkmal mit Radius 2 und einem darauf liegenden kleineren halbkreisförmigen Merkmal mit jeweils unterschiedlichem Radius. In Abbildung 5.1 beträgt er links 0,7 und rechts 0,2. Hierdurch ändert sich auch der Radius des größeren Merkmals, der nur außen bei 2 bleibt, innen aber um den Radius des kleineren Merkmals verringert wird, also links 1,3 und rechts 1,8 ist. Die Daten sind unverrauscht, können also fehlerfrei gelernt werden. Um auszuschließen, dass das Netz aufgrund eines Regularisierungsterms in den linearen Bereich gedrängt wird, wurde kein *weight-decay* benutzt, daher also keine großen Gewichte und keine Nichtlinearität

bestraft.
In Abbildung 5.2 ist zu sehen, dass das Netz bis zu einem Radius von $0,3$ des kleineren Merkmals die nichtlineare Struktur nicht erkennt. Es schneidet einfach die Dimension in Richtung der kleinsten Varianz ab, wie bei linearer PCA. Im Bereich von $0,4$ bis $0,6$ des Radius des kleineren Merkmals wird die nichtlineare Struktur noch nicht oder sehr schlecht erkannt. Erst ab einem Radius von $0,7$ wird die nichtlineare Struktur gut erkannt. Abbildung 5.1 rechts zeigt somit die kleinste Varianz des kleineren Merkmals, bei der die gesamte nichtlineare Struktur noch gut erkannt wird. Ab einem Radius des kleineren Merkmals von $1,2$ wird der quadratische Fehler wieder etwas größer. Dies darf nicht als schlechteres Ergebnis bewertet werden, da die Daten nicht normiert sind. Bei einem größeren Radius ist der Fehler bei falsch erkanntem Merkmalsraum viel größer. Die nichtlineare Struktur wurde somit nur bis zu einem Größenverhältnis der zwei Merkmale von $1:3$ sehr gut und ab einem Größenverhältnis von $1:7$ sogar gar nicht mehr erkannt. Der klassische Autoencoder tendiert dazu, eine Lösung so linear wie möglich zu liefern. In den folgenden Kapiteln wird sich zeigen, dass eine hierarchische Ordnung in solchen Spezialfällen auch helfen kann, einen Unterraum mit geringerem quadratischen Fehler zu bestimmen, siehe Abbildung 6.4 .

5.2 Stabilität der Merkmale

Wird ein Autoencoder mit nur einem Knoten in der Merkmalsschicht benutzt, kann damit ein nichtlineares Merkmal bestimmt werden, welches die Daten sehr gut beschreibt, vorausgesetzt, es handelt sich um ein Merkmal mit moderater Nichtlinearität, siehe Kapitel 4. Dieses Merkmal ist das Merkmal größter Varianz und somit das erste Merkmal der NLPCA.
Werden weitere Merkmale mit Hilfe weiterer Knoten in der Merkmalsschicht extrahiert, geht das nichtlineare Merkmal größter Varianz meist verloren. Keines der Merkmale einer mehrdimensionalen Lösung ist identisch mit dem nichtlinearen Merkmal einer eindimensionalen Lösung. Die nichtlinearen Merkmale eines niedrigdimensionalen Merkmalsraumes kommen in einem höherdimensionalen Merkmalsraum im Allgemeinen nicht mehr vor, da sich dieser höherdimensionale Merkmalsraum mit Merkmalen geringeren nichtlinearen Grades beschreiben lässt. Mit weiteren Merkmalen nähert sich die Lösung immer mehr einer linearen Lösung an, die aber entgegen der PCA keine hierarchisch geordneten Merkmale besitzt. Es macht daher keinen Sinn, mit dem klassischen symmetrischen Autoencoder einen Merkmalsraum zu extrahieren, mit einer Dimension, die der linearen intrinsischen Dimensionalität der Daten entspricht, da als Lösung nur eine lineare Lösung erwartet werden kann, die effizienter mit der linearen PCA erzeugt wird. Erst recht macht es keinen Sinn, einen Merkmalsraum gleicher Dimensionalität wie die des originalen Datenraumes zu extrahieren. Das Ergebnis wäre der originale Datenraum, nur willkürlich gestaucht und gedreht, abhängig von der zufällig gewählten Initialisierung der Gewichte.

Der Grund für den Verlust der nichtlinearen Merkmale in höherdimensionalen Merkmalsräumen liegt in der Symmetrie des Autoencoders. Es wird beim symmetrischen Autoencoder kein Merkmal bevorzugt bzw. stärker bewertet. Der Autoencoder beschreibt den gesuchten Merkmalsraum mit mehreren gleichwertigen, eher linearen Merkmalen als mit einem starken nichtlinearen Merkmal und weiteren weniger bedeutenden Merkmalen. Was fehlt, ist eine hierarchische Ordnung der Merkmale wie

bei der linearen PCA.
Die Bedingung, den quadratischen Fehler zu minimieren, enthält kein hierarchisches Kriterium. Sie erzwingt nur einen nichtlinearen Unterraum, nicht aber eine Ordnung der Merkmale, die diesen Unterraum beschreiben.
Es sei anzumerken, dass die Symmetrie in vielen Anwendungsfällen ausreicht, da oft nur der nichtlineare Unterraum gesucht ist und die Anordnung der Merkmale keine Rolle spielt.

Kapitel 6

Hierarchische NLPCA

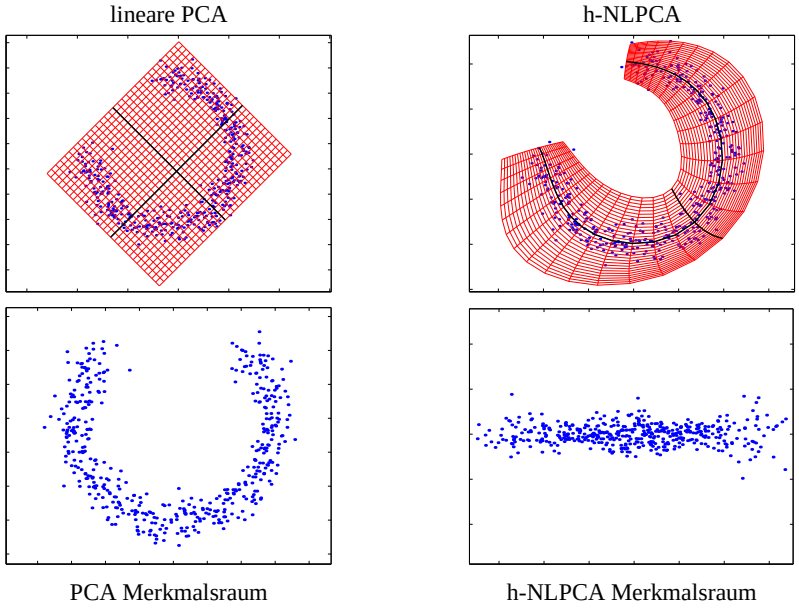


Abbildung 6.1: Gegeben ist ein zweidimensionaler Datensatz, generiert aus einem 3/4 Kreis und additivem gaußschen Rauschen. Dargestellt sind zwei Merkmale, jeweils die der linearen PCA und die der in diesem Kapitel vorgestellten hierarchischen, nichtlinearen PCA (h-NLPCA). Das erste Merkmal bezeichnet die Richtung/Kurve maximaler Varianz. Die Merkmale sind als Gitternetz dargestellt, welches die Koordinaten des Merkmalsraumes (unten) repräsentiert. Alle Daten entlang einer Linie werden dem gleichen Merkmalswert zugeordnet. Der Wert Null ist gekennzeichnet durch eine dicke Linie. Die h-NLPCA entfernt nichtlineare Korrelationen. Die lineare PCA ist dazu nicht in der Lage.

Ziel ist die Erweiterung der nichtlinearen PCA auf Basis des Autoencoders (s-NLPCA) um ein hierarchisches Kriterium zu einer hierarchischen, nichtlinearen PCA (h-NLPCA), welche die im Folgenden definierten NLPCA-Kriterien weitgehend erfüllt. Es werden die Schwierigkeiten erläutert und darauf aufbauend ein hierarchischer Algorithmus in Form einer hierarchischen Fehlerfunktion entwickelt.

6.1 Kriterien einer nichtlinearen PCA

Als NLPCA wird eine nichtlineare Erweiterung der linearen PCA bezeichnet. Die Merkmale sollten daher im Wesentlichen die gleichen Eigenschaften aufweisen wie die der PCA, mit der einzigen Ausnahme, dass sie nichtlinear, d.h. gekrümmt sein können. Die Eigenschaften linearer Merkmale der PCA werden dahingehend überprüft, inwieweit sie von nichtlinearen Merkmalen erfüllt werden können.

- **Hierarchische Ordnung der Merkmale**
  bezüglich Varianz und Rekonstruktionsfehler

  - *Varianz*: Das erste Merkmal bezeichnet die Richtung größter Varianz der Daten. Die ersten m Merkmale beschreiben den m-dimensionalen linearen Unterraum größter Varianz der Daten.
  - *Rekonstruktionsfehler*: Eine Projektion $\hat{x}$ der Daten x auf die erste Richtung besitzt den kleinsten mittleren quadratischen Fehler $E = \frac{1}{dN}\sum_n^N \sum_k^d |x_k^n - \hat{x}_k^n|^2$ bezüglich jeder anderen Richtung. Eine Projektion der Daten auf den Unterraum, beschrieben durch die ersten m Merkmale, ist minimal bezüglich jedem anderen linearen Unterraum.

  Nichtlineare Merkmale können ebenfalls eine hierarchische Ordnung besitzen. Das erste Merkmal ist die Kurve größter Varianz der Daten. Die ersten m Merkmale beschreiben den m-dimensionalen nichtlinearen, gekrümmten Unterraum größter Varianz der Daten. Eine Projektion auf diesen m-dimensionalen nichtlinearen Unterraum besitzt den kleinsten mittleren quadratischen Fehler. Der Grad der Nichtlinearität bzw. die Komplexität des verwendeten Modells sollte dabei eine sinnvolle Regularisierung besitzen.

- **Stabilität der Merkmale**
  Bei der PCA sind die Merkmale unabhängig von der Anzahl aller bestimmten Merkmale. Die Richtung des ersten Merkmals einer Bestimmung von n Merkmalen entspricht der Richtung des ersten Merkmals von m Merkmalen, $m \neq n$. Die Stabilität der Merkmale kann auch für nichtlineare Merkmale gelten und ist erreichbar durch eine hierarchische Ordnung.

- **Orthogonalität**
  PCA ist eine orthogonale Basistransformation, die Merkmale sind orthogonale Richtungen, beschreibbar durch orthogonale Vektoren. Nichtlineare Merkmale bezeichnen Kurven und sind daher nicht durch Vektoren beschreibbar. Die Eigenschaft der Orthogonalität kann nur lokal gelten.

- **Mittelwert gleich Null**
  Die Projektionen der Daten auf die Merkmale der PCA haben einen Mittelwert gleich Null. Projektionen auf nichtlineare Merkmale können ebenfalls einen Mittelwert gleich Null besitzen.

Keine der hier betrachteten Eigenschaften werden von der s-NLPCA erfüllt. Ein Sortieren der s-NLPCA Merkmale nach der Varianz führt nicht zu einer hierarchischen Ordnung.

6.2 Hierarchische Lösungsansätze

Es werden zwei Lösungsansätze zur Bestimmung einer hierarchischen Ordnung der Merkmale im Autoencoder vorgestellt, welche jedoch für den nichtlinearen Fall nicht anwendbar sind. Sie sollen aber die Schwierigkeiten bei der Einführung hierarchischer Bedingungen verdeutlichen und so zum besseren Verständnis für den im nächsten Kapitel vorgeschlagenen hierarchischen Optimierungsalgorithmus beitragen. Es handelt sich zum einen um eine hierarchische Bedingung an die Varianz und zum anderen um eine sequenzielle (deflationäre) Minimierung des Rekonstruktionsfehlers.

6.2.1 Varianz Maximierung

Eine hierarchische Bedingung an die Varianz lautet folgendermaßen: das erste Merkmal soll die größte Varianz besitzen, das zweite die zweit-größte und so weiter, mit der Nebenbedingung orthogonaler Merkmale.
Eine Schwierigkeit bei der Maximierungsbedingung ist, dass keine Begrenzung existiert. Die Varianz kann beliebig groß werden. Dies ist aber durch eine Begrenzung der maximalen Varianz auf eins lösbar.
Das eigentliche Problem liegt aber darin, dass der originale Datenraum linear so gestreckt oder gestaucht werden kann, dass er ohne Drehung und ohne nichtlineare Transformation einen Merkmalsraum darstellen kann, der eine beliebige Bedingung unterschiedlicher Varianzen erfüllt.
Eine nichtlineare hierarchische Merkmalsextraktion mit Hilfe von Bedingungen an die Varianz der Merkmale erscheint daher sehr schwierig. Es sollte daher versucht werden, die hierarchische Bedingung nicht direkt an den Merkmalsraum und daher nicht an die Merkmalsschicht, die diesen repräsentiert, zu stellen.

6.2.2 Deflationäre Fehlerminimierung

Gleichbedeutend zur Forderung maximaler Varianz ist im linearen Fall die Forderung eines minimalen Rekonstruktionsfehlers. Bei der linearen PCA ist die größtmögliche Varianz der ersten m orthogonalen Merkmale gegeben durch die Summe der Eigenwerte der Kovarianzmatrix $\sum_{i=1}^{m} \lambda_i$. Der Rekonstruktionsfehler entspricht genau der Summe der restlichen Eigenwerte $\sum_{i=m+1}^{N} \lambda_i$. Die Eigenwerte entsprechen den Varianzen der einzelnen Merkmale. Folglich ist eine Minimierung des Rekonstruktionsfehlers gleichbedeutend mit der Minimierung der Varianzen der restlichen $N - (m + 1)$ Merkmale und damit der Maximierung der ersten m Merkmale.
Für den nichtlinearen Fall sind die beiden Forderungen nicht identisch. Der Unterschied resultiert aus der unterschiedlichen Berechnung der Varianzen und der Rekonstruktionsfehler. Die Varianz wird im Merkmalsraum berechnet, was im originalen Datenraum einem Abstand der Daten zum Mittelwert entlang eines nichtlinearen Merkmals, einer Kurve, entspricht. Der Rekonstruktionsfehler dagegen ist der exakte lineare Abstand zwischen dem originalen Datum und dem rekonstruierten Datum, also die Sehne über dieser Kurve. Der mittlere quadratische Rekonstruktionsfehler über m Merkmale kann daher geringer ausfallen als die Varianz der restlichen $N - (m + 1)$ Merkmale. Dieser Unterschied wird im Folgenden vernachlässigt.
Eine Bedingung an den Rekonstruktionsfehler hat offensichtliche Vorteile gegenüber einer Bedingung an die Varianzen. Zum einen handelt es sich dabei um ein Minimierungsproblem, welches durch Null nach unten beschränkt ist. Zum anderen wird die

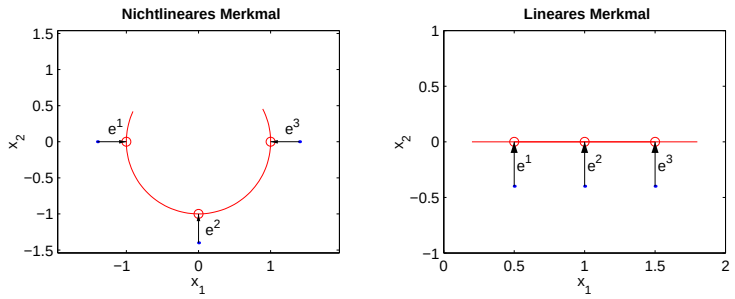


Abbildung 6.2: Die Fehler e^1, e^2 und e^3 ($e^n = x^n - \hat{x}^n$) besitzen bezüglich des Merkmals (Linie) die gleiche Richtung. Im nichtlinearen Fall (links) können diese Fehler bezüglich des originalen Datenraumes $\mathcal{X}$, wo sie bestimmt werden, in verschiedene Richtungen zeigen. Im linearen Fall (rechts) sind die Richtungen der Fehlervektoren bezüglich des Merkmals und bezüglich des originalen Datenraumes identisch.

Bedingung an den originalen Datenraum gestellt, wodurch die zu minimierenden Abstände fest gegeben sind und durch keine Skalierung verfälscht werden können.
Die Schwierigkeit besteht aber noch darin, eine hierarchische Ordnung der Merkmale über den Rekonstruktionsfehler zu erzwingen. Ein Ansatz hierfür ist die Methode des sequenziellen (deflationären) oder seriellen Lernens einzelner Merkmale durch mehrere hintereinander geschaltete Autoencoder, siehe auch [12]. Die Autoencoder besitzen jeweils nur einen Knoten in der Merkmalsschicht. Der erste Autoencoder bestimmt auf den Daten x wie bisher das größte Merkmal. Der zweite Autoencoder bestimmt auf den Fehlern $e^n = x^n - \hat{x}^n$ des ersten Autoencoders das zweite Merkmal. Die Fehler werden hierzu nicht quadriert. Jeder weitere Autoencoder hat die Aufgabe, den Fehler des vorherigen Autoencoders zu korrigieren. Auf den Fehlern e^n wird die Richtung bestimmt, wo die Fehlerbeträge am größten sind. Diese Richtung entspricht der Richtung größter Varianz des orthogonalen Unterraumes zu den bereits extrahierten Merkmalen und daher der gesuchten Richtung des nächsten Merkmals.
Mit jedem weiteren Autoencoder wird der Fehler geringer und geht letztendlich gegen Null. Dieser Prozess kann nacheinander, *sequenziell (deflationär)* oder gleichzeitig, *seriell* durchgeführt werden. Für die lineare Merkmalsextraktion funktioniert dieser Algorithmus einwandfrei, bei sequenzieller Ausführung von linearen Autoencodern erhält man als Lösung nicht nur den optimalen linearen Unterraum, sondern auch exakt die hierarchisch geordneten Merkmale der linearen PCA.
Bei der nichtlinearen Merkmalsextraktion funktioniert dieser Algorithmus leider nicht. Der Grund liegt darin, dass die Richtung der Fehlervektoren bezüglich der bereits extrahierten nichtlinearen Merkmale gesucht wird, aber nur bezüglich des originalen Datenraumes bestimmt werden kann. Für den linearen Fall spielt dies keine Rolle. Zwei Fehlervektoren gleicher Richtung im Datenraum zeigen auch bezüglich der linearen Merkmale in die gleiche Richtung. Im nichtlinearen Fall gilt dies nicht, zwei Fehlervektoren mit gleicher Richtung bezüglich der nichtlinearen Merkmale haben nicht zwangsläufig die gleiche Richtung im originalen Datenraum, wo der Fehler bestimmt wird. Die Fehler können nicht unabhängig von der nichtlinearen Abbildung betrachtet werden, siehe Abbildung 6.2.

6.3 Hierarchische Fehlerfunktion

Aus den beiden vorgestellten Ansätzen folgt, dass die Bedingung hierarchischer Ordnung der Merkmale nicht an die Varianz im Merkmalsraum $\mathcal{Z}$ gestellt werden sollte, sondern an den Rekonstruktionsfehler. Dieser muss aber bezüglich der Merkmale bewertet werden. Der Rekonstruktionsfehler darf nicht unabhängig von der nichtlinearen Abbildung $\mathcal{X} \rightarrow \mathcal{Z}$ betrachtet werden. Er muss daher in den Merkmalsraum rücktransformiert werden.
Eine solche Rücktransformation des Fehlers in den Merkmalsraum erfolgt bereits durch den Backpropagation-Algorithmus, mit dem das Netz trainiert wird. Zur Extraktion weiterer Merkmale muss daher das gleiche Netz benutzt werden, mit dem bereits Merkmale extrahiert wurden.
Der Lösungsansatz dieser Arbeit lässt sich aus der hierarchischen Bedingung ableiten. Der Einfachheit halber wird zunächst nur der zweidimensionale Fall betrachtet. Gesucht sind folglich zwei Merkmale in einem zweidimensionalen Datenraum $\mathcal{X}$. E_1 ist der Rekonstruktionsfehler unter Verwendung des ersten Merkmals und $E_{1,2}$ ist der Rekonstruktionsfehler unter Verwendung beider Merkmale. Gesucht wird ein erstes Merkmal, so dass E_1 minimal ist, und zusätzlich ein zweites Merkmal, so dass $E_{1,2}$ minimal ist. Die Lösung ist die Minimierung der Summe beider Fehler in Form einer hierarchischen Fehlerfunktion.

$$E_H = E_1 + E_{1,2}$$

In den meisten nichtlinearen Fällen ist es nicht möglich, dass beide Terme E_1 und $E_{1,2}$ gemeinsam ihr Optimum erreichen, siehe Abbildung 6.4. Die Gesamtlösung stellt daher einen Kompromiss zwischen den Einzellösungen dar.
Das Minimum der hierarchischen Fehlerfunktion kann folgendermaßen abgeschätzt werden: Einzeln könnte der Term E_1 bis zu einem Wert ξ minimiert werden, welcher vom Grad der Komplexität des verwendeten Modells (Autoencoder) abhängig ist. Der Term $E_{1,2}$ könnte, im Falle eines zweidimensionalen Datenraumes, einzeln bis auf Null minimiert werden.
Wenn E_1 den minimal möglichen Wert ξ hat, ist für den Term $E_{1,2}$ mindestens der gleiche Fehler ξ, somit $E_{1,2} \leq \xi$ und $E_H \leq 2\xi$, erreichbar.
Wenn $E_{1,2} = 0$ ist, dann kann für E_1 mindestens ein Fehlerwert erreicht werden, der dem Fehler $E_{1.PC}$ des ersten Merkmals der linearen PCA entspricht. Folglich ist für den Gesamtfehler $E_H \leq E_{1.PC}$ erreichbar.
Das Minimum der hierarchischen Fehlerfunktion ist nach unten durch den optimalen Lösungswert ξ und nach oben durch den kleineren Wert der beiden Maxima beschränkt.

$$\xi \leq E_H \leq min\{E_{1.PC}, 2\xi\}$$

Die hierarchische Fehlerfunktion ist nicht auf zwei Merkmale begrenzt, sie kann analog auf m Merkmale verallgemeinert werden.

$$E_H = E_1 + E_{1,2} + E_{1,2,3} + ... + E_{1,2,3,\ldots,m} \tag{6.1}$$

Im Sinne einer möglichst großen Ähnlichkeit zur linearen PCA sollte für die Merkmale ein Mittelwert gleich Null gefordert werden. Bis auf das erste Merkmal ist diese Bedingung bereits indirekt in der Fehlerfunktion enthalten, da die einzelnen Merkmale so extrahiert werden, dass sie im Mittel der Daten liegen. Für das erste Merkmal muss die

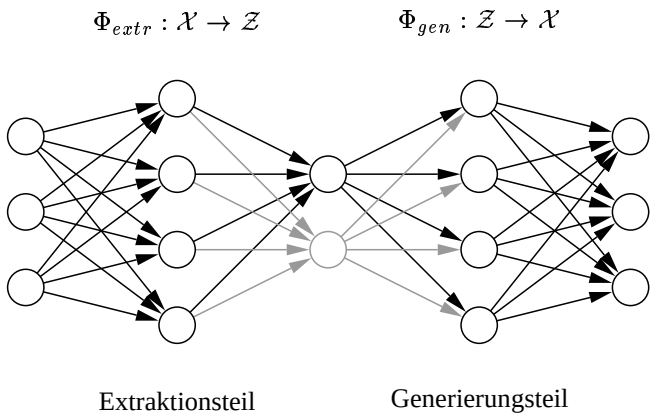


Abbildung 6.3: Hierarchischer Autoencoder bestehend aus einem [3-4-**1**-4-3]-Teilnetz (schwarz) und dem [3-4-**2**-4-3]-Gesamtnetz. In jedem Lernschritt werden die Fehler E_1 (ein Knoten in der Merkmalsschicht) und $E_{1,2}$ (zwei Knoten) jeweils einzeln berechnet. Die Optimierung der Gewichte erfolgt in jeder Iteration gemeinsam bezüglich des Gesamtfehlers $E = E_1 + E_{1,2}$.

Bedingung noch hinzugefügt werden. Dazu wird der gemeinsame Fehler $E_{2,3,\ldots,m}$ der Merkmale $2...m$ minimiert. Um die hierarchische Ordnung minimal zu beeinflussen, wird der Term $E_{2,3,\ldots,m}$ mit einem kleinen Wert γ gewichtet (z.B.: $\gamma = 0.01$).

$$E_{NLPCA} = E_H + \gamma * E_{2,3,\ldots,m}$$

Die Forderung eines Mittelwertes gleich Null erfüllt nicht nur den Zweck gleicher Eigenschaften der h-NLPCA und PCA, sondern führt auch zu besseren Ergebnissen bei der Bestimmung der nichtlinearen Merkmale. Sie beschränkt die Lösungsmenge und führt hierdurch zu eindeutigen, aussagekräftigeren Merkmalen, was sich auch vorteilhaft auf die Stabilität des Algorithmus auswirkt.

6.4 Hierarchischer Autoencoder

Die hierarchische Fehlerfunktion (6.1) wird auf den Autoencoder angewendet, in dem in jeder Iteration die einzelnen Fehlerterme E_1,...,$E_{1,\ldots,m}$ berechnet werden, mit jeweils entspechend vielen Knoten in der Merkmalsschicht. Zur Berechnung von $E_{1,\ldots,n}$ ($n < m$) wird daher nur ein Teilnetz des gesamten Autoencoders benutzt, siehe Abbildung 6.3. Die Gradienten der Gewichte werden ebenfalls zunächst einzeln berechnet. Der Gradient der hierarchischen Fehlerfunktion ∇E_H ist die Summe der Gradienten der einzelnen Fehlerterme $\nabla E_H = \nabla E_1 + ... + \nabla E_{1,..,m}$. Existiert ein w_i nicht (im Teilnetz), wird $\frac{\partial E_{1,\ldots,n}}{\partial w_i}$ auf Null gesetzt.
Um ein robusteres Verhalten zu erreichen, wird der Autoencoder bereits mit der linearen PCA-Lösung initialisiert. Die Daten x_n müssen dazu auf eine geringe Varianz (z.B.: 0.01) skaliert werden, um eine Initialisierung im linearen Bereich zu erreichen.

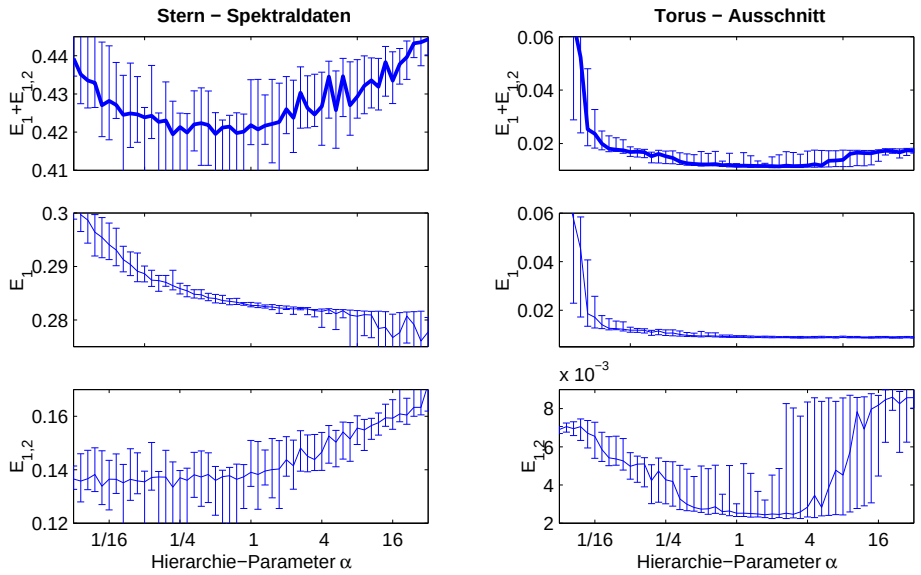


Abbildung 6.4: Fehler E_1 und $E_{1,2}$ für verschiedene α unter Verwendung von $E_H = \alpha E_1 + E_{1,2}$. Ein $\alpha = 1$ entspricht der normalen h-NLPCA, während $\alpha \to \infty$ einer NLPCA mit nur einem Merkmal und $\alpha \to 0$ einer s-NLPCA mit zwei Merkmalen entspricht. Sowohl auf den realen Stern-Spektraldaten (links) als auch auf den künstlich generierten Torus-Teildaten (rechts) ist $\alpha = 1$ eine gute Wahl.

6.5 Der Hierarchie-Parameter

Der Einfluss der einzelnen Fehlerterme kann durch einen Hierarchie-Parameter α gewichtet werden:

$$E_H = \alpha E_1 + E_{1,2} \quad ; \qquad \alpha \in (0, \infty)$$

allgemein:

$$E_H = \alpha^{m-1} E_1 + \alpha^{m-2} E_{1,2} + \alpha^{m-3} E_{1,2,3} + ... + \alpha^1 E_{1,2,\ldots,m-1} + \alpha^0 E_{1,2,\ldots,m}$$

wobei m die Gesamtzahl extrahierter Merkmale ist.
α ist ein Hyper-Parameter, welcher das Verhältnis der nichtlinearen Merkmale zueinander beeinflusst. Er bestimmt den Grad der Hierarchie der Merkmale. Von α hängt ab, wie stark die ersten Merkmale bevorzugt werden. Ein $\alpha = 0$ entspricht einer s-NLPCA, mit steigendem α wird aus der s-NLPCA kontinuierlich eine h-NLPCA.
Die Wahl eines α ist daher abhängig von der gewünschten Lösung einer h-NLPCA, ob eher viel Varianz in den ersten Merkmalen oder größtmögliche Varianz mit allen Merkmalen gefordert wird. Die Stärke der Hierarchie der h-NLPCA lässt sich folglich stufenlos variieren.
Abbildung 6.4 zeigt die Abhängigkeit der Fehlerterme E_1 und $E_{1,2}$ von α. Anhand von Stern-Spektraldaten (Kapitel 7.2.1) ist zu sehen, dass bei $\alpha = 1$ ein guter Kompromiss bezüglich der gegenläufigen Fehler E_1 und $E_{1,2}$ liegt. Auf dem künstlich generierten Datensatz aus Kapitel 5.1 erreicht h-NLPCA bei $\alpha = 1$ sogar einen kleineren Fehlerwert $E_{1,2}$ als s-NLPCA (h-NLPCA mit $\alpha \to \infty$). Die h-NLPCA bestimmt auf diesem Datensatz einen besseren zweidimensionalen Unterraum.
Verschiedene α zu testen ist sehr rechenintensiv und führte auch in anderen Experi-

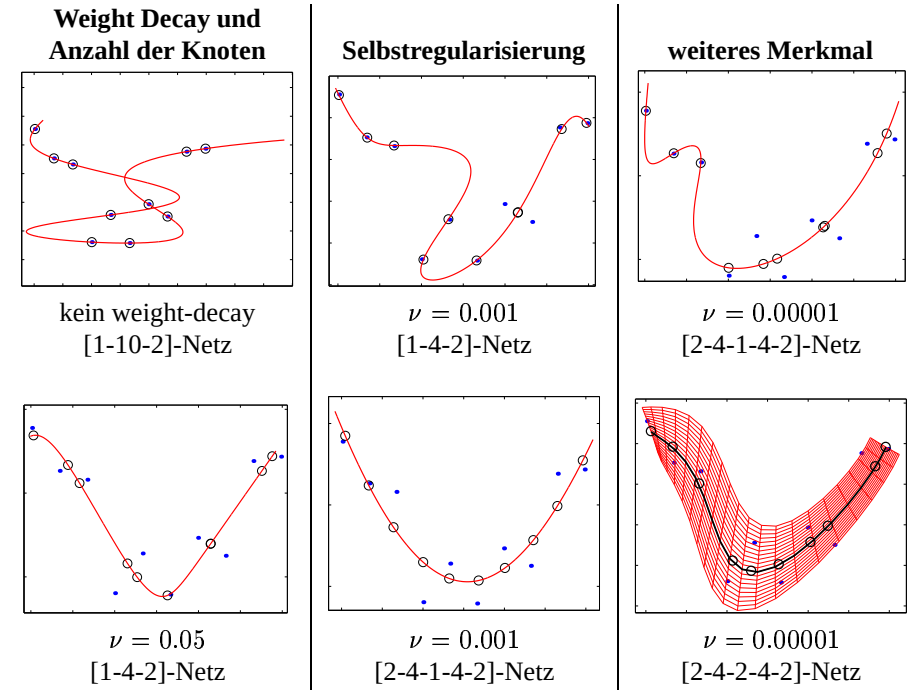


Abbildung 6.5: Regularisierungsvarianten. Oben: unzureichende Generalisierung (*overfitting*), statt die generierende Funktion zu approximieren wurde eine Funktion bestimmt, welche die Daten zum Teil exakt beschreibt, aber auf einem unabhängigen Testdatensatz ein schlechtes Ergebnis liefert. Unten: Anwendung jeweils einer der drei Regularisierungen.
Die Daten x ('$\cdot$') wurden generiert aus einer quadratischen Funktion mit additivem gaußschen Rauschen η der Stärke $\sigma = 0.4$, Die Projektion $\hat{x}$ auf das erste Merkmal ist mit '$\circ$' gekennzeichnet.

menten mit weiteren Datensätzen zu keinen signifikanten Verbesserungen; allgemein erscheint $\alpha = 1$ als eine gute Wahl. In allen in dieser Arbeit vorgestellten Experimenten wurde α auf 1 gesetzt.

6.6 Regularisierung

Die Merkmale sind nicht nur vom Grad der Hierarchie abhängig, die Komplexität des Modells, hier des Autoencoders, hat zusätzlich einen Einfluss auf die Merkmale. Bei unbegrenzter Komplexität ist es theoretisch möglich, mehrdimensionale Daten mit nur einem Merkmal zu beschreiben, welches exakt durch alle Datenpunkte geht. Solch ein Verhalten wird als *overfitting* bezeichnet und würde Daten eines unabhängigen Testdatensatzes schlecht beschreiben. Gesucht ist eine allgemeine Lösung, welche die zugrundeliegende Struktur der Daten nicht aber die des Rauschens beschreibt. Die Komplexität der Merkmale muss daher in geeigneter Weise beschränkt werden. Dies kann auf verschiedenen Wegen erfolgen, siehe auch Abbildung 6.5:

- Allgemein kann die Komplexität des Autoencoders durch einen Regularisierungsterm wie *weight decay* und durch die Anzahl der Knoten in den nichtlinearen verdeckten Schichten beschränkt werden.

Weight Decay begrenzt die Größe aller Gewichte w_i und folglich den Grad der Nichtlinearität: $E = E_H + \nu \sum_i w_i^2$, ν bestimmt den Einfluss des Weight Decay Terms, siehe auch [2].

- Der gesamte Autoencoder reguliert sich selbst. Der Autoencoder modelliert zwei zueinander inverse Funktionen. Die Komplexität einer Funktion muss durch die andere kompensiert werden, was eine deutlich höhere Komplexität erfordern kann. Dies führt beim Autoencoder zu relativ einfachen invertierbaren Funktionen mit Tendenz zu linearen Funktionen.

- Die Extraktion eines weiteren hierarchischen Merkmals beschränkt ebenfalls die Komplexität der Hauptmerkmale. Die Komplexität des Autoencoders steht nicht mehr allein den Hauptmerkmalen zur Verfügung.

Bei einer sinnvollen Nutzung der verschiedenen Regularisierungsvarianten zeigt sich in den nun folgenden Experimenten ein recht robustes Verhalten der h-NLPCA.

Kapitel 7

Experimente

Anhand verschiedener Experimente wird gezeigt, was die vorgestellte hierarchische nichtlineare PCA (h-NLPCA) leistet, und wie sie im Vergleich mit anderen modernen Methoden der nichtlinearen Dimensionsreduktion und Merkmalsextraktion abschneidet. Getestet werden die einzelnen Methoden mit klassischen PCA-Anwendungen wie Entrauschen, Vorverarbeitung und Visualisierung. Die Experimente erfolgen sowohl auf realen als auch auf künstlich generierten Datensätzen. Bei den Algorithmen handelt es sich überwiegend um nichtlineare Verallgemeinerungen der klassischen linearen PCA. Es werden keine Algorithmen der Quellentrennung betrachtet, da dies den Rahmen dieser Arbeit überschreiten würde.

7.1 Algorithmen

7.1.1 Lineare PCA

Die klassische PCA (Kapitel 2) ist eine lineare Methode. Sie ist beschränkt auf eine lineare Lösung und liefert daher auf Datensätzen mit nichtlinearer Struktur nicht das optimale Ergebnis. Eine Methode, welche als nichtlineare PCA bezeichnet wird, sollte in diesem Fall die Ergebnisse der linearen PCA übertreffen.

7.1.2 s-NLPCA

Die klassische NLPCA (Kapitel 3), basierend auf dem Autoencoder [12], ist ein reiner Dimensionsreduktions-Algorithmus, die extrahierten Merkmale besitzen keine speziellen Eigenschaften und keine Ordnung. Der Algorithmus behandelt die Merkmale gleichwertig und wird daher auch als symmetrischer Algorithmus bezeichnet (s-NLPCA).

7.1.3 h-NLPCA

Die in dieser Arbeit entwickelte h-NLPCA ist die hierarchische Erweiterung der klassischen s-NLPCA. Die h-NLPCA kann zum einen für die Dimensionsreduktions-Anwendungen der s-NLPCA eingesetzt werden, wobei aber keine besseren Ergebnisse zu erwarten sind. Ihr Vorteil liegt jedoch darin, dass es sich zum anderen um eine

Methode der Merkmalsextraktion handelt. Äquivalent zur linearen PCA können die h-NLPCA-Merkmale zum Sphering benutzt werden, welches als nichtlineares Sphering bezeichnet werden kann.

7.1.4 Kern PCA

Eine weitere nichtlineare Verallgemeinerung der PCA ist die Kern PCA [24]. Die Daten werden dabei in einen extrem hochdimensionalen Raum transformiert, in welchem eine lineare PCA ausgeführt wird. Ein Kern-Trick erlaubt dabei, die PCA-Komponenten zu bestimmen ohne direkte Berechnungen in diesem hochdimensionalen Raum auszuführen. Aufgrund der angewendeten klassischen PCA besitzen Kern PCA Merkmale im hochdimensionalen Raum eine hierarchische Ordnung. Bezüglich des originalen Datenraumes gilt dies aber nicht zwangsläufig.

Kern PCA versus h-NLPCA

Auch der Extraktionsteil $\Phi_{extr} : \mathcal{X} \rightarrow \mathcal{Z}$ der h-NLPCA kann prinzipiell als eine nichtlineare Transformation in einem höherdimensionalen Raum und einer darin ausgeführten linearen PCA betrachtet werden. Die Kern PCA und die h-NLPCA haben daher prinzipielle Ähnlichkeiten. Die Resultate weichen jedoch deutlich voneinander ab. Bei Bestimmung weniger Merkmale haben Experimente gezeigt, dass die Merkmale der h-NLPCA informationsreicher und varianzreicher sind als die Merkmale der Kern PCA. Ein wesentlicher Grund dafür ist vermutlich die unterschiedliche Verknüpfung mit der linearen PCA. Im Falle der Kern PCA erfolgt die nichtlineare Transformation unabhängig von der darauffolgenden PCA. Die nichtlineare Transformation ist variierbar durch verschiedene Kern-Typen und zugehörige Parameter. Die richtige Wahl kann nur über ein Qualitätskriterium einer Endanwendung (Entrauschen, Klassifikation) bestimmt werden. Die PCA selbst liefert kein Qualitätskriterium zurück. Im Falle der h-NLPCA existiert solch eine Rückkopplung, welche die nichtlineare Transformation beeinflusst. Zusätzliche Parameter dienen nur der Begrenzung der Komplexität.
Die Kern PCA ist andererseits sehr effizient im Umgang mit großen Datenmengen und der Extraktion vieler Merkmale. Sie hat daher ihre Stärke, wenn viele Merkmale benutzt werden können und viele Daten vorliegen. Sie ist sehr erfolgreich beim Entrauschen und als Vorverarbeitung für Klassifikationsanwendungen. Dagegen ist die h-NLPCA im Training sehr rechenintensiv und daher nur begrenzt geeignet für die Extraktion vieler Merkmale aus großen Datensätzen.
Die Kern PCA besteht nur aus der Extraktionsfunktion $\Phi_{extr} : \mathcal{X} \rightarrow \mathcal{Z}$. Zum Entrauschen wird zusätzlich die Generierungsfunktion $\Phi_{gen} : \mathcal{Z} \rightarrow \mathcal{X}$ benötigt, welche bei der Kern PCA nur in Form eines Optimierungsalgorithmus mit entsprechendem Rechenaufwand existiert. Beim Entrauschen ist die Kern PCA im Training effizient, in der Anwendung dagegen rechenintensiver. Bei der h-NLPCA und auch bei der s-NLPCA ist das Training sehr rechenintensiv, die Anwendung dagegen sehr effizient.

7.1.5 LLE — Locally Linear Embedding

Eine der wesentlichsten Forderungen in der Dimensionsreduktion ist die Erhaltung der Nachbarschaftsbeziehungen. Locally Linear Embedding (LLE) [19] ist ein Dimensionsreduktions-Algorithmus, welcher ein solches Nachbarschaftskriterium optimiert. Ein Parameter k bestimmt dabei die Anzahl der betrachteten Nachbarn. Der

Algorithmus ist nicht hierarchisch konzipiert, liefert jedoch in einigen Fällen eine Lösung mit bestimmter, zum Teil hierarchischer Ordnung der Merkmale. Die Ursache für die stark vom jeweiligen Datensatz abhängige Ordnung ist noch nicht vollständig geklärt.
Der Algorithmus liefert für verschiedene k sehr interessante, aber auch sehr unterschiedliche Lösungen. Ein Kriterium zur optimalen Wahl des Parameters k ist bisher nicht vorhanden. Eine erste Version des Algorithmus ist nur als einmalige Abbildung $\mathcal{X} \rightarrow \mathcal{Z}$ gegeben. Dabei wird keine Extraktionsfunktion $\Phi_{extr} : \mathcal{X} \rightarrow \mathcal{Z}$ für eine Anwendung auf weiteren Daten bestimmt. Auch eine Rekonstruktion der Daten mit einer Generierungsfunktion $\Phi_{gen} : \mathcal{Z} \rightarrow \mathcal{X}$ ist hierbei nicht möglich. Dies erschwert die Bestimmung eines optimalen Parameters k und beschränkt diesen Algorithmus auf Visualisierungsanwendungen.
Eine neuere Arbeit [20] beschreibt daher ein Wahrscheinlichkeitsmodell, mit dem die Abbildungen $\mathcal{X} \rightarrow \mathcal{Z}$ und $\mathcal{Z} \rightarrow \mathcal{X}$ bestimmt werden können. Hiermit könnte der Algorithmus auch zum Entrauschen von Daten und zur Komprimierung eingesetzt werden.

7.2 Datensätze

7.2.1 Stern-Spektraldaten

Bei dem Stern-Spektraldatensatz handelt es sich um Spektraldaten von 487 Sternen aus 6 verschiedenen Sternenklassen. Von jedem Stern wurde der gleiche Spektralbereich gemessen und die Äquivalenzbreiten von 19 verschiedenen Absorptionslinien bestimmt. Zusätzlich sind zu jedem Stern verschiedene physikalische Größen bekannt: die absolute Helligkeit Mv, die Eigenfarbe B-V sowie der Metallgehalt Fe/H.
Vorrangiges Ziel ist die Bestimmung dieser physikalischen Sternengrößen aus den 19 Spektralwerten. Dabei ist die absolute Helligkeit von besonderer Bedeutung. Eine Bestimmung der absoluten Helligkeit aus 19 Spektralwerten eines unbekannten Sternes kann zur Entfernungsschätzung benutzt werden. Die Entwicklung eines Modells auf Basis neuronaler Netze zur Bestimmung der Sternengrößen fand im Rahmen eines Praktikums in Zusammenarbeit mit dem astronomischen Institut *CIDA - Centro de Investigaciones de Astronomía* in Venezuela statt.
In dieser Arbeit steht die Datenstruktur des 19-dimensionalen Spektralraumes im Vordergrund, da es sich hier um eine deutlich nichtlineare Struktur handelt.
Eine detailliertere Beschreibung des Datensatzes ist in [26] zu finden.

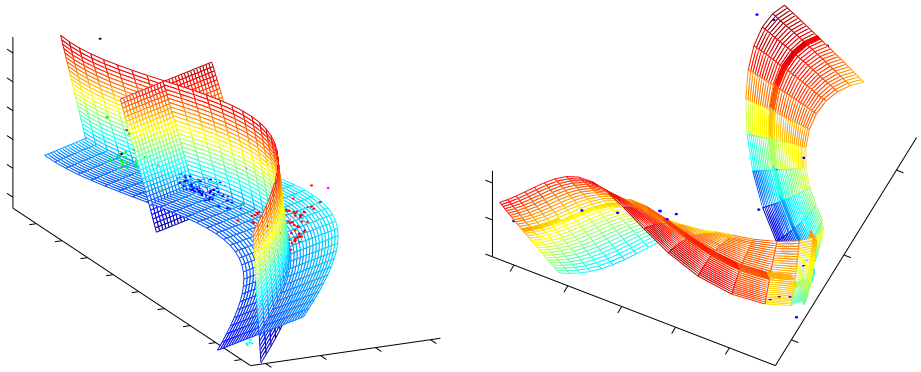
7.2.2 EMG - Datensatz

Dieser Datensatz basiert auf elektomyographischen (EMG) Aufnahmen verschiedener Muskelaktivitäten. Es sind bei 7 Versuchspersonen jeweils 5 verschiedene Kräfteniveaus gemessen worden: 0%, 10%, 30%, 50% und 70% der persönlichen Maximalkraft. Die jeweils eindimensionalen EMG-Zeitaufnahmen wurden in einen 17-dimensionalen Raum eingebettet, welcher wiederum mit der *recurrence qualification analysis* (RQA) [28] ausgewertet wurde. Aus dem dabei erstellten *recurrence plot* wurden 10 *recurrence* Attribute abgeleitet. Der endgültige EMG-Datensatz besteht somit aus 10 Attributwerten (10-dimensional) für jeweils 35 Beispiele (5 Kräfteniveaus für jeden der 7 Probanden). Eine detailliertere Beschreibung ist in [15] zu finden. Die Auswertung der Daten erfolgte im Rahmen einer Zusammenarbeit mit *Flinders University, Australia* [25].

Der EMG-Datensatz besitzt sehr interessante nichtlineare Korrelationen und eignet sich ebenfalls sehr gut zur Beurteilung der Qualität (Informationsgehalt) der extrahierten Merkmale.

7.2.3 Klassifikationsdatensatz

Zur qualitativen Bewertung der Vorverarbeitungsleistung nichtlinearer PCA-Techniken wurde ein künstlicher Datensatz generiert.
Der Datensatz besteht aus 10000 Beispielen in 3 Dimensionen. Die Klassifikationsaufgabe ist einer Einteilung in männlich (M) und weiblich (F) sowie in Spezies A und Spezies B nachempfunden. Jedes Datum ist einer der vier Klassen-Kombinationen zugeordnet: 'F,A', 'F,B', 'M,A' oder 'M,B'. Die Daten sind nichtlinear korreliert. Die Hauptkrümmung ist gegeben durch die Funktion $y = x^2$ und $z = \tanh(x)$. Senkrecht bezüglich dieses nichtlinearen Merkmals besitzen die Daten eine Gaußverteilung.



Stern - Spektraldaten EMG - Daten

Abbildung 7.1: Dargestellt sind die ersten drei nichtlinearen Merkmale der h-NLPCA als Gitternetze im PCA-Unterraum, gegeben durch die ersten drei linearen PCA-Merkmale. Jedes Gitternetz repräsentiert zwei nichtlineare Merkmale, das jeweilige dritte Merkmal ist auf Null gesetzt. Für den Stern-Spektraldatensatz (19x487) und den EMG-Datensatz (10x35) wurden entsprechend ein [19-30-10-30-19] und ein [10-7-3-7-10] Netz benutzt.

7.3 Visualisierung

Lineare PCA wird häufig auch zur Visualisierung eingesetzt, dabei steht eine Projektion hochdimensionaler Daten auf zwei oder drei Dimensionen im Vordergrund. Ist die globale Verteilung aller Daten von Interesse, so ist mit einer h-NLPCA bei dieser Art der Visualisierung kein Vorteil zu erwarten. Die bedeutende Eigenschaft der h-NLPCA, nichtlinear unkorrelierte Merkmale zu extrahieren, kann von Nachteil sein, da interessante nichtlineare Korrelationen entfernt werden.
Ist dagegen die Klassenverteilung von Interesse, kann mit der h-NLPCA eine detailliertere Darstellung erreicht werden. Abbildung 7.2 zeigt zweidimensionale Projektionen der 19-dimensionalen Stern-Spektraldaten bei Verwendung unterschiedlicher Methoden.
Die h-NLPCA ermöglicht eine weitere Art der Visualisierung. Die nichtlinearen Merkmale können im originalen Datenraum oder in einem linearen PCA-Unterraum grafisch dargestellt werden, siehe Abbildung 7.1. Die Generierungsfunktion $\Phi_{gen} : \mathcal{Z} \rightarrow \mathcal{X}$ wird dabei als Gitternetz dargestellt, welches den Merkmalsraum repräsentiert. Diese Darstellung der nichtlinearen Merkmale ist hilfreich bei der visuellen Bewertung der Datenverteilung. Die verdrehte Struktur des EMG-Datensatzes wäre ohne die Gitternetz-Darstellung nur schwer erkennbar.

7.4 Nichtlineares Sphering

Ziel des Sphering ist, eine sphärische Verteilung der Daten zu erreichen, siehe Kapitel 2.4. Eine Skalierung linear unkorrelierter PCA-Merkmale auf einheitliche Varianz führt zu linearem Sphering. Eine Skalierung nichtlinear unkorrelierter Merkmale kann daher als nichtlineares Sphering betrachtet werden.
Verschiedene Merkmalsextraktions-Methoden werden daraufhin untersucht, wie gut

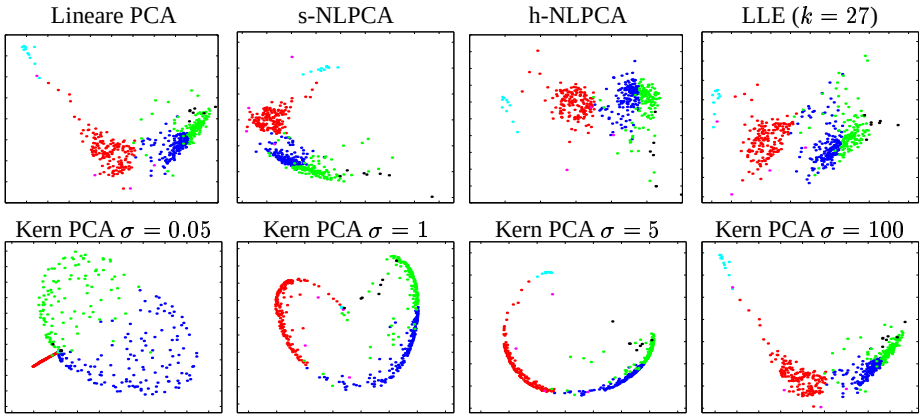


Abbildung 7.2: Sphering-Experiment. Verglichen werden zweidimensionale Merkmalsräume verschiedener Methoden der Merkmalsextraktion. Die Aufgabe besteht in der Beseitigung linearer und nichtlinearer Korrelationen. Die h-NLPCA liefert dabei das beste Resultat.

mit ihnen nichtlineare Korrelationen entfernt werden können. Betrachtet wird der zweidimensionale Merkmalsraum, gegeben durch die ersten beiden Merkmale der jeweiligen Merkmalsextraktions-Methode. Die Merkmale werden auf einheitliche Varianz skaliert. Als Datensatz wurde der Stern-Spektraldatensatz benutzt, da er nichtlineare Korrelationen aufweist.
Abbildung 7.2 zeigt die Resultate der verschiedenen Methoden. Lineares Sphering (lineare PCA) beseitigt nur lineare Korrelationen ($\mathrm{E}\{zz^T\} = \mathrm{I}$). Die nichtlineare Korrelation wird am besten von der h-NLPCA beseitigt. Die Clusterverteilung bleibt dabei erhalten. LLE erreicht nicht das Ergebnis der h-NLPCA, liefert aber ein besseres Ergebnis als die lineare PCA. Die s-NLPCA entfernt aufgrund der Symmetrie keine Korrelationen. Als Ergebnis ist auch jede beliebige Rotation des abgebildeten s-NLPCA-Ergebnisses möglich. Kern PCA liefert sehr unterschiedliche Ergebnisse mit stark veränderter Datenverteilung. Ein qualitativer visueller Vergleich mit den anderen Methoden ist daher schwierig.

7.5 Informationsgehalt der Merkmale

Anhand des EMG-Datensatzes wird der Informationsgehalt der Merkmale verschiedener Methoden bewertet. Der Datensatz beinhaltet ein nichtlineares Merkmal, welches mit der Muskelkraft korreliert und deutlich die größte Varianz aufweist. Die Qualität der extrahierten Merkmale wird daher mit der Korrelation zur Muskelkraft bewertet.
Bei Kern PCA und LLE wurden die visuell besten Ergebnisse ausgewählt. Sinnvolle Ergebnisse erzielt die Kern PCA ab einem $\sigma > 3$. Mit steigendem σ nähert sich die Kern PCA einer linearen Lösung, die der linearen PCA, an. PCA hat keinen veränderlichen Parameter und liefert daher genau ein Ergebnis. Bei der h-NLPCA wurden die optimalen Komplexitätsparameter durch Kreuzvalidierung bestimmt. Es zeigte sich jedoch aufgrund der schwierigen Struktur eine starke Abhängigkeit von der zufällig gewählten Startinitialisierung der Gewichte. Ausgewählt wurde daher das Ergebnis mit dem geringsten Rekonstruktionsfehler von fünf Trainingsläufen mit unterschiedlichen Startinitialisierungen. Abbildung 7.3 zeigt, dass h-NLPCA ein erstes nichtlineares Merkmal extrahiert, welches im Vergleich zu den anderen Methoden am besten (annä-

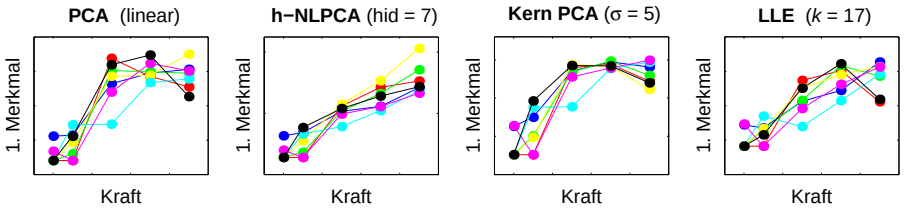


Abbildung 7.3: Dargestellt ist das erste Merkmal (y-Achse) verschiedener Methoden der Merkmalsextraktion gegen die Muskelkraft (x-Achse). Die zu einer Person gehörenden Daten sind zur Kennzeichnung durch Linien verbunden. Das Merkmal der h-NLPCA ist am informationsreichsten bezüglich der Muskelkraft (annähernd lineare Korrelation). LLE liefert auch ein gutes Ergebnis, die Merkmale der Kern PCA und der linearen PCA sind deutlich schlechter.

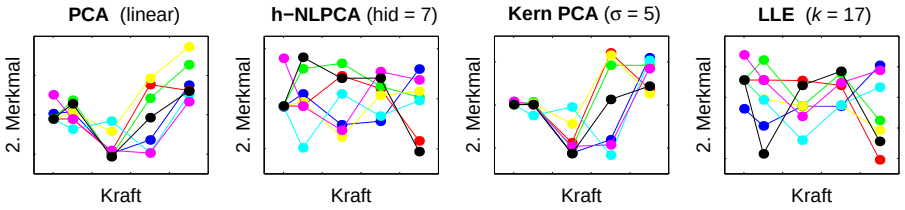


Abbildung 7.4: Hier ist das zweite Merkmal gegen die Muskelkraft dargestellt. Bei der linearen PCA und der Kern PCA ist eine Abhängigkeit des zweiten Merkmals von der Muskelkraft sichtbar, was auf ein unzureichendes erstes Merkmal zurückzuführen ist. Bei der h-NLPCA und bei LLE ist kaum eine Abhängigkeit erkennbar. Das erste Merkmal beinhaltet bereits alle Informationen über die Muskelkraft, das zweite Merkmal korreliert vermutlich mit einem anderen, bisher unbekannten, physiologischen Parameter.

hernd linear) mit der Muskelkraft korreliert. Das zweite Merkmal (Abbildung 7.4) der h-NLPCA korreliert nicht mit der Muskelkraft, es weist jedoch eine höhere Varianz auf als die folgenden Merkmale. Daher kann vermutet werden, dass es sich hierbei nicht um Rauschen handelt. Das zweite Merkmal beschreibt vermutlich einen weiteren physiologischen Parameter, der aber noch nicht zugeordnet werden konnte.
Auch die s-NLPCA ist in der Lage, ein entsprechend gutes erstes Merkmal zu extrahieren. Die s-NLPCA ist aber auf dieses eine Merkmal beschränkt. Es können keine weiteren sinnvollen Merkmale wie bei der h-NLPCA extrahiert werden.
LLE liefert auch ein relativ gutes Ergebnis. Auch das zweite LLE-Merkmal ähnelt dem der h-NLPCA. Die Kern PCA und die lineare PCA sind für diesen Anwendungsfall weniger geeignet.

7.6 Entrauschen

Voraussetzung für das Entrauschen mittels einer PCA ist eine große Varianz der relevanten Informationen und eine geringe Varianz des Rauschanteils. Durch eine Projektion der Daten auf einen Unterraum großer Varianz und hiermit verbunden ein Beseitigen der Dimensionen geringer Varianz können die Daten entrauscht werden. Entscheidend ist dabei die Dimension des Unterraumes (Anzahl der Merkmale) und die Art der Extraktion (linear, nichtlinear).
Das hier betrachtete Entrauschen ist folglich eine reine Dimensions-

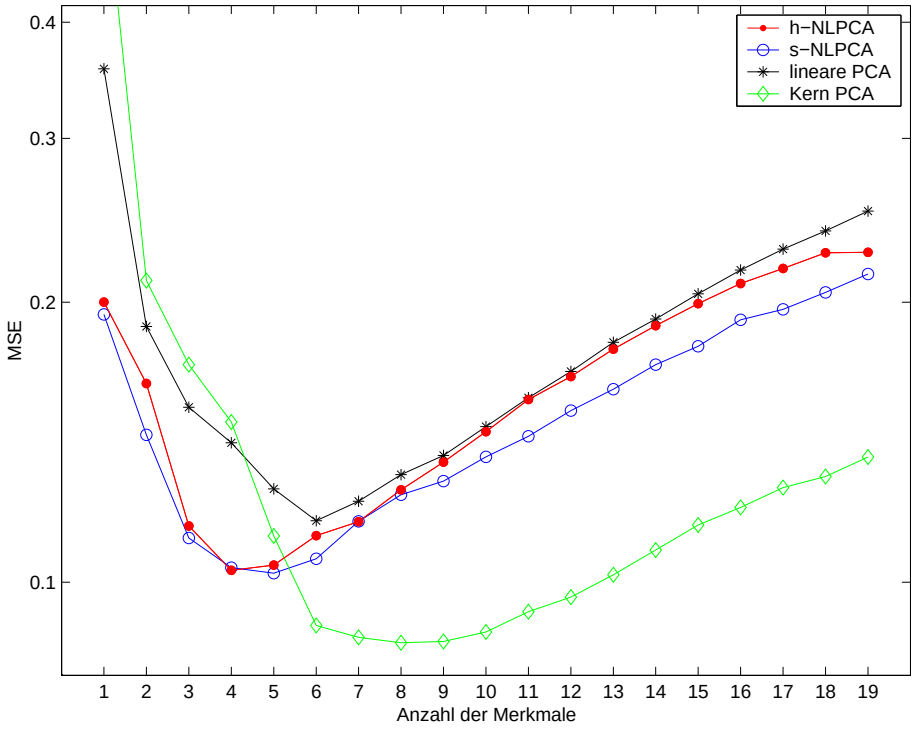


Abbildung 7.5: Entrauschungsergebnisse bei unterschiedlicher Anzahl verwendeter Merkmale. Die verschiedenen Methoden wurden auf die Stern-Spektraldaten angewendet, welche dazu künstlich mit additivem gaußschen Rauschen η (std $\sigma = 0.5$) verfremdet wurden. Der Stern-Spektraldatensatz wurde in einen Trainings- und einen Testdatensatz aufgeteilt. Dargestellt ist der mittlere quadratische Fehler auf den Testdaten.

reduktions-Anwendung, durch eine zusätzliche hierarchische Eigenschaft ist daher kein besseres Ergebnis zu erwarten. Entrauschen gehört aber zu den klassischen PCA-Anwendungen und wird daher in den Experimenten mit berücksichtigt.
Betrachtet wird der Fall, dass nur verrauschte Daten für die verschiedenen Methoden zur Anwendung kommen. Bewertet werden die Methoden anhand unverrauschter Daten. Der Stern-Spektraldatensatz wurde hierzu künstlich mit einem additivem gaußschen Rauschen η der Stärke $\sigma = 0.5$ verrauscht, dies entspricht einem Signal-Rausch Verhältnis von 26 dB. Um die Methoden auf einem 'unbekannten' Testdatensatz zu bewerten, wurde der Stern-Spektraldatensatz in einen Trainingsdatensatz (19x244) und einen Testdatensatz (19x243) aufgeteilt. Zum Trainieren wurde der verrauschte Trainingsdatensatz $(x_{train} + \eta)$ verwendet. Bewertet wurde der mittlere quadratische Fehler der entrauschten Testdaten $\hat{x}_{test} = \Phi_{gen}(\Phi_{extr}(x_{test} + \eta))$ bezüglich der originalen unverrauschten Testdaten x_{test}:

$$\mathrm{MSE} = \frac{1}{dN} \sum_{n}^{N} \sum_{k}^{d} |x^n_{k_{(test)}} - \hat{x}^n_{k_{(test)}}|^2$$

Um ein optimales Ergebnis der s-NLPCA zu erreichen, wird sie für jede Merkmalszahl einzeln ausgeführt, während mit der h-NLPCA alle 19 Merkmale gemeinsam optimiert

	Klassifikation 'F' zu 'M'						
# Merkmale	1	**2**	3	4	5	10	20
linear PCA	50.0	**28.4**	28.4	—	—	—	—
s-NLPCA	45.9	**31.3**	—	—	—	—	—
h-NLPCA	50.0	**11.4**	11.4	8.1	4.3	—	—
LLE $k = 5$	45.8	**35.1**	5.5	—	—	—	—
kPCA $\sigma = 10$	49.3	**34.6**	9.4	9.0	9.0	3.6	3.5
kPCA $\sigma = 0.5$	49.3	**41.8**	41.9	28.6	7.7	3.2	1.2

Tabelle 7.1: Klassifikationsfehler in % auf Testdaten eines künstlich generierten Datensatzes. Benutzt wurde eine lineare Support Vektor Maschine, trainiert auf den n ersten Merkmalen verschiedener Merkmalsextraktions-Algorithmen. Interessant ist die Klassifikationsrate auf zwei Merkmalen, da zur Klassifikation von 'F' und 'M' ein korrektes zweites Merkmal ausreichend ist. Auf zwei Merkmalen ist die h-NLPCA führend.

werden.
In Abbildung 7.5 ist zu sehen, dass die s-NLPCA und die h-NLPCA mit wenigen Merkmalen ein deutlich besseres Ergebnis erreichen als die lineare PCA. Die wesentliche Information ist in wenigen nichtlinearen Merkmalen komprimiert.
Die s-NLPCA und die h-NLPCA liefern zum Teil ähnliche Ergebnisse, wobei die s-NLPCA häufig etwas besser ist. Die h-NLPCA und die s-NLPCA könnten theoretisch das gleiche Ergebnis liefern, da sie das gleiche Kriterium eines optimalen Unterraumes minimieren. Die h-NLPCA besitzt aber zusätzlich das hierarchische Kriterium, welches nicht zum Entrauschen benötigt wird. Es kann vorkommen, dass sich die beiden Kriterien gegenseitig beeinträchtigen, was zu einem schlechteren Ergebnis der h-NLPCA führen würde. Andererseits kann auch gezeigt werden, dass in bestimmten Fällen (unterschiedliche Varianz, Kapitel 5.1) die hierarchische Bedingung hilfreich sein kann bei der Bestimmung des optimalen Unterraumes, und in diesen Fällen die h-NLPCA ein besseres Ergebnis liefert, siehe auch Abbildung 6.4.
Da die h-NLPCA eine Erweiterung der s-NLPCA ist, macht es Sinn, beim Entrauschen die h-NLPCA mit verschiedenen Werten des Hierarchie-Parameters α zu testen, ausgehend von einer h-NLPCA mit $\alpha = 0$ (entspricht einer s-NLPCA) bis zur normalen h-NLPCA mit $\alpha = 1$.
Die Kern PCA erreicht mit einem Gaußkern ($\sigma = 2$) ein deutlich besseres Ergebnis als alle anderen verwendeten Methoden. Sie benötigt dafür aber eine größere Anzahl von Merkmalen. Auf wenigen Merkmalen ist das Ergebnis sogar schlechter. Die Kern PCA ist zum Komprimieren der wesentlichen Informationen in wenigen Merkmalen nicht so gut geeignet, zum Entrauschen von Daten eignet sie sich dagegen sehr gut.

7.7 Klassifikation

Anhand des künstlich generierten Datensatzes (Kapitel 7.2.3) wird die Vorverarbeitungsleistung der verschiedenen Methoden verglichen. Die Aufgabe besteht in der Beseitigung nichtlinearer Korrelationen, so dass die Klassen im Merkmalsraum linear separierbar sind. Der Datensatz wurde so generiert, dass sich bezüglich der Varianz drei deutlich unterscheidbare nichtlineare Merkmale ergeben. Entlang des ersten nichtli-

	Klassifikation 'A' zu 'B'						
# Merkmale	1	2	**3**	4	5	10	20
linear PCA	44.1	44.6	**30.9**	—	—	—	—
s-NLPCA	50.0	50.0	**—**	—	—	—	—
h-NLPCA	49.2	49.4	**9.3**	8.8	6.0	—	—
LLE $k = 5$	49.9	47.8	**46.5**	—	—	—	—
kPCA $\sigma = 10$	49.0	48.6	**48.0**	34.6	29.5	13.1	13.1
kPCA $\sigma = 0.5$	51.2	51.2	**49.7**	49.9	48.0	36.3	1.8

Tabelle 7.2: Bei der Klassifikation in 'A' und 'B' ist ein korrektes drittes nichtlineares Merkmal ausreichend. Zur qualitativen Bewertung der Merkmale ist hier die Klassifikationsrate auf drei Merkmalen von Bedeutung. Die h-NLPCA Merkmale sind auch hierbei deutlich besser.

	Klassifikation 'F','M','A' und 'B'						
# Merkmale	1	2	**3**	4	5	10	20
linear PCA	70.9	61.9	**51.0**	—	—	—	—
s-NLPCA	73.6	65.3	**—**	—	—	—	—
h-NLPCA	74.4	54.6	**20.0**	16.2	10.3	—	—
LLE $k = 5$	73.0	66.3	**50.0**	—	—	—	—
kPCA $\sigma = 10$	74.0	66.5	**53.3**	42.2	37.8	16.5	16.4
kPCA $\sigma = 0.5$	75.9	71.5	**71.2**	64.9	51.5	39.2	3.0

Tabelle 7.3: Absolute Klassifikation aller 4 Gruppen 'F,A','F,B','M,A' und 'M,B'. Zwei lineare Klassifikationsgrenzen werden benutzt. Die Fehlklassifikationsrate beträgt bei zufälliger Zuordnung 75%. Das zweite und das dritte nichtlineare Merkmal sind zur Klassifikation ausreichend. Entscheidend ist daher die Klassifikationsrate auf den ersten drei Merkmalen.

nearen Merkmals ist keine Klassifikation möglich, entlang des zweiten Merkmals kann zwischen M und W unterschieden werden und mit dem dritten zwischen A und B. Zur vollständigen Klassifikation ist daher eine korrekte Extraktion des zweiten und des dritten nichtlinearen Merkmals ausreichend. Da aber die Klassifikationsgrenze entlang der höchsten Datendichte liegt, hat eine geringe Ungenauigkeit der extrahierten Merkmale eine hohe Fehlklassifikation zur Folge.
Zur linearen Klassifikation wurde eine lineare *Support Vektor Maschine* [27] benutzt. Die extrahierten Merkmale wurden vorher auf einheitliche Varianz skaliert.
Tabellen 7.1 bis 7.3 zeigen die Resultate der einzelnen Methoden. Die ersten Merkmale werden von der h-NLPCA deutlich besser bestimmt als von den anderen Methoden.
Die Kern PCA bestimmt die ersten Merkmale weniger gut. Sie erlaubt aber, mehr Merkmale als Dimensionen zu extrahieren und erreicht auf dieser großen Anzahl von Kern PCA Merkmalen ein deutlich besseres Klassifikationsresultat. Im Gegensatz zur s-NLPCA ist auch die h-NLPCA in der Lage, mehr Merkmale zu extrahieren als der originale Datenraum Dimensionen besitzt. Von diesen weiteren Merkmalen ist nicht zwangsläufig ein Informationsgewinn zu erwarten. Sie korrigieren vielmehr die unzureichend bestimmten ersten Merkmale. Dieses Experiment zeigt aber, dass mit weiteren Merkmalen die Klassifikation verbessert werden kann. Mit vielen extrahierten

h-NLPCA Merkmalen sind vermutlich ähnlich gute Ergebnisse zu erwarten wie bei der Kern PCA. Die h-NLPCA ist jedoch weitaus rechenintensiver als die Kern PCA.

Kapitel 8

Zusammenfassung

In dieser Diplomarbeit wurde eine hierarchische nichtlineare PCA (h-NLPCA) vorgestellt, mit weitgehend identischen Eigenschaften zur linearen PCA. Sie basiert auf einer hierarchischen Erweiterung der Fehlerfunktion. Die h-NLPCA extrahiert nicht nur das erste nichtlineare Merkmal, was mit anderen Methoden auch möglich ist, sondern sie ist auch in der Lage, zusätzlich weitere nichtlineare Merkmale zu extrahieren. Die h-NLPCA kann zur Beschreibung von Daten und zur Informationsgewinnung benutzt werden, was am Beispiel der EMG-Daten veranschaulicht wurde. Darüber hinaus konnte gezeigt werden, dass mit wenigen h-NLPCA Merkmalen eine bessere Klassifikation erreicht werden konnte als mit Merkmalen der Kern PCA oder Merkmalen anderer Methoden. Eine weitere Anwendung ist die Entfernung nichtlinearer Korrelationen als Vorverarbeitungsschritt beispielsweise für die Quellentrennung. Vielversprechend wäre, zukünftig die hierarchische nichtlineare PCA direkt zu einem nichtlinearen Algorithmus der Quellentrennung zu erweitern.

Literaturverzeichnis

[1] P. Baldi and K. Hornik. Neural networks and principal component analysis: Learning from examples without local minima. *Neural Networks*, 2:53 – 58, 1989.

[2] C. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.

[3] H. Bourlard and Y. Kamp. Auto-association by multilayer perceptrons and singular value decomposition. *Biological Cybernetics*, 59:291 – 294, 1988.

[4] M. Á. Carreira-Perpiñán. A review of dimension reduction techniques. Technical Report CS-96-09, Dept. of Computer Science, University of Sheffield, 1997.

[5] K. Diamantaras and S. Kung. *Principal Component Neural Networks*. Wiley, New York, 1996.

[6] K. Fukunaga. *Introduction to Statistical Pattern Recognition*. Academic Press, San Diego, 2nd edition, 1990.

[7] M. H. Hassoun and A. Sudjianto. Compression net-free autoencoders. *Workshop on Advances in Autoencoder/Autoassociator-Based Computations at the NIPS 97 Conference*, Dec.6 1997.

[8] T. Hastie and W. Stuetzle. Principal curves. *Journal of the American Statistical Association*, 84(406):502–516, June 1989.

[9] M. R. Hestenes and E. Stiefel. Methods of conjugate gradients for solving linear systems. *Journal of Research of the National Bureau of Standards*, 49 (6):409–436, 1952.

[10] A. Hyvärinen, J. Karhunen, and E. Oja. *Independent Component Analysis*. J. Wiley, 2001.

[11] J. Karhunen and J. Joutsensalo. Generalizations of principal component analysis, optimization problems, and neural networks. *Neural Networks*, 8(4):549–562, 1995.

[12] M. Kramer. Nonlinear principal component analysis using auto-associative neural networks. *AIChE Journal*, 37(2):233–243, 1991.

[13] S. Y. Kung, K. I. Diamantaras, and J. S. Taur. Adaptive principal component extraction (APEX) and applications. *IEEE Trans. Signal Processing*, 42:1202 – 1217, 1994.

[14] H. Lappalainen and A. Honkela. Bayesian nonlinear independent component analysis by multi-layer perceptrons. *In Advances in Independent Component Analysis, ed. by M. Girolami*, pages 93 – 121, 2000.

[15] D. T. Mewett, K. J. Reynolds, and H. Nazeran. Principal components of recurrence quantification analysis of EMG. *Proceedings of the 23rd Annual IEEE/EMBS Conference*, Oct.25-28 2001.

[16] S. Mika, B. Schölkopf, A. Smola, K.-R. Müller, M. Scholz, and G. Rätsch. Kernel PCA and de–noising in feature spaces. In M. Kearns, S. Solla, and D. Cohn, editors, *Advances in Neural Information Processing Systems 11*, pages 536–542. MIT Press, 1999.

[17] E. Oja. Principal components, minor components, and linear neural networks. *Neural Networks*, 5:927 – 935, 1992.

[18] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C: The Art of Scientific Computing (2nd ed.)*. Cambridge University Press, Cambridge, 1992. ISBN 0-521-43108-5.

[19] S. Roweis and L. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290:2323 – 2326, Dec.22 2000.

[20] S. Roweis, L. Saul, and G. Hinton. Global coordination of local linear models. *Neural Information Processing Systems 14 (NIPS'01)*, 2001. to appear.

[21] D. Rumelhart, G. Hinton, and R. Williams. Learning representations by back-propagating errors. *Nature*, 323(9):533–536, October 1986.

[22] T. Sanger. Optimal unsupervised learning in a single-layer linear feedforward network. *Neural Networks*, 2:459–473, 1989.

[23] B. Schölkopf, S. Mika, C. Burges, P. Knirsch, K.-R. Müller, G. Rätsch, and A. Smola. Input space vs. feature space in kernel-based methods. *IEEE Transactions on Neural Networks*, 10(5):1000–1017, September 1999.

[24] B. Schölkopf, A. Smola, and K.-R. Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10:1299–1319, 1998.

[25] M. Scholz and R. Vigário. Nonlinear PCA: a new hierarchical approach. In M. Verleysen, editor, *Proceedings ESANN*, 2002.

[26] J. Stock and M. J. Stock. Quantitative stellar spectral classification. *Revista Mexicana de Astronomia y Astrofisica*, 34:143 – 156, 1999.

[27] V. Vapnik. *The nature of statistical learning theory*. Springer Verlag, New York, 1995.

[28] C. L. Webber Jr and J. P. Zbilut. Dynamical assessment of physiological systems and states using recorrence plot strategies. *Journal of Applied Physiology*, 76:965 – 973, 1994.

Index