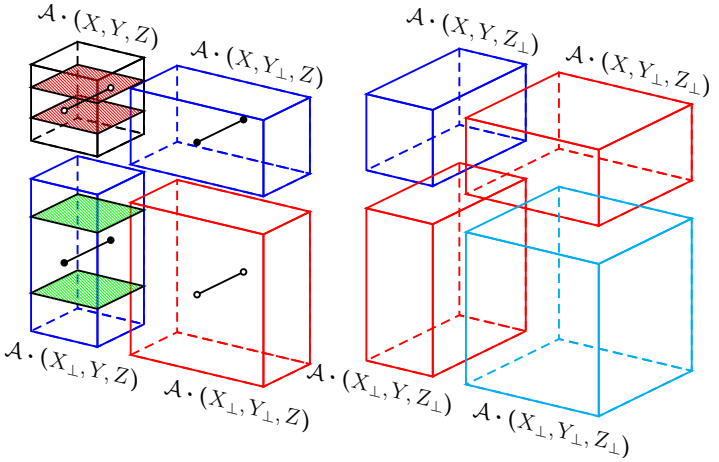
Linköping Studies in Science and Technology
Dissertations, No 1178

Algorithms in Data Mining using Matrix and Tensor Methods

Berkant Savas



Linköping University
INSTITUTE OF TECHNOLOGY

Department of Mathematics
Scientific Computing
Linköping 2008

Linköping Studies in Science and Technology
Dissertations, No 1178

Algorithms in Data Mining using Matrix and Tensor Methods

Scientific Computing
Department of Mathematics
Linköping University
SE–581 83 Linköping, Sweden

besav@math.liu.se
www.mai.liu.se/Num/

ISBN 978-91-7393-907-2
ISSN 0345-7524

The thesis is available for download at Linköping University Electronic Press:
http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-11597

Printed by LiU-Tryck, Linköping 2008, Sweden

To Alice, Jessica and my family

Abstract

In many fields of science, engineering, and economics large amounts of data are stored and there is a need to analyze these data in order to extract information for various purposes. Data mining is a general concept involving different tools for performing this kind of analysis. The development of mathematical models and efficient algorithms is of key importance. In this thesis we discuss algorithms for the reduced rank regression problem and algorithms for the computation of the best multilinear rank approximation of tensors.

The first two papers deal with the reduced rank regression problem, which is encountered in the field of state-space subspace system identification. More specifically the problem is

$$\min_{\mathrm{rank}(X)=k} \det(B - XA)(B - XA)^{\mathsf{T}},$$

where A and B are given matrices and we want to find X under a certain rank condition that minimizes the determinant. This problem is not properly stated since it involves implicit assumptions on A and B so that $(B-XA)(B-XA)^{\mathsf{T}}$ is never singular. This deficiency of the determinant minimization is fixed by generalizing the criterion to rank reduction and volume minimization of the objective matrix. The volume of a matrix is defined as the product of its nonzero singular values. We give an algorithm that solves the generalized problem and identify properties of the input and output signals causing a singular objective matrix.

Classification problems occur in many applications. The task is to determine the label or class of an unknown object. The third paper concerns with classification of handwritten digits in the context of tensors or multidimensional data arrays. Tensor and multilinear algebra is an area that attracts more and more attention because of the multidimensional structure of the collected data in various applications. Two classification algorithms are given based on the higher order singular value decomposition (HOSVD). The main algorithm makes a data reduction using HOSVD of 98–99 % prior the construction of the class models. The models are computed as a set of orthonormal bases spanning the dominant subspaces for the different classes. An unknown digit is expressed as a linear combination of the basis vectors. The resulting algorithm achieves 5% in classification error with fairly low amount of computations.

The remaining two papers discuss computational methods for the best multilinear rank approximation problem

$$\min_{\mathcal{B}} \|\mathcal{A} - \mathcal{B}\|,$$

where $\mathcal{A}$ is a given tensor and we seek the best low multilinear rank approximation tensor $\mathcal{B}$. This is a generalization of the best low rank matrix approximation problem. It is well known that for matrices the solution is given by truncating the singular values in the singular value decomposition (SVD) of the matrix. But for tensors in general the truncated HOSVD does not give an optimal approximation. A third order tensor $\mathcal{B} \in \mathbb{R}^{I\times J\times K}$ with $\text{rank}(\mathcal{B}) = (r_1, r_2, r_3)$ can be written as the product

$$\mathcal{B} = (X, Y, Z) \cdot \mathcal{C}, \qquad b_{ijk} = \sum_{\lambda,\mu,\nu} x_{i\lambda} y_{j\mu} z_{k\nu} c_{\lambda\mu\nu},$$

where $\mathcal{C} \in \mathbb{R}^{r_1\times r_2\times r_3}$ and $X \in \mathbb{R}^{I\times r_1}$, $Y \in \mathbb{R}^{J\times r_2}$, and $Z \in \mathbb{R}^{K\times r_3}$ are matrices with orthonormal columns. The approximation problem is equivalent to a nonlinear optimization problem defined on a product of Grassmann manifolds. We introduce novel techniques for multilinear algebraic manipulations enabling theoretical analysis and algorithmic implementation. These techniques are used to solve the approximation problem using Newton and quasi-Newton methods specifically adapted to operate on products of Grassmann manifolds. The presented algorithms are suited for small, large and sparse problems and, when applied to difficult problems, they clearly outperform alternating least squares methods, which are standard in the field.

Populärvetenskaplig sammanfattning

Inom många vetenskapliga, tekniska, ekonomiska och internetbaserade områden samlas och lagras stora mängder data som innehåller värdefull information. Denna information är inte direkt tillgänglig utan man behöver analysera data för att komma åt informationen. Datautvinning (data mining) är ett generellt begrepp som innehåller olika verktyg för att extrahera data och information. Särskilt viktigt är utvecklingen av matematiska modeller, algoritmer samt effektiv implementering av dessa på olika datorsystem. I denna avhandling utvecklar vi teori och algoritmer för datautvinning genom matris- och tensorberäkningar. Teorin och algoritmerna verifieras genom att tillämpas inom systemidentifiering och klassificering av handskrivna siffror.

Målet med systemidentifiering är att bestämma matematiska modeller av dynamiska system, som ett flygplan eller en bilmotor, baserat på uppmätta in- och utsignaler. Den beräknade modellen relaterar insignaler till utsignaler. Modellen används sedan i syfte att reglera och simulera systemet. I processen ingår ett minimeringsproblem som inkluderar uppmätta data och de okända parametrarna för modellen. För att problemet ska gå att lösa förutsätts att signalerna har vissa egenskaper. Vi har generaliserat minimeringsproblemet så att man kan beräkna en lösning även i de fall då de uppmätta signalerna inte har de föreskrivna egenskaperna.

Många algoritmer inom datautvinning är formulerade i termer av vektorer och matriser. Men i en hel del tillämpningar har data multidimensionell struktur, till skillnad från vektorer som är endimensionella och matriser som är tvådimensionella. Data med multidimensionell struktur finns bland annat inom ansiktsigenkänning, textanalys från internet, analys av biologiska och medicinska data. Exempel på detta är bildsekvenser, där varje bild ses som en matris och med många bilder får vi ett tredimensionellt dataobjekt eller en 3-tensor. I avhandlingen presenterar vi metoder för klassificering av handskrivna siffror, som bygger på data med multidimensionell struktur. Vi utvecklar också tensoralgoritmer och generaliserar begrepp och teori från linjär till multilinjär algebra. Vi studerar metoder för approximering av en given tensor med en annan tensor av lägre rang. Algoritmerna är baserade på Newton och quasi-Newton metoder för att optimera en objektfunktion som är definierad på en krökt multidimensionell yta.

Acknowledgements

First of all, I would like to express my deepest gratitude to Professor Lars Eldén for giving me the opportunity to conduct research studies in scientific computing, data mining and specifically matrix and tensor computations. During this work we have had many interesting, long-lasting and insightful discussions regarding tensors, notations, manifolds, algorithms, optimization and other encountered challenges, and I have enjoyed all of it. I am very grateful for his excellent guidance, careful paper reading, and encouragement.

I am grateful to the late Gene Golub for inviting me to Stanford University in the autumn of 2006. I really enjoyed this visit, the friendly and inspiring atmosphere and the interesting discussions both on and off campus with all these different people I met. Specifically, I would like to thank Lek-Heng Lim for a great collaboration and the work on quasi-Newton methods on Grassmannians. In addition I would like to thank Michael Saunders for lending me his bike during my visit and Morten Mørup for the housing during the days after my arrival.

The travel grant issued by Linköping University to the memory of Erik Månsson and the memory of Linnea and Henning Karlsson is highly acknowledged.

I would also like to thank David Lindgren and Lennart Ljung for interesting discussions and David for a very pleasant collaboration on the reduced rank regression problem in system identification.

Furthermore, I would like to thank Åke Björck, Oleg Burdakov and Lennart Simonsson, colleagues at the department of mathematics, for various and insightful discussions related to algorithms, manifolds, optimization and research in general.

Thank you Fredrik Berntsson and Gustaf Hendeby for useful advice on LaTeX.

I would also like to thank my colleges in the scientific computing group, the graduate students at MAI, and the department of mathematics for providing a good working environment.

Finally I want to express my gratitude for the grant from the Swedish Research Council making this work possible.

Papers

The following manuscripts are appended and will be referred to by their Roman numerals. The manuscripts I, II and III are published in different international journals. Manuscript IV is in the review process with SIAM Journal on Matrix Analysis and Applications. Currently manuscript V is a technical report.

[I] Berkant Savas, Dimensionality reduction and volume minimization – generalization of the determinant minimization criterion for reduced rank regression problems. *Linear Algebra and its Applications*, Volume 418, Issue 1, 2006, Pages 201–214. DOI:10.1016/j.laa.2006.01.032. Also Technical Report LiTH-MAT-R–2005-11–SE, Linköping University.

[II] Berkant Savas and David Lindgren, Rank reduction and volume minimization approach to state-space subspace system identification. *Signal processing*, Volume 86, Issue 11, 2006, Pages 3275–3285. DOI:10.1016/j.sigpro.2006.01.008. Also Technical Report LiTH-MAT-R–2005-13–SE, Linköping University.

[III] Berkant Savas and Lars Eldén, Handwritten digit classification using higher order singular value decomposition. *Pattern Recognition*, Volume 40, Issue 3, 2007, Pages 993–1003. DOI:10.1016/j.patcog.2006.08.004. Also Technical Report LiTH-MAT-R–2005-14–SE, Linköping University.

[IV] Lars Eldén and Berkant Savas, A Newton-Grassmann method for computing the best multilinear rank-(r_1, r_2, r_3) approximation of a tensor. Technical Report LiTH-MAT-R–2007-6-SE, Linköping University. Manuscript submitted to *SIAM Journal on Matrix Analysis and Applications*.

[V] Berkant Savas and Lek-Heng Lim, Best multilinear rank approximation of tensors with quasi-Newton methods on Grassmannians. Technical Report LiTH-MAT-R–2008-01–SE, Linköping University.

Contents

Appended manuscripts

1

Introduction and overview

DATA MINING is the process of finding and extracting valuable *knowledge* or *information* from a given and often large set of data. This general definition matches the problem descriptions in many scientific areas with applications in physics, finance, biology, chemistry, psychology, computer science and engineering. Data mining is highly interdisciplinary since it involves problems from various fields, it involves mathematics and statistics in terms of modeling and analysis and it involves computer science in terms of algorithmic implementations. Specific examples are face recognition, fingerprint recognition, handwritten text recognition, speech recognition, text mining in document collections (Internet web sites and scientific papers in databases), classification and clustering of DNA sequences and proteins, source identification and separation in wireless communications. Detailed and comprehensive overviews of data mining algorithms and applications are given in [45, 36, 37, 38, 30, 28].

Advances in technology give major contributions to the field of data mining and scientific computing. The computational power and storage technology are becoming cheaper which enable extensive computations on vast amounts of data. At the same time, the acquired data from various applications today are already very large or growing rapidly. Two such examples are the Large Hadron Collider (LHC) at CERN that will generate data in the order petabyte (10^{15}) per year [79, 33], and the Google link-matrix, which serves as a model of the internet and has a dimension of the order billions [58, 20, 56]. The dimension of the matrix is determined from the number of internet sites. One of the important challenges in traditional and new fields of science and engineering is to develop mathematical theory, models and numerical methods enabling reliable and efficient algorithmic design to solve problems in data mining.

In many of these fields numerical linear algebra and matrix computations are extensively used for modeling and problem solving. In this thesis we have examined and give algorithms in two specific areas. We have considered the reduced rank regression problem and generalize the determinant minimization criteria to

rank reduction and volume minimization. We have also considered the problem of approximating a given tensor (multidimensional array of numbers) by another tensor of lower multilinear rank. We introduce an algebraic framework, that enables analysis and manipulations on tensor expressions, as well as efficient numerical algorithms for the tensor approximation problem.

1 Linear systems of equations and linear regression models

Linear systems of equations is one of the most common problems encountered in scientific computing. We want to solve

$$Ax = b, \tag{1.1}$$

where $A \in \mathbb{R}^{m\times n}$ and $b \in \mathbb{R}^m$ are given and $x \in \mathbb{R}^n$ is unknown. Of course, if the vector b is not in the range space of A the equation (1.1) will not have a solution. If this is the case, one computes a solution x such that Ax is, in some measure, the best approximation to b. Mathematically we write

$$\min_x \|Ax - b\|, \tag{1.2}$$

where the norm is often the Euclidean. This problem occurs naturally when one wants to fit a linear model to measured observations. Then the number of measurements is larger than the number of unknowns, i.e. $m > n$ and we have an overdetermined set of linear equations. The approach to minimize the residual $r = Ax - b$ is sound since it is interpreted as to minimize the influence of the errors in the measurements [11].

An alternative way to approach the problem is to relate the observation vector b to the unknown vector x using the linear statistical model

$$Ax = b + \epsilon, \tag{1.3}$$

where we introduce the vector ϵ containing random errors. It is assumed in the standard model that the entries ϵ_i are uncorrelated, have zero mean and the same variance, i.e.

$$\mathbf{E}(\epsilon) = 0, \qquad \mathbf{V}(\epsilon) = \sigma^2 I,$$

where $\mathbf{E}(\cdot)$ denotes the expected value and $\mathbf{V}(\cdot)$ denotes the variance.

A related model is linear reduced-rank regression,

$$b(t_i) = Xa(t_i) + e(t_i), \qquad i = 1, 2, \ldots, N \tag{1.4}$$

where the unknown regression matrix $X \in \mathbb{R}^{m\times p}$ is constrained to have

$$\operatorname{rank}(X) = k < \min(m, p).$$

The vectors $b(t_i) \in \mathbb{R}^m$ and $a(t_i) \in \mathbb{R}^p$ are measurements at different time steps t_i and $e(t_i)$ are the corresponding errors. It is assumed that $e(t_i)$ is temporally white noise, normally distributed with unknown covariance matrix $\mathbf{E}(e(t)e(t)^\mathsf{T})$. Gathering all the measurements we can write the regression model (1.4) as

$$B = XA + E, \qquad \operatorname{rank}(X) = k, \tag{1.5}$$

where $B = [b(t_1)\, b(t_2)\, \ldots\, b(t_N)]$, $A = [a(t_1)\, a(t_2)\, \ldots\, a(t_N)]$ and correspondingly for E. Without the rank constraint on X, it would be straightforward to transform (1.5) to the model (1.3) by means of vectorization. Having the rank constraint one can consider to minimize the difference in Frobenius norm, i.e.

$$\min_{\mathrm{rank}(X)=k} \|B - XA\|_F. \tag{1.6}$$

An alternative way is to estimate X by the maximum likelihood method [7, 80, 53, 6, 31]. We will not go in to the statistical background but rather analyze the matrix problem to which it reduces.

2 The determinant minimization criterion

The first paper [72, Paper I] in this thesis is concerned with the determinant minimization problem

$$\min_{\mathrm{rank}(X)=k} \det(B - XA)(B - XA)^{\mathsf{T}}, \tag{2.1}$$

which, under certain circumstances, gives the maximum likelihood estimate for the reduced-rank regression model (1.4), see [80] for details. In the derivation of this problem there are several assumptions on the data that are justified due to the existence of enough noise in the measurements. The effect of the assumptions are that the determinant in (2.1) can not be made equal to zero no matter how we choose X. Thus, the most simple case where $B = XA$, which trivially gives $\det(B - XA)(B - XA)^{\mathsf{T}} = 0$, can not be solved by this approach. In addition algorithms for the numerical computation of X under these conditions fail [31].

Our objective was to determine the different scenarios where the determinant minimization criterion fails, in the sense that it does not give a well defined solution, and generalize the minimization criterion in order to obtain a well defined solution in all cases.

To clarify the problem with the determinant criterion, we recall that the determinant of a matrix $F \in \mathbb{R}^{m\times m}$ can be written as

$$\det(F) = \prod_{i=1}^{m} \sigma_i,$$

where σ_i are the singular values of F. If now the matrix F depends on some variables X, in particular if we set

$$F(X) = (B - XA)(B - XA)^{\mathsf{T}},$$

then

$$\det\left(F(X)\right) = \prod_{i=1}^{m} \sigma_i(F(X)). \tag{2.2}$$

The singular values $\sigma_i(F(X))$ now depend on X and zeroing one singular value of $F(X)$ would zero the determinant. One can, in certain cases, zero the determinant by simply setting parts of the matrix X to zero. The rest of the matrix would be undetermined, nonetheless the determinant is minimized but with no solution of substance. Consider Example 1.1 in [72] (page 34) for this scenario.

3 Generalization to rank reduction and volume minimization

Zeroing one singular value is not sufficient for computing a solution. We propose an approach to continue and zero as many singular values as possible. In addition one should also minimize the product of the rest of the singular values that can not be zeroed. Mathematically we write the generalization as follows.

$$\min_{\substack{\operatorname{rank}(X)=k \\ \operatorname{rank}(F(X))=r_{\min}}} \operatorname{vol}(F(X)), \quad r_{\min} = \min_{\operatorname{rank}(X)=k} \operatorname{rank}(F(X)), \tag{3.1}$$

where the volume of a matrix is defined as the product of the nonzero singular values [10]. If $\operatorname{rank}(F) = r$ then

$$\operatorname{vol}(F) = \prod_{i=1}^{r} \sigma_i,$$

where $\sigma_1, \ldots, \sigma_r$ are the nonzero singular values.

The tools for analyzing, but also computing the solution to, the generalized minimization problem are the singular value decomposition (SVD).

Theorem 3.1 (SVD). *Any given matrix $A \in \mathbb{R}^{m\times n}$ can be factorized as*

$$A = U\Sigma V^{\mathsf{T}},$$

where $U \in \mathbb{R}^{m\times m}, V \in \mathbb{R}^{n\times n}$ are orthogonal matrices and $\Sigma \in \mathbb{R}^{m\times n}$ is a diagonal matrix with the nonnegative entries $\sigma_1 \geq \cdots \geq \sigma_{\min(m,n)}$. The matrices U and V are called the left and right singular matrices, respectively, and σ_i are the singular values.

Proof and algorithm of this decomposition can be found in [35].

In [72, Paper I] we prove that the generalized problem (3.1) with

$$F(X) = (B - XA)(B - XA)^{\mathsf{T}}$$

has the solution given by

$$X = U_B \begin{bmatrix} S_B U \Sigma_k V^{\mathsf{T}} S_A^{-1} & 0 \\ 0 & 0 \end{bmatrix} U_A^{\mathsf{T}}, \tag{3.2}$$

where

$$U_B \begin{bmatrix} S_B \\ 0 \end{bmatrix} P^{\mathsf{T}} = B, \quad U_A \begin{bmatrix} S_A \\ 0 \end{bmatrix} Q^{\mathsf{T}} = A, \quad U\Sigma V^{\mathsf{T}} = P^{\mathsf{T}} Q, \tag{3.3}$$

are the SVD's of the corresponding matrices and Σ_k contains the k largest singular values from Σ. Recall that B and A are from equation (1.5).

The solution can be interpreted as making the angles between the subspaces spanned by rows of B and XA as small as possible. We have also shown that the minimal volume $v_{\min} = \min \operatorname{vol}(F(X))$ is proportional to

$$\prod_{i=t+1}^{k} \sin^2(\theta_i),$$

where θ_i are the principal angles between the subspaces spanned by rows of P and Q. The integer parameter t indicates the size of the rank reduction in $F(X)$. If $t = 0$ then, of course, the volume reduces to the determinant.

In the next section we apply this generalized criterion on problems from system identification.

4 Application to system identification

The objective in system identification is to determine mathematical models of dynamical systems and processes based on measured observations. A system in this context could be an aircraft, a power plant or a car engine. Systems have the characteristics that they can be influenced by input signals[1] and depending on their inherent state they produce measurable output data. The measured input and output signals serve as source data in the computation of the model that relates the input signals to the output signals. The computed models are often used for control and simulations.

There are many different approaches to deal with system identification problems. A straight forward approach is to describe the present output at time t as a linear combination of previous inputs and outputs,

$$y(t) = a_1 y(t-1) + \cdots + a_n y(t-n) + b_1 u(t-1) + \cdots + b_m u(t-m).$$

Here we assume that the signals are sampled and use n and m previous samples of the output signal y and input signal u, respectively. A second approach is to describe a system as a state-space model[2],

$$\begin{aligned} x(t+1) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t) + Du(t), \end{aligned}$$

where x is the state vector and the matrices A, B, C and D, which describe the system, are to be determined during the identification process.

In many cases, given an input sequence, there is a difference in the output of the model compared to the actual output of the system. The constructed models attempt to minimize this difference in some measure. There are many other approaches and aspects concerning system identification with extensive literature background. A good overview is found in [59].

The second paper in this thesis [75, Paper II] uses the generalized rank reduction and volume minimization criterion in the framework of state-space subspace system identification, which can be seen as a linear regression multi-step ahead prediction error method [44]. The matrix form of the linear regression model can be written

$$Y_\alpha = L_1 Y_\beta + L_2 U_\beta + L_3 U_\alpha + E_\alpha,$$

where U_α, U_β are matrices with α future and β past input signals, Y_α, Y_β are, similarly, future and past output signals, and L_1, L_2, L_3 are the sought regression

[1] Input signals can often be specified but they can also include disturbances and noise.

[2] Actually this model contains additive noise terms in both equations but they are omitted here.

matrices. Computing the regression matrices gives the extended observability matrix from which the state-space matrices can be extracted. The rank constraint on the regression matrices comes from the order of the system matrix A.

Assuming that the input signal matrix U_α has full row rank[3] we can eliminate L_3 from the equations, and write the regression problem in the same form as (1.5), where now $[L_1\, L_2]$ is the regression matrix.

Without any assumptions on the measured signal matrices involved in the regression we identify three different scenarios where the determinant criterion is not sufficient. These are:

1. Absence of noise in all or part of the output signals,
2. Collinear output signals,
3. Direct linear dependencies between input and output signals.

We have demonstrated the validity of our algorithm using numerical experiments in a series of different situations, including cases with rank deficient and collinear data matrices. The generalized criterion gives the correct estimates in both the full rank case and the rank deficient case.

In applications the rank of the regression matrix $[L_1\, L_2]$ (or X) is often unknown and has to be determined. One additional advantage of our algorithm is that the appropriate rank can be chosen during the identification process by analyzing the singular values of $P^\mathsf{T}Q$, see equations (3.3) and (3.2).

5 Tensors and numerical multilinear algebra

5.1 Introduction to tensors

In numerical linear algebra, matrix computations and other mathematical sciences we mostly use quantities as vectors $x \in \mathbb{R}^n$ and matrices $A \in \mathbb{R}^{m\times n}$ for many different purposes. Vectors are usually elements of a vector space and matrices represent linear operators with respect to some basis, as the Hessian of a real valued function. Matrices also represent measured data, as a digital image or a collection of sensor signals. A vector is written as a one dimensional array of numbers and single index is used to address its entries. Similarly, a matrix is written as a two dimensional array of numbers whose entries are accessed with two indices.

An order n tensor $\mathcal{A} \in \mathbb{R}^{I_1\times\cdots\times I_n}$ is a generalization of these algebraic objects to one with n indices. Vectors and matrices are in fact first and second order tensors, respectively. The dimension of $\mathcal{A}$ along the different modes (or "directions" as rows and columns in a matrix) are given by I_i. Tensors are often found in differential geometry where they most of the time (if not exclusively) represent (abstract) multilinear operators. In this thesis tensors will, most of the time, constitute multidimensional data arrays, which we want to analyze, model and extract information from. In general tensors are difficult to visualize but we can imagine order three tensors as in Figure 5.1. The modes in a tensor have different meanings in different applications [73, 29, 86, 64, 21, 60]. Table 5.1 gives a few examples of their meanings. As we see these are essentially different properties of the data and it is unnatural to reshape the tensor into a matrix.

[3] In a controlled experiments one can chose input signals to fulfill this assumption.

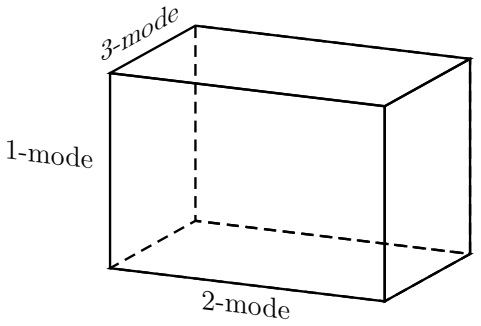


Figure 5.1: *Visualization of an order three tensor.*

It is important to distinguish the notation of tensors in the same way we distinguish matrices (written with capital letters) and vectors (written in lower case letters) in order to clarify the presentation. For this reason we will use capital letters with calligraphic font ($\mathcal{A}, \mathcal{B}, \ldots$) to denote tensors.

The usual way to analyze such data is to reshape the data into matrices and vectors and use the well developed theory of numerical linear algebra. On the other hand, we can find benefits in preserving the underlying structure of the data using theory and algorithms from numerical multilinear algebra. Tensor theory has been used for modeling in psychometrics and chemometrics [83, 39, 54, 55] for a long time now. Recently tensors have started to attract the attention from other fields of science as well [21, 23, 57, 25, 5, 52, 17, 19, 13, 69, 88, 85, 86, 60, 61, 71]. The main research topics are to analyze the basic properties of tensors, generalize the existing linear algebra theory to include tensors and construct efficient algorithms. The connection to applications is often very close.

5.2 Basic operations, tensor properties and notation

In this section we give some of the basic operations and properties which are the building blocks for the theory and algorithms. In the presentation we will use tensors of order three, four or five but the generalization to tensors of arbitrary order is straightforward.

Given a third order tensor $\mathcal{A} \in \mathbb{R}^{I \times J \times K}$ we write a_{ijk} to address its entries, where the subscripts i, j, k are ranging from one to I, J, K, respectively. The

Application	1-mode	2-mode	3-mode
Handwritten digit classification	pixels	samples	classes
Electronic nose data	sensors	time	gases
Computer vision	persons	pixel	view/illumination
Bioinformatics	genes	exp. variable	microarray

Table 5.1: *Examples of applications with tensors and the information content along the different modes.*

addition of two tensors $\mathcal{A}$ and $\mathcal{B}$ in $\mathbb{R}^{I\times J\times K}$ is

$$\mathcal{C} = \mathcal{A} + \mathcal{B}, \qquad c_{ijk} = a_{ijk} + b_{ijk}.$$

The multiplication of a tensor $\mathcal{A}$ with a scalar γ is the tensor

$$\mathcal{B} = \gamma\mathcal{A}, \qquad b_{ijk} = \gamma a_{ijk}.$$

The inner product between two tensors $\mathcal{A}$ and $\mathcal{B}$ of same dimensions is the scalar

$$\langle \mathcal{A}, \mathcal{B} \rangle = \sum_{i,j,k} a_{ijk} b_{ijk}.$$

The corresponding tensor norm is

$$\|\mathcal{A}\| = \langle \mathcal{A}, \mathcal{A} \rangle^{1/2}.$$

We exclusively use this Frobenius norm in this thesis. The outer product between two tensors $\mathcal{A} \in \mathbb{R}^{I\times J\times K}$ and $\mathcal{B} \in \mathbb{R}^{L\times M\times N}$ is an order six tensor,

$$\mathbb{R}^{I\times J\times K\times L\times M\times N} \ni \mathcal{C} = \mathcal{A} \circ \mathcal{B}, \qquad c_{ijklmn} = a_{ijk} b_{lmn}.$$

This operation is a generalization of the outer product between vectors x and y which results in the matrix xy^{T}.

5.3 Matrix-tensor multiplication

A matrix can be multiplied by other matrices on two sides – from left and from right. An order n tensor can be multiplied with matrices along each one of its modes. For example, multiplication of an order three tensor $\mathcal{A} \in \mathbb{R}^{I\times J\times K}$ by matrices $X \in \mathbb{R}^{L\times I}$,$Y \in \mathbb{R}^{M\times J}$ and $Z \in \mathbb{R}^{N\times K}$ is written and defined by

$$\mathbb{R}^{L\times M\times N} \ni \mathcal{B} = (X, Y, Z) \cdot \mathcal{A}, \qquad b_{lmn} = \sum_{i,j,k} x_{li} y_{mj} z_{nk} a_{ijk}.$$

If multiplication is done in just one or a few modes, we write

$$(X, Y)_{1,2} \cdot \mathcal{A} = (X, Y, I) \cdot \mathcal{A}.$$

With this notation we can write standard matrix multiplication of three matrices as

$$XFY^{\mathsf{T}} = (X, Y) \cdot F.$$

For convenience, we also introduce a separate notation for the multiplication by a transposed matrix $U \in \mathbb{R}^{I\times L}$,

$$\mathbb{R}^{L\times J\times K} \ni \mathcal{C} = \left(U^{\mathsf{T}}\right)_1 \cdot \mathcal{A} = \mathcal{A} \cdot (U)_1, \qquad c_{ljk} = \sum_i a_{ijk} u_{il}.$$

Multiplication in all modes with the transposed of the matrices U, V and W of appropriate dimensions is analogous,

$$\left(U^{\mathsf{T}}, V^{\mathsf{T}}, W^{\mathsf{T}}\right) \cdot \mathcal{A} = \mathcal{A} \cdot (U, V, W).$$

We want also to mention that there is a variety of notations for these manipulations, [47, 40, 23, 50, 17]. The notation we use for matrix-tensor products in all modes was suggested in [25] and for matrix multiplication along a few modes we are inspired by [8].

5.4 Canonical tensor matricization

Another concept that is useful when working with tensors is the notion of matricizing, unfolding or flattening of a tensor. This is an operation for transforming a given multidimensional array into a matrix. For example a 3-tensor $\mathcal{A} \in \mathbb{R}^{I \times J \times K}$ can be reshaped to form matrices of dimensions $I \times JK$, $J \times IK$ or $K \times IJ$ and each matrix has columns as specified in Figure 5.2.

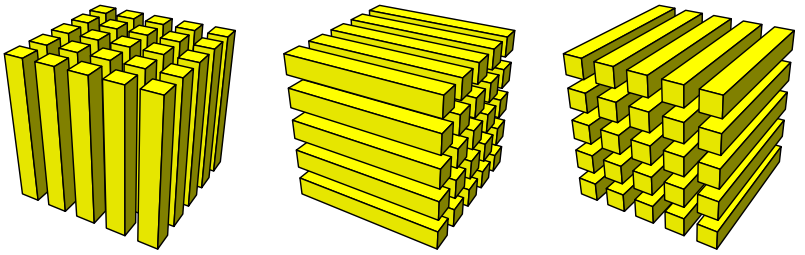


Figure 5.2: *Matricizing a 3-tensor results in matrices where the columns of the respective matrices are given by the illustrated fibers.*

Different matrices, with respect to column pivoting, will be obtained depending on which order the tensor fibers in each case are taken into the matricized forms. Consider references [47, 23] for two different examples. In [32, Paper IV] we introduce a canonical way of matricizing a tensor which is directly related to the matrix-tensor multiplication. For example, given a 5-tensor $\mathcal{A}$ and matrices V, W, X, Y, Z of appropriate dimensions, the matricization of the product $\mathcal{B} = \mathcal{A} \cdot (V, W, X, Y, Z)$ can be

$$B^{(2;1,3\ldots5)} \equiv B^{(2)} = W^{\mathsf{T}} A^{(2)} (V \otimes X \otimes Y \otimes Z), \qquad \mathsf{r} = [2],\ \mathsf{c} = [1,3,4,5],$$
$$B^{(3,2;1,4,5)} \equiv B^{(3,2)} = (X \otimes W)^{\mathsf{T}} A^{(3,2)} (V \otimes Y \otimes Z), \qquad \mathsf{r} = [3,2],\ \mathsf{c} = [1,4,5],$$

where $\otimes$ denotes the Kronecker product and r and c indicate the modes of the tensors that are mapped to the rows and columns in the matrix form, respectively. This matricization has been very useful when analyzing, deriving expressions and implementing algorithms for the Newton-Grassmann method. See [32, Section 2.2] (page 92) for a detailed discussion and more examples.

5.5 Contracted products and multilinear algebraic manipulations

Matrix-vector, matrix-matrix, and also matrix-tensor products are all examples of contracted products. It is necessary to generalize these products as well and define contracted products between two general tensors. Contracted products between tensors arise naturally when deriving the expressions for the gradient and the Hessian of an objective function for the low rank tensor approximation problem, which we will consider in Section 6.2. The only requirement is that the dimensions of the contracting modes are equal. For example, given $\mathcal{A} \in \mathbb{R}^{I \times J \times K \times L}$

and $\mathcal{B} \in \mathbb{R}^{I\times L\times M}$, we can define

$$\mathbb{R}^{J\times K\times L\times L\times M} \ni \mathcal{C} = \langle \mathcal{A}, \mathcal{B}\rangle_1 , \qquad c_{jkl_1l_2m} = \sum_i a_{ijkl_1} b_{il_2m}, \tag{5.1}$$

$$\mathbb{R}^{J\times K\times M} \ni \mathcal{D} = \langle \mathcal{A}, \mathcal{B}\rangle_{1,4;1,2} , \qquad d_{jkm} = \sum_{i,l} a_{ijkl} b_{ilm}. \tag{5.2}$$

In our case the tensors involved in the contractions are actually matrix-tensor products, see equations (6.10), (6.11) and (6.12). In order to properly matricize the derivative expressions for the algorithmic implementation of Newton and quasi-Newton methods, it is desirable to extract a matrix from a matrix-tensor product in a contraction. Given tensors $\mathcal{B}$ and $\mathcal{C}$ and a matrix Q with appropriate dimensions the following equality holds,

$$\langle \mathcal{B} \cdot (Q)_1 , \mathcal{C}\rangle_{-2} = \left\langle \langle \mathcal{B}, \mathcal{C}\rangle_{-(1,2)} , Q \right\rangle_{1,3;1,2},$$

where negative contraction subscript indicates that all but the specified modes are contracted. This is a special case from [32, Lemma (2.3)].

6 Tensor rank and low rank tensor approximation

In many applications involving tensor data the objective is to compute *low-rank* approximations of the data for modeling, information retrieval and explanatory purposes, [24, 22, 25, 42, 78, 89, 18, 16, 46, 49, 48, 76, 81]. These approximations are usually expressed in terms of tensor decompositions.

By a rank-one tensor we mean a tensor that can be factorized as the outer product of vectors. For example a 3-tensor of rank-one has the form $\mathcal{A} = x \circ y \circ z$. Observe that with only two vectors $x \circ y = xy^{\mathsf{T}}$ we obtain a rank-one matrix. The rank of a tensor can be defined as the minimal number of terms when expressing a tensor as a sum of rank-one tensors.

The second way to define a rank of a tensor is given by the dimension of the subspaces spanned by the different n-mode vectors. Given an order n tensor $\mathcal{A}$, we write

$$\operatorname{rank}(\mathcal{A}) = (r_1, \ldots, r_n), \qquad r_i = \dim\left(\operatorname{span}\left(A^{(i)}\right)\right),$$

where $A^{(i)}$ is the matricization of $\mathcal{A}$ along mode i. This is called the multilinear rank of a tensor. For tensors in general the ranks r_i are different. For a matrix A we have $r_1 = r_2 = \operatorname{rank}(A)$, since the row and column ranks of a matrix are equal. The theoretical properties of problems involving the multilinear rank of a tensor are much simpler than problems involving the outer-product tensor rank [25]. It can, for example, be shown that a sequence of rank-2 tensors can approximate a rank-3 tensor arbitrary well. In this thesis we have only considered the problem of approximating a given tensor by another tensor of lower multilinear rank.

The approximation problem of an order three tensor $\mathcal{A} \in \mathbb{R}^{I\times J\times K}$ is stated,

$$\min_{\mathcal{B}} \|\mathcal{A} - \mathcal{B}\|, \qquad \operatorname{rank}(\mathcal{B}) = (r_1, r_2, r_3). \tag{6.1}$$

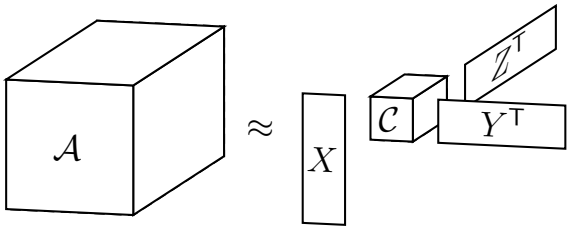


Figure 6.1: *The approximation of a tensor $\mathcal{A}$ by another tensor $\mathcal{B} = (X, Y, Z) \cdot \mathcal{C}$ of lower multilinear rank.*

Assuming the rank constraint on $\mathcal{B}$, we can decompose $\mathcal{B} = (X, Y, Z) \cdot \mathcal{C}$ where $X \in \mathbb{R}^{I \times r_1}$, $Y \in \mathbb{R}^{J \times r_2}$, $Z \in \mathbb{R}^{K \times r_3}$ have full column rank and $\mathcal{C} \in \mathbb{R}^{r_1 \times r_2 \times r_3}$. This is a *Tucker decomposition* of a tensor [82, 83]. The approximation problem, as well as the decomposition of $\mathcal{B}$ are visualized in Figure 6.1. The decomposition of $\mathcal{B}$ is in fact a consequence of the higher order singular value decomposition (HOSVD) [23].

Theorem 6.1 (HOSVD)**.** *Any 3-tensor $\mathcal{A} \in \mathbb{R}^{I \times J \times k}$ can be factorized*

$$\mathcal{A} = (U, V, W) \cdot \mathcal{S}, \tag{6.2}$$

where $U \in \mathbb{R}^{I \times I}$, $V \in \mathbb{R}^{J \times J}$, and $W \in \mathbb{R}^{K \times K}$, are orthogonal matrices, and the tensor $\mathcal{S} \in \mathbb{R}^{I \times J \times K}$ is all-orthogonal*: the matrices $\langle \mathcal{S}, \mathcal{S} \rangle_{-i}$, $i = 1, 2, 3$, are diagonal, and*

$$\|\mathcal{S}(1, :, :)\| \geq \|\mathcal{S}(2, :, :)\| \geq \cdots \geq 0, \tag{6.3}$$
$$\|\mathcal{S}(:, 1, :)\| \geq \|\mathcal{S}(:, 2, :)\| \geq \cdots \geq 0, \tag{6.4}$$
$$\|\mathcal{S}(:, :, 1)\| \geq \|\mathcal{S}(:, :, 2)\| \geq \cdots \geq 0, \tag{6.5}$$

are the 1-mode, 2-mode, and 3-mode singular values, also denoted $\sigma_i^{(1)}$, $\sigma_i^{(2)}$, $\sigma_i^{(3)}$.

The tensor $\mathcal{S}$ in the HOSVD is in general full and not sparse or diagonal as Σ in the SVD of a matrix, see Theorem 3.1. But the all-orthogonality concept is still valid in matrix SVD. The singular values in (6.3) correspond to norms of row vectors from Σ and those in (6.4) correspond to norms of column vectors, which for matrix SVD are equal since Σ is diagonal. In addition any two different column or row vectors from Σ are orthogonal to each other. The HOSVD is an important result both for analysis and applications with tensors since it gives an ordering of the basis vectors in U, V and W. The ordering also implies that elements of largest magnitude in $\mathcal{S}$ are concentrated towards the $(1, 1, 1)$ corner. The entries in $\mathcal{S}$ generally decay when going away in all directions from this corner.

In fact truncating the HOSVD, i.e. taking the first r_1, r_2 and r_3 columns from U, V and W, respectively and correspondingly truncating $\mathcal{S}$, will give (hopefully) a good approximation of $\mathcal{A}$. The difference compared to the matrix case is that the truncated HOSVD is not the solution of (6.1) [24]. From an application point of view this may still be good enough for different purposes.

There are other higher order generalizations of the SVD, singular values/vectors as well as eigenvalues/vectors of tensors [15, 39, 57, 66, 67, 68].

6.1 Application of truncated higher order SVD to handwritten digit classification

Pattern recognition and classification is one of the main topics in data mining and there are many different ways to approach and solve the encountered problems. Classical approaches are the nearest and k-nearest neighbor methods. The idea is to interpret objects as vectors or elements in a finite dimensional linear space and compute some kind of proximity measure between known and unknown objects. The label of the closest or k closest known objects determine the label of the unknown object to be classified. A second and intuitive approach is to find linear or nonlinear functions partitioning the space into disjoint regions where objects from different classes belong to a certain region. Other ways of solving pattern recognition problems include linear algebra methods (least squares, eigenvalue problems, data compression and feature extraction by low rank approximation), statistical methods (maximum likelihood estimation, linear and nonlinear regression, support vector machines) and neural networks. Good references for the mentioned methods are [26, 41, 30].

In [73, Paper III] we use the HOSVD for reduction of data consisting of handwritten digits. The data have pixels along the first mode, samples along the second mode and classes along the third mode. We show that we can make a 98% compression of the data without losing classification performance. This is done by reducing the pixel dimension from 400 to 30–60 and the sample dimension from 1000 to 30–60 as well. After the reduction each digit is described with only 30–60 pixels and only 30–60 samples represent the variation within each class. The class-models are given by the dominant pixel-subspaces obtained by regular SVD on the reduced data. An unknown digit is projected on each one of the 10 subspaces and the subspace with smallest residual gives the output of the algorithm. Straightforward implementation of this method gives a classification error of 5%.

6.2 Best low rank tensor approximation

In [32, Paper IV] and [74, Paper V] we develop Newton and quasi-Newton algorithms for computing the best multilinear low-rank tensor approximation. Rewriting the tensor approximation problem (6.1) with the decomposition $\mathcal{B} = (X, Y, Z) \cdot \mathcal{C}$ we get

$$\min_{X,Y,Z,\mathcal{C}} \|\mathcal{A} - (X, Y, Z) \cdot \mathcal{C}\|.$$

One can show that this problem is overparameterized and with little algebraic manipulations we show (see [32, Section 3] and [24]) that the tensor approximation problem is equivalent to

$$\max_{X,Y,Z} \|\mathcal{A} \cdot (X, Y, Z)\|, \tag{6.6}$$

where now the matrices X, Y and Z have orthonormal columns, i.e. $X^\mathsf{T}X = I_{r_1}$, $Y^\mathsf{T}Y = I_{r_2}$ and $Z^\mathsf{T}Z = I_{r_3}$. In addition, expression (6.6) only depends on the subspaces spanned by columns of X, Y and Z, thus it is invariant to the specific basis representation of a subspace. In other words the following equality holds,

$$\|\mathcal{A} \cdot (X, Y, Z)\| = \|\mathcal{A} \cdot (XU, YV, ZW)\|,$$

where U, V and W are orthogonal matrices. It follows that the maximization problem is defined on a product of three Grassmann manifolds. A point on a Grassmann manifold $\mathrm{Gr}(n,r)$ consists of all $n \times r$ matrices with orthonormal columns that span the same subspace, we write

$$\mathrm{Gr}(n,r) \ni [X] = \{XU : U \text{ orthogonal}\}.$$

These observations lead to the following nonlinear maximization problem constrained in the variables X, Y and Z;

$$\max_{(X,Y,Z)\in \mathrm{Gr}^3} \Phi(X,Y,Z), \qquad \mathrm{Gr}^3 = \mathrm{Gr}(I,r_1) \times \mathrm{Gr}(J,r_2) \times \mathrm{Gr}(K,r_3), \tag{6.7}$$

where

$$\Phi(X,Y,Z) = \frac{1}{2} \left\| \mathcal{A} \cdot (X,Y,Z) \right\|^2 = \frac{1}{2} \sum_{j,k,l} \left(\sum_{\lambda,\mu,\nu} a_{\lambda\mu\nu} x_{\lambda j} y_{\mu k} z_{\nu l} \right)^2. \tag{6.8}$$

This expression is a polynomial of degree six. In the next sections we will briefly describe algorithms for maximizing Φ.

6.3 Optimization on a product of Grassmann manifolds

If the objective function in an optimization problem is defined on a manifold, one needs to take into account the nature of the manifold when constructing algorithms [27, 34, 3, 12, 77, 2, 1, 43]. In particular we need to modify:

1. The gradient and Hessian of the function,
2. The update of the current iterate in a given direction,
3. The update of a vector, if it is computed at one point on the manifold but used on another point.

These modifications are required both for theoretical and practical reasons. All computations involved in the algorithms are performed on the tangent spaces for different points of the manifolds. Thus given a point $X \in \mathrm{Gr}(n,r)$ the gradient of a function $f(X)$ is a vector in the tangent space $\mathbb{T}_X$ to the manifold at X, i.e.

$$\nabla f(X) \in \mathbb{T}_X.$$

Similarly, the Grassmann Hessian H_f of the function f at X is an operator mapping vectors from $\mathbb{T}_X$ to vectors in $\mathbb{T}_X$,

$$H_f : \mathbb{T}_X \to \mathbb{T}_X.$$

If we consider the objective function for the tensor approximation problem, $\Phi(X,Y,Z)$, we will have three different tangent spaces, $\mathbb{T}_X$, $\mathbb{T}_Y$ and $\mathbb{T}_Z$. In this case the Hessian will involve linear operators from each tangent space to every other tangent space. Writing the Hessian of Φ as

$$H_\Phi = \begin{pmatrix} H_{xx} & H_{xy} & H_{xz} \\ H_{yx} & H_{yy} & H_{yz} \\ H_{zx} & H_{zy} & H_{zz} \end{pmatrix},$$

we will have $H_{xx} : \mathbb{T}_X \to \mathbb{T}_X$, $H_{xy} : \mathbb{T}_Y \to \mathbb{T}_X$, $H_{zy} : \mathbb{T}_Y \to \mathbb{T}_Z$ and similarly for the other parts. On manifolds we choose to move from a point along geodesic curves that are defined by the direction of movement, which is vector in the tangent space. The geodesic never leaves the manifold and corresponds to a straight line in Euclidean spaces. Given a point $X \in \mathrm{Gr}(n, r)$ and direction $\Delta \in \mathbb{T}_X$, the explicit expression of a geodesic on the Grassmann manifold has the form

$$X_\Delta(t) = XV \cos(t\Sigma)V^\mathsf{T} + U \sin(t\Sigma)V^\mathsf{T}, \tag{6.9}$$

where U, Σ and V are obtained from the thin singular value decomposition of Δ. If there is a second tangent vector Δ_2 at X the parallel transport of it along the geodesic is given by

$$\mathbb{T}_{X(t)} \ni \Delta_2(t) = \left(\begin{pmatrix} XV & U \end{pmatrix} \begin{pmatrix} -\sin \Sigma t \\ \cos \Sigma t \end{pmatrix} U^\mathsf{T} + (I - UU^\mathsf{T}) \right) \Delta_2.$$

This kind of parallel transport is necessary in quasi-Newton algorithms. For example in Euclidean space we would need to compute $\nabla f(X_1) - \nabla f(X_2)$ but on manifolds the two terms are defined on different points of the manifold, and thus belong to different tangent spaces. In order to perform this computation we have to parallel transport one of the gradients to the tangent space of the other, then we can perform the subtraction.

6.4 The Grassmann gradient and the the Grassmann Hessian

The expressions for the gradient and the Hessian of a function are derived from the Taylor expansion, in which the linear term gives the gradient and the quadratic term gives the Hessian. With the following terms $\mathcal{F} = \mathcal{A} \cdot (X, Y, Z)$, $\widehat{\mathcal{B}}_x = \mathcal{A} \cdot (X_\perp, Y, Z)$, $\widehat{\mathcal{B}}_y = \mathcal{A} \cdot (X, Y_\perp, Z)$ and $\widehat{\mathcal{B}}_z = \mathcal{A} \cdot (X, Y, Z_\perp)$ we can write the Grassmann gradient in local coordinates as

$$\nabla \Phi = \left(\langle \widehat{\mathcal{B}}_x, \mathcal{F} \rangle_{-1}, \langle \widehat{\mathcal{B}}_y, \mathcal{F} \rangle_{-2}, \langle \widehat{\mathcal{B}}_z, \mathcal{F} \rangle_{-3} \right). \tag{6.10}$$

The matrices $X_\perp$, $Y_\perp$ and $Z_\perp$ are the orthogonal complements of X, Y and Z, respectively. The columns of the orthogonal complements form in fact a basis for each tangent space, for example $\mathbb{T}_X = \mathrm{span}(X_\perp)$. One can also show that the following eigenspace-like equations are valid at a stationary point of the objective function, i.e. when $\nabla \Phi = 0$,

$$\begin{aligned}
\langle \mathcal{A} \cdot (I, Y, Z), \mathcal{A} \cdot (I, Y, Z) \rangle_{-1} X &= X \langle \mathcal{F}, \mathcal{F} \rangle_{-1}, \\
\langle \mathcal{A} \cdot (X, I, Z), \mathcal{A} \cdot (X, I, Z) \rangle_{-2} Y &= Y \langle \mathcal{F}, \mathcal{F} \rangle_{-2}, \\
\langle \mathcal{A} \cdot (X, Y, I), \mathcal{A} \cdot (X, Y, I) \rangle_{-3} Z &= Z \langle \mathcal{F}, \mathcal{F} \rangle_{-3}.
\end{aligned}$$

Writing the gradient in matrix form yields,

$$\begin{aligned}
\langle \widehat{\mathcal{B}}_x, \mathcal{F} \rangle_{-1} &= X_\perp^\mathsf{T} A^{(1)} (Y \otimes Z)(Y \otimes Z)^\mathsf{T} (A^{(1)})^\mathsf{T} X, \\
\langle \widehat{\mathcal{B}}_y, \mathcal{F} \rangle_{-2} &= Y_\perp^\mathsf{T} A^{(2)} (X \otimes Z)(X \otimes Z)^\mathsf{T} (A^{(2)})^\mathsf{T} Y, \\
\langle \widehat{\mathcal{B}}_z, \mathcal{F} \rangle_{-2} &= Z_\perp^\mathsf{T} A^{(3)} (X \otimes Y)(X \otimes Y)^\mathsf{T} (A^{(3)})^\mathsf{T} Z,
\end{aligned}$$

where $A^{(1)}$, $A^{(2)}$ and $A^{(3)}$ are matricizations of $\mathcal{A}$.

The Grassmann Hessian is more complex than the gradient, but it also has a symmetric structure and consists of contracted tensor products. Here we give two blocks of the Hessian operator acting on tangent vectors[4] D_x, D_y and visualize how they are computed. In local coordinate the xx- and xy-part of the Grassmann Hessian acting on coordinate matrices have the form

$$\widehat{\mathcal{H}}_{xx}(D_x) = \left\langle \widehat{\mathcal{B}}_x, \widehat{\mathcal{B}}_x \right\rangle_{-1} D_x - D_x \left\langle \mathcal{F}, \mathcal{F} \right\rangle_{-1}, \tag{6.11}$$

$$\widehat{\mathcal{H}}_{xy}(D_y) = \left\langle \left\langle \widehat{\mathcal{C}}_{xy}, \mathcal{F} \right\rangle_{-(1,2)}, D_y \right\rangle_{2,4;1,2} + \left\langle \left\langle \widehat{\mathcal{B}}_x, \widehat{\mathcal{B}}_y \right\rangle_{-(1,2)}, D_y \right\rangle_{4,2;1,2}, \tag{6.12}$$

where we also introduce $\widehat{\mathcal{C}}_{xy} = \mathcal{A} \cdot (X_\perp, Y_\perp, Z)$. Figure 6.2 shows all tensor-matrix products involved in the computation of the Hessian. These are blocks of

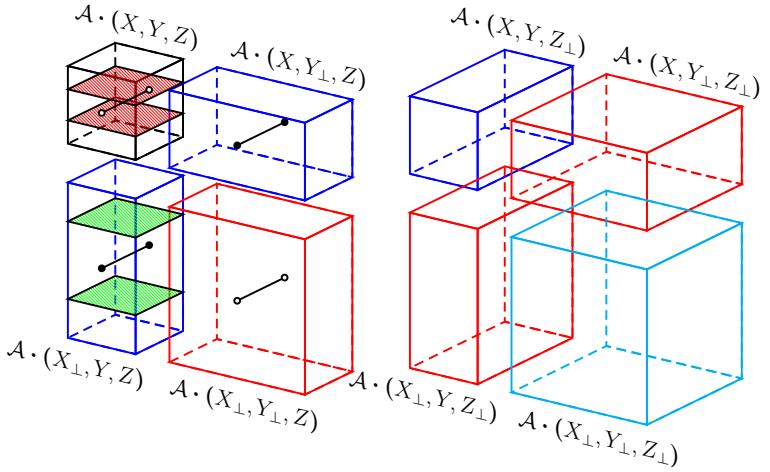


Figure 6.2: *Illustration of the tensor-matrix products involved in the computation of the Hessian. The matrix slices are used to compute xx-part and the fibers along the third mode are used to compute xy-part of the Hessian.*

$\mathcal{A} \cdot ((X\, X_\perp), (Y\, Y_\perp), (Z\, Z_\perp))$. Recall that the objective is to maximize the norm of $\mathcal{A} \cdot (X, Y, Z)$.

The first term in (6.11) is a matrix since it is a contraction in all but the first mode between $\widehat{\mathcal{B}}_x$ and itself. Each entry in the resulting matrix is the scalar product between two slices in $\mathcal{A} \cdot (X_\perp, Y, Z)$. Similarly, the second term $\langle \mathcal{F}, \mathcal{F} \rangle_{-1}$ is also a matrix whose elements are different scalar products between slices of $\mathcal{A} \cdot (X, Y, Z)$. See the corresponding blocks in Figure 6.2.

The first term in (6.12) involves two different contractions resulting in a matrix. The first one is a contraction between the 3-tensors $\widehat{\mathcal{C}}_{xy}$ and $\mathcal{F}$ in all but the first and second mode, thus in this case contraction in the third mode. This results in a 4-tensor whose elements are scalar products between third mode fibers from each tensor. In Figure 6.2 these are illustrated with the ⟷ symbol. In the

[4] These are actually matrices but represent the coordinates of elements on the tangent space.

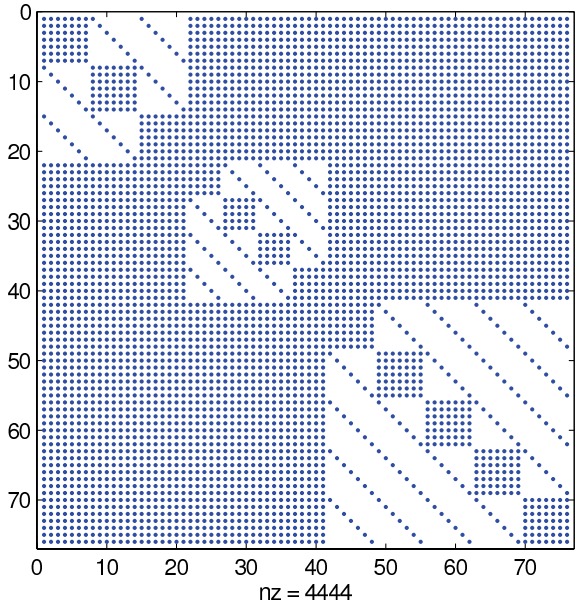


Figure 6.3: *Illustration of the sparsity pattern in the matricized Hessian. The diagonal blocks are sparse but these are coupled tightly together with non-sparse off-diagonal blocks.*

second contraction we multiply the 4-tensor $\langle \widehat{\mathcal{C}}_{xy}, \mathcal{F} \rangle_{-(1,2)}$ with the matrix D_y and contract second and fourth mode of the tensor with the first and second mode of the matrix which results in a matrix. Similarly, in the second term of (6.12), the third mode contractions are done between the tensors $\widehat{\mathcal{B}}_x = \mathcal{A} \cdot (X_\perp, Y, Z)$ and $\widehat{\mathcal{B}}_y = \mathcal{A} \cdot (X, Y_\perp, Z)$, these are symbolized with •——• in Figure 6.2.

The remaining blocks of the Hessian are computed in a analogous fashion. The contractions are in other modes and involve other matrix-tensor product blocks. Consider Sections 4.2 and 4.3 in [32] for further details.

The zero structure of the matricized Hessian is given in Figure 6.3 for a test example of a $10 \times 9 \times 12$ tensor approximated by a rank-$(3, 4, 5)$ tensor. Introducing zeros in the matrices X, Y, Z and the basis $X_\perp, Y_\perp, Z_\perp$ for the tangent spaces $\mathbb{T}_X, \mathbb{T}_Y, \mathbb{T}_Z$, respectively, by applying coordinate transformations does not result in a sparser Hessian structure.

6.5 Newton-Grassmann and quasi-Newton-Grassmann algorithms

In this thesis we introduce three new methods for computing the best low rank approximation of a tensor; a Newton-Grassmann algorithm, a quasi-Newton-Grassmann algorithm based on the BFGS updates and an algorithm based on the limited memory BFGS updates. These are standard algorithms for unconstrained optimization problems [63]. The novelty in this presentation is the incorporation of the product Grassmann manifold structure, inherent in the tensor approximation problem, into the algorithms.

The iterative framework of the three algorithms is similar and originates from the Newton equations,

$$H_k p_k = -\nabla f_k, \tag{6.13}$$

which is derived from a second order Taylor approximation of an objective function $f(x)$ with goal to maximize it. The algorithms are started with an initial approximation x_0 of a local minimizer and in each iteration one computes a direction of movement p_k. A better approximation is obtained with the update $x_{k+1} = x_k + t_k p_k$, where t_k is step length of movement. Of course, if x_k is a point on a manifold then the appropriate update is to move along a geodesic in the direction p_k. Figure 6.4 illustrates this on the Grassmann manifold $\mathrm{Gr}(3,1)$ which is simply the sphere in $\mathbb{R}^3$. The computations of the direction of movements

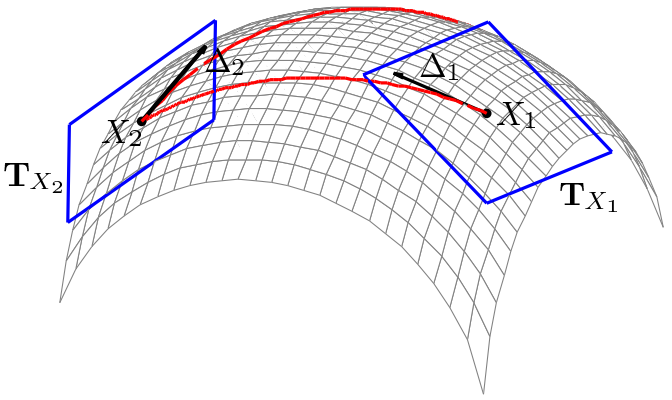


Figure 6.4: *Illustration of geodesic movement form X_1 in the direction given by Δ_1. This is part of a sphere in $\mathbb{R}^3$ which is the Grassmann manifold $\mathrm{Gr}(3,1)$. The geodesics on the sphere are given by great circles. Also the squares indicate the tangent spaces at which the computations are performed.*

are performed on the tangent spaces, illustrated with a square, at each iterate and the update is a movement along the great circles defined by the tangent vectors, Δ_1 and Δ_2 in Figure 6.4.

The different algorithms differ in the way H_k is computed. In Newton's method H_k is the exact Hessian computed at each new iterate. In quasi-Newton methods one instead updates the current Hessian approximation by a rank-2 matrix modification. One looses on convergence rate but gains in computation time. Quasi-Newton methods with BFGS updates are recognized to be one of the best performing algorithms for general medium sized unconstrained nonlinear optimization problems [63]. If the problems are large the actual Hessian approximation $H_k \in \mathbb{R}^{n\times n}$ may be too big to fit in the computer memory. In limited memory BFGS methods this is overcome by expressing H_k in a compact form using very few vectors of size n, [14].

Figure 6.5 contains two plots illustrating the convergence of the different algorithms compared to higher order orthogonal iteration (HOOI), which is related to alternating least square methods (ALS). The setup for the left figure is as follows:

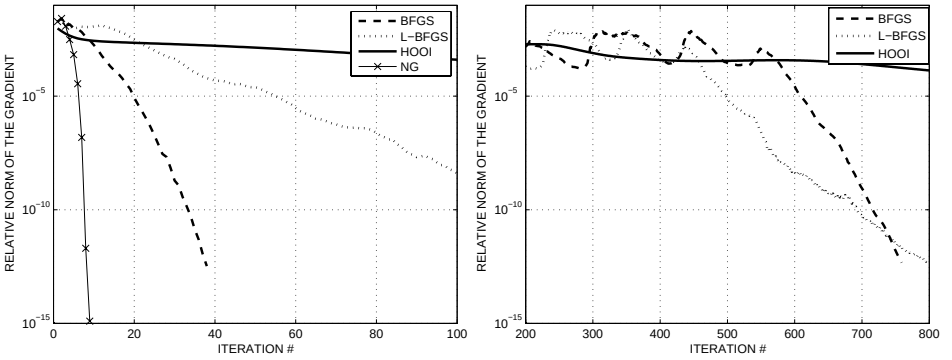


Figure 6.5: *Left: A* $20 \times 20 \times 20$ *tensor with random entries* $(\mathbf{N}(0,1))$ *is approximated with a rank-*$(5,5,5)$ *tensor. Right: A* $100 \times 100 \times 100$ *tensor with random entries* $(\mathbf{N}(0,1))$ *is approximated with a rank-*$(5,10,20)$ *tensor.*

A random tensor $\mathcal{A} \in \mathbb{R}^{20\times 20\times 20}$ with random entries $(\mathbf{N}(0,1))$ is approximated with a rank-$(5,5,5)$ tensor. All algorithms are initialized with truncated HOSVD. The QN method with BFGS updates was initialized with the exact Hessian whereas the limited memory BFGS algorithm uses scaled identity matrix as initial Hessian approximation. We observe a quadratic convergence for the Newton-Grassmann algorithm and superlinear convergence for the BFGS algorithm. The right figure shows convergence for a random tensor $\mathcal{A} \in \mathbb{R}^{100\times 100\times 100}$ approximated with a rank-$(5,10,20)$ tensor. All three algorithms are initiated with truncated HOSVD and 50 HOOI iterations. Both BFGS and L-BFGS methods used a scaled identity as initial Hessian approximation. The second problem is too big for the Newton-Grassmann algorithm, therefore not present in the right figure.

7 Future research directions

In this section I will give a short presentation with open questions on different research topics related to tensors.

7.1 Multilinear systems of equations

The theoretical properties of systems of multilinear equations need to be investigated. Consider the generalization

$$\mathcal{A} \cdot (y,z)_{2,3} = b \quad \Leftrightarrow \quad A^{(1)}\,(y \otimes z) = b,$$

where $\mathcal{A} \in \mathbb{R}^{I\times J\times K}$ and $b \in \mathbb{R}^I$ are given while $y \in \mathbb{R}^J$ and $z \in \mathbb{R}^K$ are unknown. What properties of $\mathcal{A}$ and b determine the existence of a solution? Obviously it is not enough for b to belong to the range space of $A^{(1)}$. If the multilinear equations are consistent when is the solution unique? Is there a close form solution? What about solutions to

$$\min_{y,z} \|\mathcal{A} \cdot (y,z)_{2,3} - b\| \quad \Leftrightarrow \quad \min_{y,z} \|A^{(1)}\,(y \otimes z) - b\|.$$

Generalizing these equations to involve higher order tensors is obvious.

For linear systems of equations, e.g. $Ax = b$, the accuracy of the computed solution x is determined by the condition number of the matrix, $\kappa(A) = \|A\|\|A^{-1}\|$. What is the condition number or the inverse of an order n-tensor? What is the stability in numerical computations with tensors regarding disturbances?

7.2 Convergence of alternating least squares methods

The convergence properties of alternating least squares (ALS) methods [54] need to be investigated, since it is probably the most widely used method to solve multilinear problems. The method converges linearly but the limit may not be a stationary point [70, 63, 65]. ALS methods converge very rapidly, when approximating a noisy signal tensor with the correct rank or tensors generated by sampling potential functions of three variables [46]. Experiments indicate that the convergence rate might be related to the multilinear singular values obtained from the HOSVD. The exact tensor properties that determine the asymptotic rate of convergence are unknown.

7.3 Computations with large and sparse tensors

In applications tensors may have large dimensions in some of its modes or be large and sparse [52, 62, 84, 4]. Software supporting sparse and factored tensors already exist [9]. Alternative algorithms are needed that are better suited for large dense tensors as well as large sparse tensors. Parallel implementation of algorithms can also address these large problems on supercomputers but also on regular computers with multicore processors. Multilinear algebra package corresponding to the Linear Algebra PACKage – LAPACK would be useful as well.

7.4 Attempts for the global minimum

Can anything be said about the global maximum of the tensor approximation problem written in the form

$$\max_{X,Y,Z} \|\mathcal{A} \cdot (X, Y, Z)\|, \qquad X^{\mathsf{T}}X = I, \quad Y^{\mathsf{T}}Y = I, \quad Z^{\mathsf{T}}Z = I. \tag{7.1}$$

Different algorithms look for a local stationary point and at best deliver one. Is it possible to determine if a given stationary point is the global maximizer of the objective function? The objective function is "only" a polynomial of degree six but with a large number of variables. And as such it has at most a certain number of local maximum points. How many stationary points are there? Can we compute all or a subset of them?

7.5 Other multilinear models

We have considered only the Tucker decom-position in this thesis. Another fundamental decomposition is the CANDECOMP/PARAFAC (CANonical DECOMPosition and PARAllel FACtor) decomposition (CP) [15, 39] which is a different generalization of the singular value decomposition to tensors. The CP model is related to the outer-product rank of a tensors. These and several other decompositions are overviewed in [51]. Theoretical properties of the different decompositions need to be further investigated as well as efficient computational methods need to be developed.

2
Summary of papers

Paper I
Dimensionality reduction and volume minimization – generalization of the determinant minimization criterion for reduced rank regression problems

In this paper we generalize the determinant minimization criterion for the reduced rank regression problem into dimensionality reduction of the objective matrix and then volume minimization, where volume of a matrix is defined as the product of its nonzero singular values [10]. The determinant criterion is appealing since it is closely related to maximum likelihood estimation [59, 44] but it has implicit assumptions not allowing rank deficiency in the objective matrix [80, 87]. The analysis reveals several cases where rank deficiency is obtained.

Paper II
Rank reduction and volume minimization approach to state-space subspace system identification

In the second paper we consider the generalized minimization criterion for the reduced rank regression problem in the context of state-space subspace system identification. The theoretical background of the algorithms solving the regression problem rely on the presence of noise to ensure that the full rank assumption is fulfilled [80, 44]. If this is not true both the theory and algorithm will fail and yet there are cases where rank deficiency is only natural. We identify the different cases of rank deficiency in terms of properties in the measured signals. We also show that the given algorithm deals implicitly with all these cases.

Paper III
Handwritten digit classification using higher order singular value decomposition

In this paper we discuss two classification algorithms based on higher order singular value decomposition (HOSVD) [23]. The first algorithm uses HOSVD to construct a set of orthonormal basis matrices spanning the dominant subspace for each class. The basis matrices serve as models for the different classes. Classification is conducted by expressing the unknown digit as a linear combination

with class specific basis matrices. In the second algorithm HOSVD is used to make a data reduction prior the construction of the class models. The achieved classification rate is 95% despite the 98%–99% data reduction.

Paper IV
A Newton-Grassmann method for computing the best multilinear rank-(r_1, r_2, r_3) approximation of a tensor

In this paper we derive a Newton method for computing the best rank-(r_1, r_2, r_3) approximation of a given $I \times J \times K$ tensor $\mathcal{A}$. The problem is formulated as an optimization problem defined on a product of Grassmann manifolds. Incorporating the manifold structure into Newton's method ensures that all iterates generated by the algorithm are points on the Grassmann manifolds. We also introduce a consistent notation for matricizing a tensor, for contracted tensor products and some tensor-algebraic manipulations, which simplify the derivation of the Newton equations and enable straightforward algorithmic implementation. Experiments show a quadratic convergence rate for the Newton-Grassmann algorithm.

Paper V
Best multilinear rank approximation of tensors with quasi-Newton methods on Grassmannians

In this report we present quasi-Newton methods for computing the best rank-(r_1, r_2, r_3) approximation of a given tensor. Both the general and the symmetric case are considered. We consider algorithms based on BFGS and limited memory BFGS updates modified to operate on a product of Grassmann manifolds. We give a complexity analysis of the different algorithms both in local and global coordinate implementations. The presented BFGS algorithms are compared with the Newton-Grassmann and alternating least squares (ALS) methods. Experiments show that the performance of the quasi-Newton methods is much better than the other methods when applied on difficult problems.

References

[1] P.-A. Absil, C. G. Baker, and K. A. Gallivan. Trust-region methods on Riemannian manifolds. *Foundations of Computational Mathematics*, 7(3):303–330, 2007.

[2] P.-A. Absil, R. Mahony, and R. Sepulchre. Riemannian geometry of Grassmann manifolds with a view on algorithmic computation. *Acta Appl. Math.*, 80(2):199–220, 2004.

[3] P.-A. Absil, R. Mahony, and R. Sepulchre. *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, Princeton, NJ, January 2008.

[4] E. Acar and B. Yener. Unsupervised multiway data analysis: A literature survey. Technical report, Computer Science Department, Rensselaer Polytechnic Institute, 2007.

[5] O. Alter, P.O. Brown, and D. Botstein. Generalized singular value decomposition for comparative analysis of genome-scale expression data sets of two different organisms. *PNAS*, 100(6):3351–3356, March 2003.

[6] T. W. Anderson. Canonical correlation analysis and reduced rank regression in autoregressive models. *Ann. Statist.*, 30(4):1134–1154, 2002.

[7] T. W. Anderson. *An Introduction to Multivariate Statistical Analysis*. Wiley Interscience, 3 edition, 2003.

[8] B. W. Bader and T. G. Kolda. Algorithm 862: MATLAB tensor classes for fast algorithm prototyping. *ACM Trans. Math. Softw.*, 32(4):635–653, 2006.

[9] B. W. Bader and T. G. Kolda. Efficient MATLAB computations with sparse and factored tensors. *SIAM Journal on Scientific Computing*, 30(1):205–231, 2007.

[10] A. Ben-Israel. A volume associated with $m \times n$ matrices. *Linear Algebra and its Applications*, 167:87–111, 1992. Sixth Haifa Conference on Matrix Theory (Haifa, 1990).

[11] Å. Björck. *Numerical Methods for Least Squares Problems*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1996.

[12] W. M. Boothby. *An Introduction to Differentiable Manifolds and Riemannian Geometry*, volume 120 of *Pure and Applied Mathematics*. Academic Press Inc., Orlando, FL, second edition, 1986.

[13] R. Bro. *Multi-way Analysis in the Food Industry Models, Algorithms, and Applications*. PhD thesis, University of Amsterdam (NL), 1998.

[14] R. H. Byrd, J. Nocedal, and R. B. Schnabel. Representations of quasi-Newton matrices and their use in limited memory methods. *Math. Programming*, 63(2, Ser. A):129–156, 1994.

[15] J. D. Carroll and J. J. Chang. Analysis of individual differences in multidimensional scaling via an n-way generalization of "Eckart-Young" decomposition. *Psychometrika*, 35:Psychometrika, 1970.

[16] S. R. Chinnamsetty, M. Espig, B. N. Khoromskij, and W. Hackbusch. Tensor product approximation with optimal rank in quantum chemistry. *The Journal of Chemical Physics*, 127(084110):14, 2007.

[17] P. Comon. Tensor decompositions. In J. G. McWhirter and I. K. Proudler, editors, *Mathematics in Signal Processing V*, pages 1–24. Clarendon Press, Oxford, UK, 2002.

[18] P. Comon, G. Golub, L.-H. Lim, and B. Mourrain. Symmetric tensor and symmetric tensor rank. SIAM J. Matrix Anal. Appl. to appear.

[19] P. Comon, G. Golub, L.-H. Lim, and B. Mourrain. Genericity and rank deficiency of high order symmetric tensors. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '06)*, volume 31, pages 125–128, 2006.

[20] D. J. Cook and L. B. Holder, editors. *Mining Graph Data*. Wiley, 2007.

[21] L. De Lathauwer. *Signal Processing Based on Multilinear Algebra*. PhD thesis, K. U. Leuven, Department of Electrical Engineering (ESAT), 1997.

[22] L. De Lathauwer, L. Hoegaerts, and J. Vandewalle. A Grassmann-Rayleigh quotient iteration for dimensionality reduction in ICA. In *Proc. 5th Int. Workshop on Independent Component Analysis and Blind Signal Separation (ICA 2004)*, pages 335–342, Granada, Spain, Sept. 2004.

[23] L. De Lathauwer, B. De Moor, and J. Vandewalle. A multilinear singular value decomposition. *SIAM Journal on Matrix Analysis and Applications*, 21(4):1253–1278, 2000.

[24] L. De Lathauwer, B. De Moor, and J. Vandewalle. On the best rank-1 and rank-$(R_1, R_2, \ldots, R_N)$ approximation of higher-order tensors. *SIAM Journal on Matrix Analysis and Applications*, 21(4):1324–1342, 2000.

[25] V. de Silva and L.-H. Lim. Tensor rank and the ill-posedness of the best low-rank approximation problem. *SIAM J. Matrix Anal. Appl., to appear*, 2007.

[26] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. Wiley-Interscience, New York, second edition, 2001.

[27] A. Edelman, T.A. Arias, and S.T. Steven. The geometry of algorithms with orthogonality constraints. *SIAM J. Matrix Anal. Appl.*, 20(2):303–353, 1999.

[28] L. Eldén. Numerical linear algebra in data mining. *Acta Numerica*, 15:327–384, 2006.

[29] L. Eldén. Decomposition of a 3-way array of data from the electronic nose baseline–corrected data. Manuscript, March 2002.

[30] L. Eldén. *Matrix Methods in Data Mining and Pattern Recognition*. SIAM, 2007.

[31] L. Eldén and B. Savas. The maximum likelihood estimate in reduced-rank regression. *Numerical Linear Algebra with Applications*, 12:731–741, 2005.

[32] L. Eldén and B. Savas. A Newton-Grassmann method for computing the best multilinear rank-(r_1, r_2, r_3) approximation of a tensor. Technical Report LITH-MAT-R-2007-6-SE, Department of Mathematics, Linköpings Universitet, 2007.

[33] I. Foster. The grid: A new infrastructure for 21st century science. *Physics Today*, 2002.

[34] D. Gabay. Minimizing a differentiable function over a differential manifold. *Journal of Optimization Theory and Applications*, 37(2):177–219, 1982.

[35] G. H. Golub and C. F. Van Loan. *Matrix Computations*. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, Baltimore, MD, third edition, 1996.

[36] R. L. Grossman, C. Kamath, P. Kegelmeyer, V. Kumar, and R. R. Naburu, editors. *Data Mining for Scientific and Engineering Applications*, Massive computing. Kluwer Academic Publishers, 2001.

[37] J. Han and M. Kamber. *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers, San Francisco, 2001.

[38] D. Hand, H. Mannila, and P. Smyth. *Principles of Data Mining*. MIT Press, Cambridge, MA, 2001.

[39] R. A. Harshman. Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multi-modal factor analysis. *UCLA Working Papers in Phonetics*, 16:1–84, 1970.

[40] R. A. Harshman. An index formalism that generalizes the capabilities of matrix notation and algebra to n-way arrays. *J. Chemometrics*, (15):689–714, 2001.

[41] T. Hastie, R. Tibshirani, and J Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics. Springer-Verlag, New York, 2001.

[42] F. L. Hitchcock. Multiple invariants and generalized rank of a p-way matrix or tensor. *J. Math. Phys. Camb.*, 7:39–70, 1927.

[43] M. Ishteva, L. De Lathauwer, P.-A. Absil, and S. Van Huffel. Dimensionality reduction for higher-order tensors: algorithms and applications. Internal Report 07-187, ESAT-SISTA, K.U.Leuven (Leuven, Belgium), 2007. submitted to: International Journal of Pure and Applied Mathematics.

[44] M. Jansson and B. Wahlberg. A linear regression approach to state-space subspace system identification. *Signal Processing*, 52:103–129, 1996.

[45] C. Kamath. On mining scientific datasets. In R. L. Grossman, C. Kamath, P. Kegelmeyer, V. Kumar, and Namburu R. R., editors, *Data Mining For Scientific and Engineering Applications*. Kluwer Academic Publishers, 2001.

[46] B. N. Khoromskij and V. Khoromskaia. Low rank Tucker-type tensor approximation to classical potentials. *Central European Journal of Mathematics*, 5(3):523–550, 2007.

[47] H. A. L. Kiers. Towards a standardized notation and terminology in multiway analysis. *Journal of Chemometrics*, 14:106–125, 2000.

[48] T. G. Kolda. Orthogonal tensor decompositions. *SIAM Journal on Matrix Analysis and Applications*, 23(1):243–255, 2001.

[49] T. G. Kolda. A counterexample to the possibility of an extension of the Eckart-Young low-rank approximation theorem for the orthogonal rank tensor decomposition. *SIAM Journal on Matrix Analysis and Applications*, 24(3):762–767, 2003.

[50] T. G. Kolda. Multilinear operators for higher-order decompositions. Technical Report SAND2006-2081, Sandia National Laboratories, April 2006.

[51] T. G. Kolda and B. W. Bader. Tensor decompositions and applications. Technical Report SAND2007-6702, Sandia National Laboratories, 2007.

[52] T. G. Kolda, B. W. Bader, and J. P. Kenny. Higher-order web link analysis using multilinear algebra. Technical report, Sandia National Laboratories, Albuquerque, New Mexico 87185 and Livermore, California 94550, 2005.

[53] T. Kollo and D. von Rosen. *Advanced Multivariate Statistics with Matrices*. Springer, 2005.

[54] P. M. Kroonenbeg. Principal component analysis of three-mode data by means of alternating least squares algorithms. *Psychometrika*, 45:69–97, 1980.

[55] P. M. Kroonenbeg. *Three–Mode Principal Component Analysis: Analysis and Applications*. PhD thesis, Department of Data Theory, Faculty of Social and Behavioural Sciences, Leiden University, 1983.

[56] A. N. Langville and C. D. Meyer. *Google's PageRank and Beyond: The Science of Search Engine Rankings*. Princeton University Press, July 2006.

[57] L.-H. Lim. Singular values and eigenvalues of tensors: a variational approach. In *Proceedings of the IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing*, volume 1, pages 129–132, 2005.

[58] B. Liu. *Web Data Mining - Exploring Hyperlinks, Contents, and Usage Data*. Springer, 2007.

[59] L. Ljung. *System Identification: Theory for the User*. Prentice-Hall, Upper Saddle River, N.J. 07458, USA, second edition, 1999.

[60] M. Mørup. Analysis of brain data - using multi-way array models on the EEG. Master's thesis, Informatics and Mathematical Modelling, Technical University of Denmark, DTU, Richard Petersens Plads, Building 321, DK-2800 Kgs. Lyngby, 2005. Supervised by Prof. Lars Kai Hansen.

[61] M. Mørup. Non-negative multi-way/tensor factorization NMWF/NTF extended to encorporate sparseness constraints, 2006. Technical report.

[62] M. Mørup, L. K. Hansen, C. S. Hermann, J. Parnas, and S. M. Arnfred. Parallel factor analysis as an exploratory tool for wavelet transformed event-related EEG. *NeuroImage*, 29(3):938–947, feb 2006.

[63] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer Series in Operations Research and Financial Engineering. Springer, New York, second edition, 2006.

[64] L. Omberg, G. H. Golub, and O. Alter. A tensor higher-order singular value decomposition for integrative analysis of dna microarray data from different studies. *Proceedings of the National Academy of Sciences*, 104(47):18371–18376, 2007.

[65] M. J. D. Powell. On search directions for minimization algorithms. *Mathematical Programming*, 4:193–201, 1973.

[66] L. Qi. Eigenvalues of a real supersymmetric tensor. *J. Symb. Comput.*, 40(6):1302–1324, 2005.

[67] L. Qi. Rank and eigenvalues of a supersymmetric tensor, the multivariate homogeneous polynomial and the algebraic hypersurface it defines. *J. Symb. Comput.*, 41(12):1309–1327, 2006.

[68] L. Qi, F. Wang, and Y. Wang. Z-eigenvalue methods for a global polynomial optimization problem. *Mathematical Programming*, 2007.

[69] Y. Rong, S. A. Vorobyov, A. B. Gershman, and N. D. Sidiropoulos. Blind spatial signature estimation via time-varying user power loading and parallel factor analysis. *IEEE Transactions on signal processing*, 53(5), May 2005.

[70] A. Ruhe and P. Å. Wedin. Algorithms for separable nonlinear least squares problems. *SIAM Review*, 22:318–337, 1980.

[71] B. Savas. Analyses and tests of handwritten digit recognition algorithms. Master's thesis, Linköping University, 2003. `http://www.mai.liu.se/~besav/`.

[72] B. Savas. Dimensionality reduction and volume minimization – generalization of the determinant minimization criterion for reduced rank regression problems. *Linear Algebra and its Applications*, 418:201–214, 2006.

[73] B. Savas and L. Eldén. Handwritten digit classification using higher order singular value decomposition. *Pattern Recognition*, 40:993–1003, 2007.

[74] B. Savas and L.-H. Lim. Best multilinear rank approximation of tensors with quasi-Newton methods on Grassmannians. Technical Report LiTH-MAT-R–2008-01–SE, Department of Mathematics, Linkoping University, April 2008.

[75] B. Savas and D. Lindgren. Rank reduction and volume minimization approach to state-space subspace system identification. *Signal processing*, 86:3275–3285, 2006.

[76] N. D. Sidiropoulos and R. Bro. On the uniqueness of multilinear decomposition of n-way arrays. *Journal of Chemometrics*, 14:229–239, 2000.

[77] L. Simonsson. *Subspace Computations via Matrix Decompositions and Geometric Optimization*. Linköping studies in science and technology, Linköping University, 2007. Dissertations No. 1052.

[78] A. Smilde, R. Bro, and P. Geladi. *Multi-way Analysis: Applications in the Chemical Sciences*. Wiley, 2004.

[79] K. Stockinger. *Multi-Dimensional Bitmap Indices for Optimising Data Access within Object Oriented Databases at CERN*. PhD thesis, University of Vienna, Austria, 2001.

[80] P. Stoica and M. Viberg. Maximum likelihood parameter and rank estimation in reduced-rank multivariate linear regression. *IEEE Transactions on signal processing*, 44(12):3069–3078, 1996.

[81] G. Tomasi. *Practical and Computational Aspects in Chemometric Data Analysis*. PhD thesis, The Royal Veterinary and Agricultural University, Denmark, 2006.

[82] L. R. Tucker. The extension of factor analysis to three-dimensional matrices. In H. Gulliksen and N. Frederiksen, editors, *Contributions to Mathematical Psychology*, pages 109–127. Holt, Rinehart and Winston, New York, 1964.

[83] L. R. Tucker. Some mathematical notes on three-mode factor analysis. *Psychometrika*, 31:279–311, 1966.

[84] M. A. O. Vasilescu and D. Terzopoulos. Multilinear analysis of image ensembles: Tensorfaces. In *Proc. 7th European Conference on Computer Vision (ECCV'02)*, Lecture Notes in Computer Science, Vol. 2350, pages 447–460, Copenhagen, Denmark, 2002. Springer Verlag.

[85] M. A. O. Vasilescu and D. Terzopoulos. Multilinear subspace analysis of image ensembles. In *Proc. Computer Vision and Pattern Recognition Conf. (CVPR'03)*, volume 2, pages 93–99, Madison WI, 2003.

[86] M. A. O. Vasilescu and D. Terzopoulos. Multilinear independent component analysis. In *Proc. Computer Vision and Pattern Recognition Conf.*, 2005.

[87] M. Viberg. On subspace-based methods for the identification of linear time-invariant systems. *Automatica*, 31(12):1835–1852, 1995.

[88] S. A. Vorobyov, Y. Rong, N. D. Sidiropoulos, and A. B. Gershman. Robust iterative fitting of multilinear models. *IEEE Transactions on signal processing*, 53(8), August 2005.

[89] T. Zhang and G. H. Golub. Rank-one approximation to high order tensors. *SIAM Journal on Matrix Analysis and Applications*, 23(2):534–550, 2001.