# Analyse dünnbesetzter Hauptachsen für Frequenzdaten

Tobias Bork

Geboren am 21. November 1997 in Reutlingen

24. Oktober 2019

Bachelorarbeit Mathematik

Betreuer: Prof. Dr. Jochen Garcke

Zweitgutachter: Prof. Dr. X Y

Mathematisches Institut für Numerische Simulation

Mathematisch-Naturwissenschaftliche Fakultät der

Rheinischen Friedrich-Wilhelms-Universität Bonn

# *Danksagung*

The acknowledgments and the people to thank go here, don't forget to include your project advisor…

# Inhaltsverzeichnis

# Kapitel 1

# Einführung

[3] [9] [13] [5] [8] [6] [15] [7] [10] [12] [16] [17] [18] [11]

## 1.1 Motivation

So ist man meist besonders an der Bildung sog. Cluster, also Gruppierungen, interessiert. Datenpunkte, die im entstehendem Bild nach Anwendung der Hauptkomponentenanalyse nah beieinander liegen, sind in gewisser Weise ähnlich zueinander während Datenpunkte, die weit von einander entfernt liegen, wenig Ähnlichkeit aufweisen. Abbildung CITE zeigt die Entstehung solcher Cluster auf dem Datensatz. Mit diesem Verfahren lässt sich daher eine Art Struktur in den Daten erkennen, die für weitere Analysezwecke ausgenutzt werden kann.

The goals of PCA are to

(1)

extract the most important information from the data table; (2)

compress the size of the data set by keeping only this important information; (3)

simplify the description of the data set; and (4)

analyze the structure of the observations and the variables.

## 1.2 Dimensionsreduktionsverfahren

High dimensionality means that the dataset has a large number of features. The primary problem associated with high-dimensionality in the machine learning field is model overfitting, which reduces the ability to generalize beyond the examples in the training set. Richard Bellman described this phenomenon in 1961 as the Curse of Dimensionality where “Many algorithms that work fine in low dimensions become intractable when the input is high-dimensional. “

Let’s say that you want to predict what the gross domestic product (GDP) of the United States will be for 2017. You have lots of information available: the U.S. GDP for the first quarter of 2017, the U.S. GDP for the entirety of 2016, 2015, and so on. You have any publicly-available economic indicator, like the unemployment rate, inflation rate, and so on. You have U.S. Census data from 2010 estimating how many Americans work in each industry and American Community Survey data updating those estimates in between each census. You know how many members of the House

and Senate belong to each political party. You could gather stock price data, the number of IPOs occurring in a year, and how many CEOs seem to be mounting a bid for public office. Despite being an overwhelming number of variables to consider, this just scratches the surface. TL;DR — you have a lot of variables to consider. If you've worked with a lot of variables before, you know this can present problems. Do you understand the relationships between each variable? Do you have so many variables that you are in danger of overfitting your model to your data or that you might be violating assumptions of whichever modeling tactic you're using? You might ask the question, "How do I take all of the variables I've collected and focus on only a few of them?" In technical terms, you want to "reduce the dimension of your feature space." By reducing the dimension of your feature space, you have fewer relationships between variables to consider and you are less likely to overfit your model. (Note: This doesn't immediately mean that overfitting, etc. are no longer concerns — but we're moving in the right direction!) Somewhat unsurprisingly, reducing the dimension of the feature space is called "dimensionality reduction." There are many ways to achieve dimensionality reduction, but most of these techniques fall into one of two classes: Feature Elimination Feature Extraction

2. Why is Dimensionality Reduction required? Here are some of the benefits of applying dimensionality reduction to a dataset:

Space required to store the data is reduced as the number of dimensions comes down Less dimensions lead to less computation/training time Some algorithms do not perform well when we have a large dimensions. So reducing these dimensions needs to happen for the algorithm to be useful It takes care of multicollinearity by removing redundant features. For example, you have two variables – 'time spent on treadmill in minutes' and 'calories burnt'. These variables are highly correlated as the more time you spend running on a treadmill, the more calories you will burn. Hence, there is no point in storing both as just one of them does what you require It helps in visualizing data. As discussed earlier, it is very difficult to visualize data in higher dimensions so reducing our space to 2D or 3D may allow us to plot and observe patterns more clearly

## 1.3 Sparse Approximations / Representations

## 1.4 Interpretierbarkeit

## 1.5 Compressed Sensing Beispiel

# Kapitel 2

# Mathematische Grundlagen

## 2.1 Normen und deren Effekte

### 2.1.1 l0-Norm

### 2.1.2 l1-Norm

### 2.1.3 l2-Norm

## 2.2 Regression

Lineare Regression (Least Squares)

### 2.2.1 LASSO

### 2.2.2 Ridge Regression

## 2.3 Orthogonalprojektion

**Definition 2.1.** Zwei Vektoren $\vec{a}$ und $\vec{b}$ sind genau dann orthogonal, wenn ihr Skalarprodukt null ist, also

$$\vec{a} \perp \vec{b} \iff \vec{a} \cdot \vec{b} = 0.$$

Was sind orthogonale, orthonormale Matrizen, orthogonale, orthonormale Basis? Skalarprodukt? Von einem Skalarprodukt induzierte Norm? Projektionsmatrizen?

Allgemeine orthogonale Projektionsmatrix falls keine ONB gegeben ist.

$$\mathbf{P_A} = \mathbf{A}(\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}$$

Von Wikipedia:

**Definition 2.2.** Eine Orthogonalprojektion auf einen Untervektorraum $U$ eines Vektorraums $V$ ist eine lineare Abbildung $P_U \colon V \to V$, die für alle Vektoren $v \in V$ die beiden Eigenschaften

- $P_U(v) \in U$ (Projektion)
- $\langle P_U(v) - v, u \rangle = 0$ für alle $u \in U$ (Orthogonalität)

erfüllt.

Allgemeine orthogonale Projektion auf einen affinen linearen Unterraum.

$$P_{U_0}(v) = r_0 + \sum_{i=1}^{k} \frac{\langle v - r_0, w_i \rangle}{\langle w_i, w_i \rangle} w_i$$

WÖRTLICH VON WIKIPEDIA: Der orthogonal projizierte Vektor minimiert den Abstand zwischen dem Ausgangsvektor und allen Vektoren des Untervektorraums bezüglich der von dem Skalarprodukt abgeleiteten Norm $\|\cdot\|$, denn es gilt mit dem Satz des Pythagoras für Skalarproduktträume

$$\|u - v\|^2 = \|u - P_U(v)\|^2 + \|P_U(v) - v\|^2 \geq \|P_U(v) - v\|^2$$

## 2.4 Matrixzerlegungen

Diagonalisierbarkeit?

### 2.4.1 Eigenwertzerlegung

**Eigenwerte, Eigenvektoren**

### 2.4.2 Singulärwertzerlegung

**Singulärwerte**

## 2.5 Signaltheorie

### 2.5.1 Fouriertransformation

### 2.5.2 Nyquist-Shannon Abtasttheorem

## 2.6 Statistik

Varianz, Erwartungswert

### 2.6.1 Empirische Kovarianzmatrix

## 2.7 Mannigfaltigkeit

## 2.8 Dictionary Learning

# Kapitel 3

# Hauptkomponentenanalyse

Die Hauptkomponentenanalyse ist ein weitverbreitetes multivariates statistisches Verfahren zur Dimensionsreduktion. Multivariate Verfahren zielen darauf ab, die in einem Datensatz enthaltene Zahl der Variablen zu verringern, ohne die darin enthaltene Information (zu verlieren) / (wesentlich zu reduzieren). Dadurch können umfangreiche Datensätze strukturiert, veranschaulicht und vereinfacht werden. Somit ist das Verfahren Teil der explorativen Statistik, welche Datensätze hinsichtlich ihrer Zusammenhänge analysiert. Die sich ergebende Struktur kann für weitere Analysezwecke ausgenutzt werden.

So hat die Hauptkomponentenanalyse in vielen Bereichen erfolgreich Anwendung gefunden. Darunter fällt die Erkennung handgeschriebener Zahlen, welche zum Beispiel zur automatischen Sortierung von Briefen nach Postleitzahl genutzt wird [13]. Hier lässt sich die Bedeutung der Findung von Clustern besonders gut verdeutlichen. Man erhofft, dass nach Anwendung einer Dimensionsreduktion wie PCA 10 verschiedene Gruppierungen zu erkennen sind, die für die Ziffern 0 bis 9 stehen. Optimalerweise gehören alle Datenpunkte im demselbem Cluster zur selben Ziffer. Außerdem korrespondieren nahe beieinanderliegende Cluster mit Ziffern, die ähnlich aussehen. Weitere Anwendungen findet das Verfahren in der Bildverarbeitung. Hier kann es zum Beispiel zur Rauschunterdrückung [2] oder zur Gesichtserkennung [1] genutzt werden. Hierbei werden einzelne Pixel oder patches, also lokale Gruppierungen von Pixeln, eines Bildes als Variable interpretiert.

Das dahinterstehende mathematische Problem kann auf verschiedene Weisen beschrieben werden. Zunächst wollen wir die Hauptkomponentenanalyse so konstruieren, dass die Idee des minimalen Informationsverlust im Vordergrund steht. Anschließend werden wir das Problem auf eine Singulärwertzerlegung zurückführen, die auch zur effizienten Implementierung genutzt wird. Des Weiteren werden wir die Hauptkomponentenanalyse als Regressionsproblem betrachten und die geometrische Interpretation weiter verdeutlichen. Zu Schluss werden wir einige theoretische Aussagen zeigen.

## 3.1 Konstruktion

Gegeben sei ein Datensatz mit $n$ samples und $p$ Variablen. Die zentrale Idee der Hauptkomponentenanalyse besteht darin, die $p$ bestehenden Variablen in $r$ neue, unkorrelierte Variablen zu überführen. Um eine Reduktion der Dimension, also $r < p$ zu erreichen, müssen die bestehenden Variablen *zusammengefasst* werden. Idealerweise sollte bei diesem Prozess möglichst wenig Information verloren gehen. Als Maß für den Informationsgehalt der Daten wird hierbei die Varianz verwendet. Das

heißt, je größer die Varianz einer Variable, desto mehr Information birgt sie und desto *wichtiger* ist sie. Hätte man eine Variable, die für alle Beobachtungen ähnliche Werte hat, so ist Diese nicht von Nutzen bei der Unterscheidung verschiedener samples. PCA sucht also nach Eigenschaften, die viel Varianz zeigen. Dabei wählt PCA aber nicht einfach nur bestimmte Eigenschaften mit viel Varianz aus, sondern konstruiert neue Variablen, die die bestehenden zusammenfassen.

Konkret suchen wir also sukzessive nach einer Linearkombination der bestehenden Variablen. Die entstehenden Vektor zeigen dann in die Richtung größter Varianz in unserem Datensatz. Wir nennen sie die Hauptachsen bzw. Hauptrichtungen unseres Datensatzes. Nachdem wir Diese berechnet haben wollen wir unsere Beobachtungen bezüglich der neuen Variablen darstellen. Dazu verwenden wir die orthogonale Projektion. Dies entspricht einer Rotation des Koordinatensystems, so dass die Hauptrichtugen den Standardachsen entsprechen. Die so entstehenden Werte bezüglich der neuen Variablen werden Hauptkomponenten genannt.

Abbildung Höhe Gewicht mit Eigenvektoren und gedrehtes Bild

Um dieses Prinzip zu veranschaulichen, wenden wir uns nun einem simplem Beispiel zu. Gegeben seien die Größe [cm] und das Gewicht [kg] zu 1000 Personen (Daten sind simuliert, keine real-world-data) (siehe dazu Abbildung). In diesem Fall ist also $n = 1000$ und $p = 2$. Bei Betrachtung der Abbildung fällt schnell auf, dass die beiden Variablen positiv korreliert sind, d.h. prinzipiell gibt es die folgende Tendenz: Je größer eine Person, desto schwerer ist sie. Wir können

Konkret konstruieren wir Variablen, die sich aus Linerakombinationen der Alten zusammensetzen. Dabei sollen die neuen Variablen der Wichtigkeit nach sortiert sein. In anderen Worten enthält die erste Variable die meiste Information bzw. die größte Varianz, dann die zweite, usw.

Abbildung Scree Plot

Die eigentliche Dimensionsreduktion findet dann durch Selektierung statt. Je nach Komplexität des Modells und Informationsverlust können so mehr oder weniger ausgewählt werden. Somit haben wir eine kleine, neue Menge an Variablen, die aber trotzdem den Großteil an Informationen / Varianz beinhaltet. Anordnung nach absteigender Varianz bzw. Information.

Bevor wir die Hauptkomponentenanalyse auf den Datensatz anwenden können gibt es noch einen wichtigen Bearbeitungsschritt zu beachten. Wenn eine Variable weniger variiert als eine Andere aufgrund der verwendeten Einheit oder Skala (meter oder kilo) kann dies zu ungewollten Ergebnissen führen. Ohne eine Vorbehandlung der Daten hat so im obigen Beispiel eine Änderung von 1m die gleiche Bedeutung wie eine Änderung von 1kg. Allerdings ist ein Mensch, der 1m größer ist, ein ganz Anderer während ein Mensch, der 1kg mehr wiegt, noch sehr ähnlich ist. Daher werden die Daten häufig einem sog. preprocessing unterzogen. Ein zu diesem Zweck oft verwendetes Verfahren ist die Standardisierung (auch z-Transformation genannt). In diesem Schritt werden die Variablen so transfomiert, dass sie "vergleichbarer"werden. Seien dazu $X_i$ die Zufallsvariablen mit Erwartungswert $E[X_i] = \mu$ und Varianz $Var[X_i] = \sigma^2$. So erhält man die zugehörigen standardisierten Zufallsvariablen $Z_i$ durch Zentrierung und anschließender Division durch die Standardabweichung $Z = \frac{X-\mu}{\sigma}$. Somit gilt:

- $E[Z_i] = 0$ für alle $1 \leq i \leq p$

- $Var[Z_i] = 1$ für alle $1 \leq i \leq p$

Mathematisch gesehen endet man das Verfahren also nicht auf die Kovarianzmatrix, sondern auf die Korrelationsmatrix an.

### 3.1.1 Problemformulierung als Varianzmaximierung

Wir wollen nun die Intuition des minimalen Informationsverlust mathematisch formulieren. Gegeben sei dazu eine Matrix $\mathbf{X} \in \mathbb{R}^{n \times p}$, wobei $n$ die Anzahl der Samples bzw. Beobachtungen und $p$ die Anzahl der Variablen ist. Wir nehmen im Folgenden ohne Beschränkung der Allgemeinheit an, dass die Variablen zuvor zentriert wurden. Aufgabe der Hauptkomponentenanalyse ist es nun sukzessive Richtungen größter Varianz zu finden. Die erste Hauptkomponente ist definiert durch $Z_1 = \sum_{j=1}^{p} v_{1j} X_j = \mathbf{X}v$ wobei die Hauptachse $v_1 = (v_{11}, \ldots, v_{1p})^T$ so gewählt wird, dass die Varianz von $Z_1$ maximiert wird, d.h.

$$v_1 = \underset{\|v\|_2=1}{\arg\max} \operatorname{Var}[\mathbf{X}v] = \underset{\|v\|_2=1}{\arg\max}\, v^T \mathbf{K}_{xx} v$$

mit $\mathbf{K_{xx}} = \frac{\mathbf{X}^T\mathbf{X}}{n-1}$ als Stichprobenkovarianzmatrix. Die restlichen Hauptachsen können nun sukzessive definiert werden.

$$v_{k+1} = \underset{\|v\|=1}{\arg\max}\, v^T \mathbf{K}_{xx} v$$

$$v_{k+1}^T v_l = 0 \quad \forall 1 \leq l \leq k$$

Man sucht also unter den Richtungen, die orthogonal zu allen bisherigen Hauptachsen sind, diejenige, die die Varianz maximiert. Wie oben erhält man dann die Hauptkomponenten, also die Darstellung der Daten bezüglich der neu gefundenen Variablen, durch $Z_i = \mathbf{X}v_i$. [18] CITE JOLLIFE

Wie wir bereits in THEOREM gesehen haben, entsprechen die Eigenvektoren der Kovarianzmatrix genau den Richtungen maximaler Varianz wie oben definiert. Daher können wir anstatt sukzessiver Berechnung einzelner Hauptachsen die Kovarianzmatrix $\mathbf{K}_{xx}$ diagonalisieren. Da $\mathbf{K}_{xx}$ symmetrisch ist

$$\mathbf{K}_{xx} = \mathbf{V}\mathbf{L}\mathbf{V}^T$$

wobei $\mathbf{V}$ die Matrix der Eigenvektoren ist (Jede Spalte ist ein Eigenvektor) und $\mathbf{L}$ ist eine Diagonalmatrix mit Eigenwerten $\lambda_i$ in absteigender Reihenfolge. Die Eigenvektoren entsprechen den Hauptachsen und die Projektion der Daten auf die Hauptachsen wird erreicht durch Multiplikation des Datensatz mit den Eigenvektoren $\mathbf{Z} = \mathbf{X}\mathbf{V}$. Die Spalten in $\mathbf{Z}$ sind also die Hauptkomponenten. Die i-te Beobachtung bezüglich der neuen Variablen sind die Zeilen von $\mathbf{X}\mathbf{V}$.

Wie wir bereits in CITE gesehen haben, entsprechend diese Richtungen genau den Eigenvektoren der Stichprobenkovarianzmatrix. Die Stichprobenkovarianzmatrix ist gegeben durch $\mathbf{K}_{xx} = \frac{(\mathbf{X}^T\mathbf{X})}{n}$.

### 3.1.2 Formulierung als Singulärwertzerlegung

$$\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^T$$

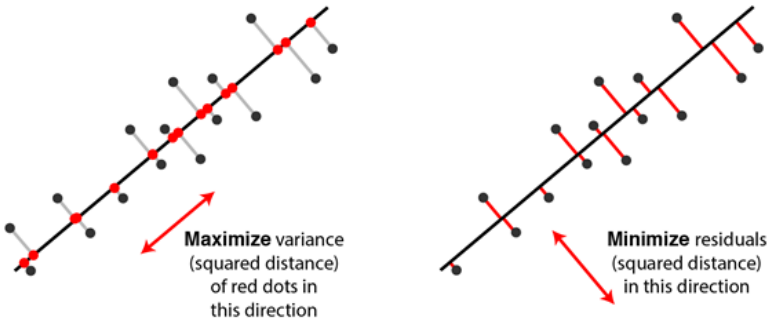


ABBILDUNG 3.1: Die obenstehende Abbildung zeigt die Äquivalenz von Maximierung der Varianz und Minimierung der Projektionsdistanz

wobei **D** eine Diagonalmatrix mit Elementen $d_1, \ldots, d_p$ in absteigender Reihenfolge, **U** eine $n \times p$ und **V** eine $p \times p$ orthogonale Matrix. **UV** sind die Hauptkomponenten und die Spalten von **V** sind die Eigenvektoren von **X**.

### 3.1.3 Formulierung als Regressionsproblem

$$\hat{\mathbf{V}}_k = \arg\min_{\mathbf{V}_k} \sum_{i=1}^{n} \left\| x_i - \mathbf{V}_k \mathbf{V}_k^T x_i \right\|^2 + \lambda \sum_{j=1}^{k} \|\beta_j\|^2$$

subject to $\mathbf{V}_k^T \mathbf{V}_k = I_{k \times k}$

Wie bereits erwähnt (Wurde es erwähnt?) ist PCA ein lineares Dimensionsreduktionsverfahren, d.h. dass die Daten in den niedrigdimensionalen Raum linear, also durch eine Kombination von Rotationen und Translationen, transformiert / projeziert werden.

[17]

Man projiziert die Daten auf einen k-dimensionalen linearen Unterraum. Man kann zeigen, dass die Lösung dieses Problem genau die ersten k Hauptachsen sind.

Ausgehend von dieser Formulierung als Regressionsproblem werden wir im nächsten Kapitel die Variante der dünnbesetzten Hauptkomponentenanalyse beschreiben.

## 3.2 Dimensionsreduktion

Wie viele Hauptkomponenten sollen wir auswählen?

Optimal singular threshold [4]

## 3.3 Grenzen der Anwendbarkeit

Obwohl die Hauptkomponentenanalysen in vielen Situationen helfen kann, Datensätze zu veranschaulichen und zu strukturieren, gibt es keine Garantie für sinnvolle Ergebnisse. Im Folgendem werden wir Szenarien beschreiben, bei denen unerwünschte Effekte bei der Verwendung dieses Verfahrens auftreten. Daher gilt es den Datensatz vorest hinsichtlich folgender Gesichtspunkte zu untersuchen:

- Lineare Beziehung zwischen Variablen
- Korrelation der Variablen
- Vollständigkeit des Datensatzes
- Ausreißer in den Daten
- Anzahl an Beobachtungen in Relation zu Anzahl an Variablen

Wie in REF beschrieben versuchen wir Daten in einen niedrigdimensionaleren linearen oder affinen Unterraum zu transformieren. Es kann aber durchaus vorkommen, dass es keine lineare Beziehung zwischen den Variablen gibt. Nichtlineare Strukturen können von PCA nicht erfasst werden und gehen somit verloren. [14] Vidal et al. zeigen diese Grenze konkret am Beispiel von Porträt-Fotos auf. Seit der Entstehung von PCA gab es aber zahlreiche nicht-lineare Erweiterungen. So nutzt zum Beispiel Kernel PCA den *Kernel Trick* aus, bei welchem man die Daten zuerst durch eine nichtlineare Transformation in ein höherdimensionalen Raum einbettet von dem man sich erhofft, dass die Daten in diesem linear verteilt. Erst anschließend wird dann die eigentliche Reduktion durchgeführt. Hierbei muss man die Daten aber nicht im höherdimensionalen Raum auswerten. CITE. Andere Erweiterungen, die allgemein unter *manifold learning* zusammengefasst werden können, basieren auf der Idee, dass die Dimension des Datensatz nur künstlich hoch ist. Man versucht die lokale Geometrie der Mannigfaltigkeit (Begriff erklären?) zu approximieren und damit direkt eine niedrigdimensionale Einbettung zu erhalten. Hierunter fallen zum Beispiel die multidimensionale Skalierung oder ISOMAP.

Damit der Datensatz für eine Dimensionsreduktion per PCA geeignet ist, müssen die verschiedenen Variablen einen gewissen Grad an Korrelation aufweisen. Im extremen Fall der Unabhängigkeit der Variablen bewirkt eine Hauptachsentransformation nichts. Reduziert man dann die Anzahl der Hauptkomponenten verliert man mit jeder Variable einen Großteil der Information.

Ein weiterer Gesichtspunkt ist die Vollständigkeit eines Datensatzes. Finden wir fehlende oder korrupte Einträge in unserem Datensatz vor, kann die klassische Hauptkomponentenanalyse ... . Für dieser Art Probleme existieren entsprechende Ergänzungen von PCA wie zum Beispiel in cite und cite. Ausreißer in den Daten können die Resultate drastisch beeinflussen. Genaue Effekte überlegen und CITE. Aus diesem Grund sollten Ausreißer vor der Anwendung von PCA entfernt werden.

Außreiser in den Daten.

Anzahl der Variablen zu hoch.

Darüber hinaus gibt es noch eine Reihe Spezialfälle, bei denen Probleme auftreten können. So kann es zum Beispiel passieren, dass die relevanten Informationen in den Variablen mit niedriger Varianz versteckt sind. Da die Hauptkomponentenanalyse gerade diese Variablen vernachlässigt, wird sich unter Umständen nicht die

erwünschte Struktur auf den Daten ergeben. Es bedarf anderer Methoden mit anderen Ansätzen, um eine Dimensionsreduktion zu ermöglichen. Oftmals weiß man aber im Vorhinein nicht, in welchen Variablen diese Unterscheidungsmöglichkeit versteckt ist.

Das wohl wichtigste/größte Hindernis im Zuge dieser Arbeit ist sicherlich die durch die Transformation entstehenden Interpretationsschwierigkeiten. Jede Hauptkomponente entsteht wie oben beschrieben durch eine Linearkombination der Ausgangsvariablen. Während die Ausgangsvariablen Bedeutungen wie Gewicht oder Größe hatten ist in vor allem in hochdimensionalen Fällen eine Interpretation der Hauptkomponenten nur schwer möglich (Rotation Techniques CITE). Dieser Interpretationsverlust ist Ausgangspunkt der Idee der dünnbesetzten Hauptkomponentenanalyse, genannt sparse PCA. Diesem Verfahren ist das folgende Kapitel gewidmet.

## 3.4 Theoretische Aussagen

**Theorem 3.1.** *PCA always gives unique solution.*

**Theorem 3.2** ([14])**.** *Sei* $\mathbf{X} \in \mathbb{R}^n$ *und* $\mathbf{A}_{p,k} = [\alpha_1, \dots \alpha_k]$

**Theorem 3.3.** *PCA inconsistent for* $p \gg n$.

# Kapitel 4

# Dünnbesetzte Hauptkomponentenanalyse

Ein Nachteil der Hauptkomponentenanalyse ist, dass sich die neuen Variablen meist aus einer Linearkombination aller bestehenden Variablen zusammensetzt. Dies macht es besonders für hochdimensionale Daten schwierig die Hauptachsen zu interpretieren. Oft können somit nicht die relevanten features/Variablen herausgelesen werden. Es kann durchaus passieren, dass nicht alle Variablen relevant zur Strukturerkennung sind.

## 4.1 Motivation

## 4.2 Problemformulierung

NP-schwere Formulierung

## 4.3 Relaxation / Approximation Ideen

## 4.4 Konstruktion

Sparse PCA Kriterium.

$$(\hat{\mathbf{A}}\hat{\mathbf{B}}) = \underset{\mathbf{A},\mathbf{B}}{\arg\min} \sum_{i=1}^{n} \left\| x_i - \mathbf{A}\mathbf{B}^T x_i \right\|^2 + \lambda \sum_{j=1}^{k} \left\| \beta_j \right\|^2 + \sum_{j=1}^{k} \lambda_{1,j} \left\| \beta_j \right\|_1$$

subject to $\mathbf{A}^T\mathbf{A} = I_{k\times k}$

## 4.5 Theoretische Aussagen Sparse PCA

z.B. wie werden neue Varianzen berechnet

# Kapitel 5

# Implementierung

## 5.1 Implementierung nach original paper

Eigene Implementierung

## 5.2 Implementierung in scikit-learn in python

## 5.3 Laufzeitvergleich

# Kapitel 6

# Anwendung

## 6.1 Anwendung auf Simulationsdaten

Vergleich Tabelle PCA / Sparse PCA (Loadings)

## 6.2 Der Datensatz

## 6.3 Anwedung auf Frequenzdaten

## 6.4 Auswertung der Ergebnisse

## 6.5 Vergleich mit PCA Resultaten

## 6.6 Hyperparameter

Veränderung des Hyperparameters und dessen Effekte

### 6.6.1 Zeit

### 6.6.2 Effekt auf Resultate

# Kapitel 7

# Ausblick / Zusammenfassung

## 7.1 Einsetzbarkeit

Wann ist die Methode sinnvoll einzusetzen?

## 7.2 Übertragbarkeit

Übertragbarkeit auf andere Datensätze

## 7.3 Ongoing Research / Weitere Techniken

# Literatur


[1] In: ().

[2] Y. Murali Mohan Babu, Dr. M. V. Subramanyam und Dr. M. N. Giri Prasad. „PCA based image denoising“. In: 2012.

[3] Michael Elad. *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*. English. Bd. 1. Springer Science+Business Media, 2010, S. 376.

[4] gavish. „The Optimal Hard Threshold for Singular Values is $4/\sqrt{3}$“. In: *IEEE Transactions on Information Theory* 60.8 (2014), S. 5040–5053. DOI: 10.1109/TIT.2014.2323359.

[5] Rémi Gribonval, Rodolphe Jenatton und Francis R. Bach. „Sparse and spurious: dictionary learning with noise and outliers“. In: *CoRR* abs/1407.5155 (2014). arXiv: 1407.5155. URL: http://arxiv.org/abs/1407.5155.

[6] Iain M. Johnstone und Arthur Yu Lu. „On Consistency and Sparsity for Principal Components Analysis in High Dimensions“. In: *Journal of the American Statistical Association* 104.486 (2009). PMID: 20617121, S. 682–693. DOI: 10.1198/jasa.2009.0121. eprint: https://doi.org/10.1198/jasa.2009.0121. URL: https://doi.org/10.1198/jasa.2009.0121.

[7] Jean Ponce Guillermo Sapiro Julien Mairal Francis Bach. „Online Dictionary Learning for Sparse Coding“. In: *Proceedings of the 26th Annual International Conference on Machine Learning*. ICML '09. ACM, 2009, S. 689–696. DOI: 10.1145/1553374.1553463. URL: http://doi.acm.org/10.1145/1553374.1553463.

[8] Francis R. Bach Rodolphe Jenatton Guillaume Obozinski. „Structured Sparse Principal Component Analysis“. In: *Artificial Intelligence and Statistics (AISTATS)* 9 (2010). URL: https://arxiv.org/abs/0909.1440.

[9] Holger Rauhut Simon Foucart. *A Mathematical Introduction to Compressive Sensing*. English. Bd. 1. Birkhäuser Basel, 2013, S. 625.

[10] Robert Tibshirani. „Regression Shrinkage and Selection via the Lasso“. In: *Journal of the Royal Statistical Society. Series B (Methodological)* 58.1 (1996), S. 267–288. URL: http://www.jstor.org/stable/2346178.

[11] Robert Tibshirani u. a. „Least angle regression“. In: *The Annals of Statistics* 32.2 (2004), 407–499. DOI: 10.1214/009053604000000067. URL: http://dx.doi.org/10.1214/009053604000000067.

[12] Ryan J. Tibshirani. „The Lasso Problem and Uniqueness“. In: (2012). URL: https://arxiv.org/abs/1206.0313.

[13] Jerome Friedman Trevor Hastie Robert Tibshirani. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. English. Bd. 2. Springer-Verlag New York, 2009, S. 745.

[14] R. Vidal, Y. Ma und S. Sastry. *Generalized Principal Component Analysis*. Bd. 1. Interdisciplinary Applied Mathematics. Springer New York, 2016. ISBN: 9780387878119. DOI: 10.1007/978-0-387-87811-9. URL: https://books.google.de/books?id=I9H7CwAAQBAJ.

[15] Kazuyoshi Yata und Makoto Aoshima. „Effective PCA for high-dimension, low-sample-size data with noise reduction via geometric representations“. In: *Journal of Multivariate Analysis* 105.1 (2012), S. 193–215. URL: https://doi.org/10.1016/j.jmva.2011.09.002.

[16] Hui Zou und Trevor Hastie. „Regularization and Variable Selection via the Elastic Net“. In: *Journal of the Royal Statistical Society. Series B (Statistical Methodology)* 67.2 (2005), S. 301–320. URL: http://www.jstor.org/stable/3647580.

[17] Hui Zou, Trevor Hastie und Robert Tibshirani. „Sparse Principal Component Analysis“. In: *Journal of Computational and Graphical Statistics* 15.2 (2006), S. 265–286. DOI: 10.1198/106186006X113430. URL: https://doi.org/10.1198/106186006X113430.

[18] Hui Zou und Lingzhou Xue. „A Selective Overview of Sparse Principal Component Analysis“. In: *Proceedings of the IEEE* 106.8 (2018), S. 1311–1320. DOI: 10.1109/JPROC.2018.2846588. URL: https://ieeexplore.ieee.org/document/8412518.