Methods for large volume image analysis Applied to Early detection of Alzheimer's disease by analysis of FDG-PET scans

Andreas Kodewitz

UNIVERSITÉ D'EVRY - VAL D'ÉSSONNE

Sciences et Ingénierie ED 511

Laboratoire IBISC - TADIB

NNT : EVRY

THÈSE

présentée et soutenue publiquement le **18 mars 2013**

pour l'obtention du grade de

Docteur de l'Université d'Évry - Val d'Éssonne

Spécialité : Mathematiques Appliquées

par :

Andreas KODEWITZ

Titre : Methods for large volume image analysis
Applied to
Early detection of Alzheimer's disease by analysis of FDG-PET scans

COMPOSITION DU JURY

Rapporteur :	Pierre COMON	DR CNRS, GIPSA-Lab, France
Rapporteur :	Klaus OBERMAYER	Prof. Dr. rer. nat. Université TU Berlin, Allemagne
Examinateur :	Elmar W. LANG	Prof. Dr. rer. nat. Université de Regensburg, Allemagne
Examinateur :	Christophe MONTAGNE	MCF, Université d'Évry, France
Examinateur :	Nicole VINCENT	PR, Université Paris 5, France
Directeur de thèse :	Sylvie LELANDAIS	PR, Université d'Évry, France
Directeur de thèse :	Vincent VIGNERON	HDR, Université d'Évry, France

Résumé

Dans cette thèse, nous explorons de nouvelles méthodes d'analyse d'images pour la détection précoce des changements métaboliques cérébraux causés par la maladie d'Alzheimer (MA). Nous introduisons deux apports méthodologiques que nous appliquons à un ensemble de données réelles.

Le premier est basé sur l'apprentissage automatique pour créer une carte des informations de classification pertinente dans un ensemble d'images. Pour cela nous échantillonnons des blocs de voxels de l'image selon un algorithme de Monte-Carlo. La mise en œuvre d'une classification basée sur ces patchs 3D a pour conséquence importante la réduction significative du volume de patchs à traiter, et l'extraction de caractéristiques dont l'importance est statistiquement quantifiable. Cette méthode s'applique à différentes caractéristiques de l'image et donc est adaptée à des types d'images très variés. La résolution des cartes produites par cette méthode peut être affinée à volonté et leur contenu informatif est cohérent avec les résultats antérieurs basés sur les statistiques sur les voxels obtenus dans la littérature.

Le second apport méthodologique porte sur la conception d'un nouvel algorithme de décomposition de tenseur d'ordre important, adapté à notre application. Cet algorithme permet de réduire considérablement la consommation de mémoire et donc évite la surcharge de la mémoire. Il autorise la décomposition rapide de tenseurs, y compris ceux de dimensions très déséquilibrées. Nous appliquons cet algorithme en tant que méthode d'extraction de caractéristiques dans une situation où le clinicien doit diagnostiquer des stades MA précoce ou MCI (Mild Cognitive Impairment) en utilisant la TEP FDG seule. Les taux de classification obtenus sont souvent au-dessus des niveaux de l'état de l'art.

Dans le cadre de ces tâches d'analyse d'images, nous présentons notre source de données, les scans de patients retenus et les pré-traitements réalisés. Les principaux aspects que nous voulons prendre en compte sont la nature volumétrique des données, l'information a priori disponible sur la localisation des changements métaboliques et comment l'identification des zones de changements métaboliques participe à la réduction de la quantité de données à analyser et d'extraire des caractéristiques discriminantes.

Les méthodes présentées fournissent des informations précises sur la localisation de ces changements métaboliques. Les taux de classification allant jusqu'à 92,6% pour MA et 83,8% pour MCI. En outre, nous sommes capables de séparer les patients MCI stables des MCI patients évoluant vers la MA dans les 2 ans après l'acquisition du PET-scan avec un taux de classification de 84.7%. Ce sont des étapes importantes vers une détection fiable et précoce de la MA.

Abstract

In this thesis we want to explore novel image analysis methods for the early detection of metabolic changes in the human brain caused by Alzheimer's disease (AD). We will present two methodological contributions and present their application to a real life data set.

We present a machine learning based method to create a map of local distribution of classification relevant information in an image set. The presented method can be applied using different image characteristics which makes it possible to adapt the method to many kinds of images. The maps generated by this method are very localized and fully consistent with prior findings based on voxel wise statistics.

Further we preset an algorithm to draw a sample of patches according to a distribution presented by means of a map. Implementing a patch based classification procedure using the presented algorithm for data reduction we were able to significantly reduce the amount of patches that has to be analyzed in order to obtain good classification results.

We present a novel non-negative tensor factorization (NTF) algorithm for the decomposition of large higher order tensors. This algorithm considerably reduces memory consumption and avoids memory overhead. This allows the fast decomposition even of tensors with very unbalanced dimensions. We apply this algorithm as feature extraction method in a computer-aided diagnosis (CAD) scheme, designed to recognize early-stage AD and mild cognitive impairment (MCI) using fluorodeoxyglucose (FDG) positron emission tomography (PET) scans only. We achieve state of the art classification rates.

In the context of these image analysis tasks we present our data source, scan selection and preprocessing. The key aspects we want to consider are the volumetric nature of the data, prior information available about the localization of metabolic changes, discovering the localization of metabolic changes from the data, using this information to reduce the amount of data to be analyzed and discovering discriminant features from the data.

The presented methods provide precise information about the localization of metabolic changes and classification rates of up to 92.6 % for early AD and 83.8 % for MCI. Furthermore, we are capable to separate stable MCI patients from MCI patients declining to AD within 2 years after acquisition of the PET scan with a classification rate of 84.7 %. These are important steps toward a reliable early detection of AD.

Acknowledgments

At this occasion I want to thank all people which assisted and supported me during the last three years which only made this thesis possible. A special thanks goes to

- Sylvie Lelandais and Vincent Vigneron, my supervisors, for giving me the opportunity to do my PhD in france, all the interesting discussions and providing the guidance I needed to accomplish this research. I specially want to thank Vincent for providing every imaginable help when I was trapped in bureaucracy after arriving in France and throughout the whole last 3 years.
- Prof. Dr. Elmar Lang for preparing me for the challenge of writing a PhD thesis and being member of the PhD jury.
- Pierre Comon and Prof. Dr. Klaus Obermayer for accepting to examine my thesis and devoting time to write their reports.
- Tahir Syed and Christophe Montagne for welcoming me in Evry and for helping to find flaws in publications and presentations.
- Florent Pericot for keeping my calculations running and accessible wherever and whenever I wanted.

Last but not least I want to thank my family and my friends for their constant support. More than anybody else I have to thank Katia for here support and love.

Andreas Kodewitz

Acronyms

3D-SSP 3D stereotactic surface projection.

Aβ amyloid β.

AC-PC Anterior Commissure-Posterior Commissure (AC-PC) line. *Glossary:* AC-PC.

AD Alzheimer's disease.

ADNI Alzheimer's Disease Neuroimaging Initiative.

ADRDA Alzheimer's Disease and Related Disorders Association.

AIC Akaike information criterion.

AIR automated image registration.

ALS alternating least squares.

ANCOVA analysis of co-variance.

ANLS alternating non-negative least squares.

ANOVA analysis of variance.

BPP block principal pivoting.

BSS blind source separation.

CAD computer-aided diagnosis.

CANDECOMP canonical decomposition.

CART classification and regression tree.

CBCL center for biological and computational learning.

CDR Clinical Dementia Rating (CDR). *Glossary:* CDR.

cgP preconditioned nonlinear conjugate gradient.

CP canonical polyadic. *see also:* CANDECOMP/PARAFAC.

CSF cerebrospinal fluid.

CT computed tomography.

DCT discrete cosine transform.

DLB dementia with Lewy bodies.

EEG electroencephalography.

FBP filtered backprojection.

FDA Fisher discriminant analysis.

FDG fluorodeoxyglucose. *Glossary:* FDG.

fMRI functional magnetic resonance imaging.

FNR false negative rate.

FPR false positive rate.

FTD fronto-temporal dementia.

FWHM full width at half maximum.

GM gray matter.

GUI graphical user interface.

HALS hierarchical alternating least squares.

HONMF higher order non-negative matrix factorization.

ICA independent component analysis.

ICBM International Consortium for Brain Mapping.

MCI mild cognitive impairment, *Glossary:* MCI.

$\mathsf{MCI}_{\mathsf{AD}}$ mild cognitive impairement converting to Alzheimer's disease.

MMSE Mini-mental State Exam. *Glossary:* MMSE.

MNI Montreal Neurological Institute. *Glossary:* MNI.

MRI magnetic resonance imaging.

NsTsF non-negative sub-tensor set factorization.

NC normal control group.

NFT intracellular neurofibrillary tangles.

NIH National Institutes of Health.

NINCDS National Institute of Neurological and Communicative Disorders and Stroke.

NMF non-negative matrix factorization.

NNLS non-negativity constrained least squares.

NTD non-negative Tucker decomposition.

NTF non-negative tensor factorization.

ooB out-of-Bag.

OS-EM ordered subset expectation maximization.

PARAFAC parallel factor analysis.

PCA principal component analysis.

PDD Parkinson's disease with dementia.

PET positron emission tomography.

PIB Pittsburgh compound B. *Glossary:* PIB.

RAM random access memory.

RBF radial basis function.

RF random forest.

ROI region of interest.

SNIPE scoring by non-local image patch estimator.

SNR signal to noise ratio.

SPECT single photon emission computed tomography.

SPM statistical parametric mapping.

SRT selective reminding test.

SSM scaled subprofile model.

SVD signular value decomposition.

SVM support vector machine.

VaD vascular dementia.

VOI volume of interest.

WM white matter.

Notation

$\boldsymbol{a}$	Vector
$\boldsymbol{A}$	Matrix
$\boldsymbol{\mathcal{A}}$	Tensor
$\otimes$	Kronecker product
$\circ$	outer product
$\bullet$	Hadamard or element wise product
$\frac{\boldsymbol{A}}{\boldsymbol{B}}$	Element wise division of two matrices
$\boldsymbol{\mathcal{A}} \times_n \boldsymbol{B}$	Mode-n tensor matrix multiplication
$\boldsymbol{A}_{(n)}$	Mode-n unfolding
$[\![\cdot]\!]$	Tucker operator
$\text{rank}(\cdot)$	Rank of a matrix or tensor
$\text{Tr}(\cdot)$	Trace of a matrix
$\mathbb{I}$	Identity matrix
$\boldsymbol{\mathcal{I}}$	Diagonal one tensor
$\mathbb{1}$	Matrix with all entries equal one
$\|\cdot\|_1$	L_1 norm
$\|\cdot\|_2$	L_2 norm
$\|\cdot\|_F$	Frobenius norm
$\|\cdot\|$	Euclidean norm
$\|\cdot\|$	Absolute value
$\mathbb{N}$	Natural numbers
$\mathbb{R}$	Rational numbers
$\mathbb{R}_{0+}$	Non-negative rational numbers
$\mathbb{C}$	Complex numbers
$I(j,k,l)$	A 3 dimensional image
$I(\cdot)$	The vectorization of an image
$\xi(\cdot), \xi(\cdot,\cdot,\cdot)$	Importance vector, importance map
mod	The modulo operator
$\div$	Euclidean division (division of integers)
sgn	signum of
M	Number of examples/images
Q	Number of Variables
G	Number of patches
T	Number of trees

Contents

1. Introduction

Medical and biological imaging can be considered as one of the most important fields of scientific imaging. With its advances over the last decades and its computerization a large field of research emerged which confronts computer scientists whith a large range of interesting and challenging problems. Especially 3D imaging profits from the advances in computerized image reconstruction, image processing, image analysis and visualization, as well as the availability of large data storage. Due to these advances 3D imaging has become an indispensable utility in diagnosis and treatment of cancer, surgery planing and accomplishment, and is furthermore used in various fields of medical research.

In many cases the visual analysis of medical images is a time consuming task that demands a highly trained physician. With the increasing lack of physicians and health systems on short finance all over the world, the time of medical doctors becomes more and more valuable. To uphold the quality of our medical services we need innovations that enhance the efficiency of medical procedures and boost early diagnosis. This not only enhances the patients prospects but can also be of advantage for cost effectiveness. The efficiency of medical doctors can profit in several ways from computer assistance in the analysis of medical images. Automated preprocessing can enhance contrast and sharpness. Image processing can facilitate the interpretation of the acquired image. Computer assistance can introduce quantifiable measures in the decision process, which makes results more reproducible. At present computer-aided diagnosis (CAD) systems for several imaging modalities and medical conditions exist that are capable to highlight suspicious regions and hence facilitate diagnosis [Doi, 2007]. According to Doi [2007] the accuracy of the CAD systems does not even have to be comparable or better than the physicians in order to be able to improve the overall performance. It is only necessary for the CAD system to be complementary to the physicians analysis insofar as it has to provide the physician information that would not be easily available without the supporting CAD system. The accuracy of the CAD system however influences heavily the positive impact on the overall diagnostic performance.

While positron emission tomography (PET) imaging plays a very important role in the early diagnosis of cancer and the detection of metastasization, its use in the diagnosis of neurodegenerative diseases like dementia is, even though subject of a vast amount of recent publications, in a stage that demands further advances of imaging and analysis methodology to become a routine tool for the clinician. The main limit-

ing factors for the application of PET in the diagnosis of neurodegenerative diseases, besides the high cost, are the high level of noise in the image, poor resolution in comparison to magnetic resonance imaging (MRI) and the resulting time consumption of visual evaluation of the scans. The quality of visual assessment of PET scans depends heavily on the experience and training of the medical examiner. Clear rules and quantifiable measures for the diagnosis of dementia are still missing. CAD approaches might help to facilitate the application of PET in this clinical context and therefore improve early detection and appropriate treatment of neurodegenerative diseases. The early detection is of high interest as a large amount of costly nursing cases is caused by neurodegenerative diseases. Ishii [2002] describes the use of two voxel based statistical methods, statistical parametric mapping (SPM) and 3D stereotactic surface projection (3D-SSP)/NEUROSTAT, that provide the clinician an additional representation of the PET scan which highlights areas of hypo- and hyper-metabolism compared to the average. These representations facilitate the detection of patterns specific to different types of neurodegenerative diseases and can therefore be considered a CAD system.

Most published findings concerning neurodegenerative diseases are based on voxel-wise methods (see section 4.2). The shortcoming of voxel wise statistics is that every pixel in the image is analyzed individually. The clinical specialist identifies neurodegenerative diseases in PET scans by searching for a disease specific *pattern* of hypo-metabolism in the brain. This suggests that the decision relevant information is comprised in patterns and not only in single voxels. Therefore, it can be expected that analyzing the images on a larger scale than voxels with methods aiming at the detection of *hypo-metabolism patterns* can outperform voxel-wise analysis. Nevertheless, voxel-wise statistics are valuable for the clinician as this representation visualizes information that would be not accessible without computers' aid.

Classically patterns are analyzed by texture analysis and/or segmentation. CAD system for the detection of cancer metastasization using PET imaging, for example, include or are based on segmentation of the suspected region and/or texture analysis [Guan et al., 2006, Opfer et al., 2009]. CAD systems for mammography screening often use the same techniques. These two methods, however, are unequally harder to apply in the detection of metabolic changes in the brain. Images comprise in this case less contrast and a higher level of noise which makes it even for the trained human reader hard to clearly identify abnormalities in the level of metabolism. Without a segmentation of a regions of interest (ROIs) we have the choice of selecting anatomically predefined ROIs which leads to the risk of excluding important information from the analysis or alternatively selecting the whole brain scan which makes it necessary to deal with a large amount of data.

1.1. Problem statement

In the preparation of this thesis we have selected the most common cause of dementia, Alzheimer's disease (AD), to represent the whole group of dementias as practical example. Clinical diagnosis of AD is currently primarily based on neuro-psychological testing. But medical literature agrees that metabolic changes in the brain, that can be visualized by PET scans, occur before symptoms can be recognized. Therefore, PET scans are considered a promising means for the early detection of AD. The analysis of this type of image, however, is challenging due to a high level of noise, low resolution and differences between healthy and demented being very subtle.

Selecting AD we are in the fortunate position to have a collection of approx. 400 PET scans from a study in the US available. This study also includes patients suffering from mild cognitive impairment (MCI), which is characterized by memory problems beyond normal aging but not as considerable to qualify for a diagnosis of AD. These patients are however more likely to develop AD in the future and therefore of high interest in the development of methods for the early detection of AD. With adequate selection and preparation of the images this database provides a solid basis for the development and evaluation of novel analysis and early detection methods.

The aim of this thesis is to suggest new approaches for the analysis of PET brain scans sensible to metabolic changes in the early stage and the MCI stage of AD. Class differences in PET images between normal control group (NC) and AD patients being already established [Drzezga et al., 2003, Scarmeas et al., 2004], our aim is single patient classification for early diagnosis. As previous publications mostly approached the problem by an analysis on the voxel level, we want to explore the possibilities to base the analysis on larger areas of the scan. We will follow two approaches: *1)* image patches, i.e. small regular regions of the image, as base for the analysis and *2)* transformation of the images' information to an lower dimensional basis by decomposition of the image data.

Working on the scale of image patches we want to examine how available information about the underlying condition can be used to effectively select the patches to be analyzed and how to obtain such information directly from the available image data. The primary goal is a data reduction.

In order to obtain an adequate decomposition of the image data we want to develop an adapted decomposition algorithm respecting the images' properties *a)* non-negativity and *b)* 3 dimensionallity. Therefore, we will develop a non-negative tensor decomposition algorithm which is capable to decompose large data arrays with unbalanced dimensions. We will especially attempt to avoid memory problems encountered with standard state of the art algorithms. This algorithm can then be used to trans-

form the image data into a lower dimensional space.

Our objectives are to discover discriminant features from the image data, reduce the dimensionallity of the problem and recover information about the location of disease related changes.

1.2. Contributions

In this thesis we will present novel approaches to the detection of metabolic changes for the early detection of AD. In Part I we will develop methods separated from its application as all methods can, with some minor modifications if at all necessary, as well be applied to other diseases and imaging modalities. In Part II we will present the application of the developed methods to clinical data, medical background and the preparation of the data set.

Importance mapping Dealing with a classification of a set of images re-partitioned into classes it is interesting to learn where in the image the classification relevant information is situated. We present a procedure to create a map of the distribution of important information in the image dataset. This procedure is based on the random forest (RF) classifier. We demonstrate its application using a set of face images. In the second part of this text the presented method is used to create a map of class-differences between NC and AD which is compared to a map of inter-class variance (chapter 2).

NTF algorithm This thesis also proposes a novel non-negative tensor factorization (NTF) algorithm that relaxes size restrictions encountered with state of the art algorithms. It is especially adapted to the case where a large number of examples is available and in consequence the input tensor has very imbalanced dimensions. Its properties are analyzed using a standard example for this kind of algorithm and its performance is compared to established algorithms for matrix and tensor decomposition. The algorithm is finally applied as feature extraction means in the second part of this text and allows to separate stable MCI patients from MCI to AD converters with an accuracy of $\sim 85\,\%$ (chapter 3).

PET scan data set A PET data set meeting the requirements for the development and the analysis of computer based AD early-detection methods was created. The image source and preprocessing are presented. A thorough analysis and revision of inclusion/exclusion criteria is performed. (chapter 5)

Spatial importance sampling Furthermore, we present a technique to draw a sample of patches based on a predefined distribution presented in form of a map. This *spatial* importance sampling procedure was created to include prior knowledge about the localization of task relevant information for data reduction in patch wise image analysis (chapter 8).

Alzheimer's disease classification We present a classification setup based on the means voxel intensity of patches. With 1000 patches selected by thresholding our importance map and bootstrap cross-validation we achieved classification rates of up to 95.7 %. (chapter 7)

We also present a NTF decomposition based setup that achieves a 85 % classification rate for stable mild cognitive impairment patients vs. mild cognitive impairment patients that developed Alzheimer's disease within a 2 years follow-up after the date of acquisition of the PET scan. In the classification of NC vs. early-stage AD the same setup achieves classification rates of $\sim 93\,\%$. (chapter 9)

Research presented in this thesis is object of the following publications:

- Three conference papers: "Exploratory Matrix Factorization for PET Image Analysis", Conf Proc IEEE Eng Med Biol Soc. (EMBS), 2010. "Where to search for Alzheimer's disease related changes in PET scans?", Recherche en imagerie et technologie pour la santé (RITS), 2011. "3D Local Binary Pattern for PET image classification by SVM, Application to early Alzheimer disease diagnosis", Proceedings of the 6th International Conference on Bio-Inspired Systems and Signal Processing (BIOSIGNALS 2013).

- One oral presentation: "Non-negative sub-tensor set Factorization", GdR-ISIS: Décompositions tensorielles et applications, Wed 16$^{\text{th}}$ Jan, 2013.

- Two journal papers: "Alzheimer's disease early detection from sparse data using brain importance maps", Electronic Letters on Computer Vision and Image Analysis (in revision). "A sub-tensor based NTF algorithm" (in preparation).

1.3. Organization of the thesis

We have separated the text into two parts. Part I, Methodology, which concerns the development of novel analysis methods; and Part II, Application, which treats the application of the developed methods to the early detection of AD, as well as medical background and data preparation.

In chapter 2 we present a novel approach for the extraction of a map imaging the importance of the different parts of an image for a classification. We begin this chapter

with a reminder about the classifiers we will use in the remainder of the text. Section 2.1 gives a short introduction to support vector machines (SVMs) and RFs. We go into more detail about the possibility to measure the feature importance with the RF (section 2.2) as this will use this feature of the RF in the extraction of our importance maps. We than expose the actual method for the extraction of the importance map in section 2.3 and demonstrate the application with an explicative example in section 2.4.

In the last chapter of the first part (chapter 3) we develop a novel NTF algorithm specially designed for the decomposition of large higher order tensors. Section 3.1 introduces the used notations, notably the tensor algebra and some special types of tensors. Section 3.2 introduces tensor decomposition models, such as the CP model, the Tucker model and the N-dimensional Tucker model, and discusses their advantages over models in the matrix domain. In section 3.4 we develop our NTF algorithm for large tensors, analyze its properties and compare its performance with state of the art algorithms.

The second part of the thesis starts with an introduction to the problem of AD early detection. Section 4.1 of the 3rd chapter provides medical background knowledge, including the clinical diagnosis of AD and the use of PET imaging in the diagnosis. In section 4.2 we give an overview over recent publications dealing with the computerized analysis of AD and finish the chapter presenting the most important software tools we have used in the preparation of this work. We notably introduce brain coordinate systems and spatial normalization tools in section 4.3.

Chapter 5 presents the preparation of the used data set. The data source and image acquisition procedures are presented in section 5.1. In section 5.2 we specify which properties the data set should have to fit best the needs in the development of computerized analysis methods. Section 5.3 presents an existing image set and its shortcomings, that led to the creation of a new image set, are discussed. The preparation of this data set is described in section 5.4, we especially discuss the revision of meta data and patient selection, as well as the setup of spatial normalization and all further preprocessing steps. The chapter closes with section 5.5, giving a better impression of the data set by displaying some statistic properties.

The following three chapters present possible applications of the tools that have been developed in the first part of the thesis in the early detection of AD.

In chapter 6 we extract a map that localizes the brain areas important for the detection of AD using the PET image data only. Section 6.1 explains all necessary steps to apply the procedure developed in chapter 2 and create a map for the distribution of important information in the brain images. The created maps are analyzed in section 6.2 and its properties discussed in section 6.3.

Chapter 7 presents a first application of the importance map we have extracted from the image data in the previous chapter. Section 7.1 describes how we threshold the map to reduce the data to be analyzed. The results are presented in section 7.2 and discussed in the following section.

In chapter 8 we sample patches from the brain in order to reduce the amount of data. In section 8.1 we present the algorithm for sampling points or patches in an image given a distribution stored in a map. We also describe the prior probability calculus necessary to use the algorithm to sample patches with a fixed ratio in two ROIs. Experiment procedures and results of a sampling with predefined ROIs are presented in section 8.2. Section 8.3 describes experimental setup and results of sampling with the learned importance map. A discussion of the results closes the chapter.

In chapter 9 we propose a possible use of NTF in the early diagnosis of AD. In section 9.1 we start by explaining how we use our non-negative sub-tensor set factorization (NsTsF) algorithm to obtain a new basis which is used to represent the data in an lower dimensional space, which information we use as input to the classifiers. Furthermore, we describe the two classification experiments performed and present their results. The chapter ends with a discussion of the obtained results.

Chapter 10 forms the conclusion of this thesis and is only followed by the appendix containing the definitions of some matrix operations and supplementary material that would have disturbed the reading flow.

Part I.

Methodology

In the first part of this text we want to present the development of our novel analysis methods separated from the intended application. This separation has two reasons: *1)* The methods developed are not application specific but can be applied to many similar problems. *2)* Our intended application, the early detection of Alzheimer's disease (AD), requires a certain background knowledge about brain anatomy and the disease itself and we want to spear readers interested in applying our methods to other problems to familiarize with this specific details.

We developed novel analysis methods for an application to volumetric images, such as computed tomography (CT), magnetic resonance imaging (MRI) and positron emission tomography (PET), but they can as well be applied to photographic or microscopic images. Actually we will give at the end of each chapter in this section a small demonstration of the presented method on a set of face images. The developed methods are of special interest when a large amount of image data has to be analyzed.

In chapter 2 we present a machine learning technique that extracts, based on features calculated on image patches, information about the location of class discrimination information. This information, stored in form of a map, can be used to constrain a computationally more expensive analysis to a smaller area, reduce the feature set, and thus reduce calculation time.

In chapter 3 we present a novel non-negative tensor factorization (NTF) algorithm. This algorithm is specialized to the decomposition of large scale tensor. Its special properties allow the decomposition of tensors with extremely imbalanced dimensions and they even allow to start the decomposition *before* the entries of the whole tensor are known. This algorithm allows the decomposition of tensors that are to large to be stored in the computer's RAM without generating a large memory overhead.

2. Learning importance maps from image data

In image classification often the question arises which areas of the image are the most important for the classification task. This information is especially useful if the analysis of the whole image is computationally too costly or in case we desire to determine a new, better feature. We will assume to deal with a set of cropped and aligned images, like for example in functional brain imaging or face recognition, partitioned into C classes. These classes could be different persons, healthy and diseased or different objects shown in the image. To obtain information about the location of important areas in the image we could for example calculate voxel-wise inter-class variance

$$\sigma^2_{\text{inter}}(j,k,\ell) = \sum_{c=1}^{C} \frac{M_c}{M}(\mu_c(j,k,\ell) - \mu(j,k,\ell))^2 \tag{2.1}$$

where M is the number of images, M_c the number of images in class c, $\mu(j,k,\ell)$ the image of voxel wise average intensity

$$\mu(j,k,\ell) = \frac{1}{M}\sum_{m=1}^{M} I_m(j,k,\ell) \tag{2.2}$$

with $I_m(j,k,\ell)$ the intensity of voxel (j,k,ℓ) in image m and μ_c the average voxel intensity of class c. This variance shows class differences based on intensity differences at the voxel level. In many cases however these differences might not be very specific to the intended classification.

In the following we will develop a procedure to extract a map of classification relevant areas in the image. This procedure is based on a patch wise feature extraction and supervised classification. To do so we need the class labels for the images. For the application of this procedure we have to assume that the images have been cropped and aligned in order to find similar scenes in all images. This method provides more flexibility than the inter-class variance as it gives us the liberty of choice for the feature and the patch size. Furthermore it does not treat the features independently but includes also the relations between the single features.

The extraction procedure is based on the random forest (RF) classifier which is introduced in the following section. We chose this classifier because of its high predictive power, its capability to manage large amounts of features and especially because it allows to retrieve easily a feature importance measure.

In the following we will explain the principles of support vector machine (SVM) and random forest (RF) classifiers. Readers familiar with these techniques can go on directly to section 2.2, which details the measurement of feature importance using RFs. The last sections expose the map generation process and a simple application example of this method.

2.1. Reminder on classification

At the beginning of this chapter we want to refresh our memory on classification techniques. After some general thoughts about classification we will provide a short introduction to the classifiers used in the remainder of this thesis: SVMs and the RF classifier.

Definition 2.1 (classification dataset)
A classification dataset $\mathcal{B}_0$ of n observations consists of tuples of explanatory variables $\boldsymbol{x}_i$ and class labels $y_i \in \{c_1, \ldots, c_C\}$

$$\mathcal{B}_0 = \{(\boldsymbol{x}_i; y_i)\}_{i=1}^{M}. \tag{2.3}$$

Here $\boldsymbol{x}_i$ is the vector of explanatory variables and y_i is the class label related to the i-th example. The number of different values assumed by y_i is called the number of classes C.

The classification task consists in finding a mapping Ψ from the explanatory space $\mathcal{X}$ to the class labels y_i.

$$\begin{aligned} \Psi : &\mathcal{X} \to \{c_1, \ldots, c_C\} \\ &\boldsymbol{x}_i \mapsto y_i \end{aligned} \tag{2.4}$$

If the number of classes $C = 2$ we are dealing with *binary* classification. Many classification techniques have been developed first for binary classification and later extended to multi-class classification.

2.1.1. The Support Vector Machine (SVM)

SVMs, first developed by Boser, Guyon, and Vapnik [1992], solve the classification problem geometrically. It is designed to construct a hyperplane in the feature space that separates the classes with maximum margin. The hyperplane is defined by the so-called *support vectors*, which give the method its name. Once this hyperplane is constructed new data points can be classified by testing whether a data point lies on one or the other side of the hyperplane. First let us define a hyperplane.

Definition 2.2 (Hyperplane)
Let $\mathcal{H}$ be a dot product space and $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_M \in \mathcal{H}$ a set of feature vectors. Then any hyperplane in $\mathcal{H}$ can be written as

$$\{\boldsymbol{x} \in \mathcal{H} | \langle \boldsymbol{w}, \boldsymbol{x} \rangle + b = 0\}, \quad \boldsymbol{w} \in \mathcal{H}, b \in \mathbb{R} \tag{2.5}$$

where $\boldsymbol{w}$ is a vector orthogonal to the hyperplane.

This definition of the hyperplane contains a superfluous freedom of scale. $\boldsymbol{w}$ and b can be multiplied by the same non-zero constant without causing any change. This freedom is eliminated in the definition of the canonical hyperplane.

Definition 2.3 (Canonical hyperplane)
The pair $(\boldsymbol{w}, b) \in \mathcal{H} \times \mathbb{R}$ is called a canonical form of the hyperplane (Equation 2.5) with respect to $\boldsymbol{x_1}, \ldots, \boldsymbol{x_M} \in \mathcal{H}$, if it is scaled such that

$$\min_{i=1,\ldots,M} |\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b| = 1. \tag{2.6}$$

The definition of the canonical hyperplane still leaves us with two differently orientated hyperplanes. In the search for a decision function

$$\begin{aligned} f_{\boldsymbol{w},b} : &\mathcal{H} \to \{\pm 1\} \\ &\boldsymbol{x} \mapsto f_{\boldsymbol{w},b}(x) = \operatorname{sgn}(\langle \boldsymbol{w}, \boldsymbol{x} \rangle + b) \end{aligned} \tag{2.7}$$

these two hyperplanes however are distinguished by the class labels $y_i \in \{\pm 1\}$ that are assigned to the data points. In pattern recognition we want $f_{\boldsymbol{w},b}(\boldsymbol{x}_i) = y_i$ for as many i as possible, we want the data to be classified correctly.

In Figure 2.1 we have plotted a linear separable data set for binary classification. We have plotted one hyperplane that separates the data points perfectly, but it is obvious that there is an infinite number of other hyperplanes that also separates the data points perfectly. The hyperplane plotted in the figure however maximizes the margin, the distance of the data points to the hyperplane, and therefore maximizes the prospect that new data points are also correctly classified.

Searching the hyperplane with maximum margin leads to the optimization problem:

$$\min_{\boldsymbol{w} \in \mathcal{H}, b \in \mathbb{R}} \frac{1}{2} \|\boldsymbol{w}\|^2, \quad \text{subject to } y_i(\langle \boldsymbol{x_i}, \boldsymbol{w} \rangle + b) \geq 1 \quad \forall i = 1, \ldots, M \tag{2.8}$$

It is more convenient to approach this optimization problem using the "dual problem", which we will derive in the following. First we introduce the Lagrangian

$$L(\boldsymbol{w}, b, \boldsymbol{\alpha}) = \frac{1}{2} \|\boldsymbol{w}\|^2 - \sum_{i=1}^{M} \alpha_i \left(y_i(\langle \boldsymbol{x_i}, \boldsymbol{w} \rangle + b) - 1 \right), \tag{2.9}$$

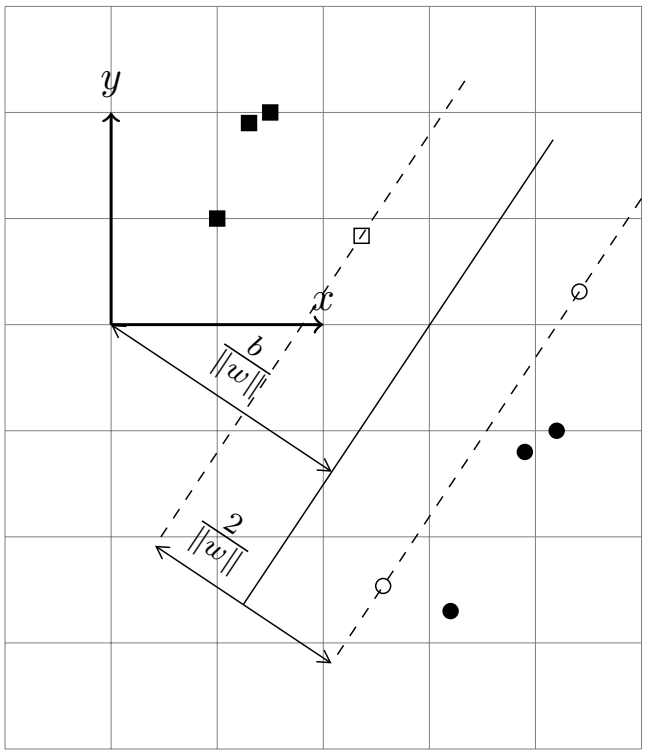


Figure 2.1.: Linear separable SVM example in 2 dimensions for binary classification with class separating hyperplane and margins. The support vectors are plotted not filled while normal data points are filled. The hyperplanes constructing the margin $\langle \boldsymbol{w}, \boldsymbol{x} \rangle + b = \pm 1$ are drawn as dashed lines.

with Lagrange multipliers $\alpha_i \geq 0$. The Lagrangian has to be maximized with respect to α_i and minimized with respect to $\boldsymbol{w}$. Consequently, the derivatives of the Lagrangian L must vanish at this saddle-point.

$$\frac{\partial}{\partial b} L(\boldsymbol{w}, b, \boldsymbol{\alpha}) = 0, \qquad \frac{\partial}{\partial \boldsymbol{w}} L(\boldsymbol{w}, b, \boldsymbol{\alpha}) = 0 \tag{2.10}$$

which leads to

$$\sum_{i=1}^{M} \alpha_i y_i = 0 \tag{2.11}$$

$$\boldsymbol{w} = \sum_{i=1}^{M} \alpha_i y_i \boldsymbol{x}_i \tag{2.12}$$

Substituting Equation 2.11 and Equation 2.12 into the Lagrangian (Equation 2.9), we derive the dual form of the optimization problem:

$$\begin{aligned} \max_{\boldsymbol{\alpha}\in\mathbb{R}^M} W(\boldsymbol{\alpha}) &= \sum_{i=1}^{M} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{M} \alpha_i \alpha_j y_i y_j \left\langle \boldsymbol{x_i} \boldsymbol{x_j} \right\rangle \\ \text{subject to } \alpha_i \geq 0, i = 1, \ldots, M \text{ and } & \sum_{i=1}^{M} \alpha_i y_i = 0 \end{aligned} \tag{2.13}$$

This is a convex optimization problem. Therefore, it is possible to find the global optimum. The decision function (Equation 2.7) can now be rewritten by substituting Equation 2.12 as

$$f(\boldsymbol{x}) = \operatorname{sgn}\left(\sum_{i=1}^{M} \alpha_i y_i \left\langle \boldsymbol{x}, \boldsymbol{x}_i \right\rangle + b \right). \tag{2.14}$$

The linear separable case, thus didactically important, is not quite frequent when dealing with real life classification problems. The most commonly used solution to make it possible to classify data that is not linearly separable by a SVM is the use of a kernel.

Definition 2.4 (Kernel)
Let the mapping Φ be a transformation of the data from the input space $x \in X$ to a higher dimensional feature space H:

$$\begin{aligned} \Phi : & X \to H \\ & x \mapsto \Phi(x). \end{aligned} \tag{2.15}$$

The kernel function $k(x, x')$ calculates the dot product of the mapping Φ

$$k(x, x') = \left\langle \Phi(x), \Phi(x') \right\rangle \tag{2.16}$$

As the dual optimization problem Equation 2.13 consists only of dot products we can apply the so called "kernel trick". By replacing the dot product with the kernel function $k(x, x')$ the data is implicitly mapped to a higher dimensional space. In the higher dimensional space the data becomes linearly separable. The actual mapping Φ has not to be known since only its dot product is used.

Commonly used kernel functions are *polynomial* kernel of degree d

$$k(x, x') = \left\langle x, x' \right\rangle^d \tag{2.17}$$

radial basis function (RBF) or *Gaussian* kernel of width $\sigma > 0$

$$k(x, x') = \exp(-\left\| x - x' \right\|^2 / \sigma^2) \tag{2.18}$$

and *sigmoid* kernel

$$k(x, x') = \tanh(\gamma \langle x, x' \rangle + \Theta) \tag{2.19}$$

For more detail about the SVM we recommend besides the original publication, the books of Schölkopf and Smola [2002] and Hamel [2009]

2.1.2. The random forest (RF) classifier

The RF classifier uses an ensemble of simple classification and regression trees (CARTs) to produce the classification result. Each of these CARTs for itself has a "weak" predictive power but the whole ensemble builds a fast classifier with high predictive power [Breiman, 2003] even in a high dimensional space. Breiman claims that the RF achieves equivalent classification accuracy as state of the art classification algorithms, such as SVMs.

A CART [Breiman et al., 1984] is a binary decision tree that can be considered the base rule of a RF. With CART, we get a classifier which is a piece-wise constant function obtained by partitioning the predictor space. CARTs are grown starting with the root node which contains the whole learning sample and ends when every node contains only one class. To create the child node the data of the parent node are separated according to a splitting criterion.

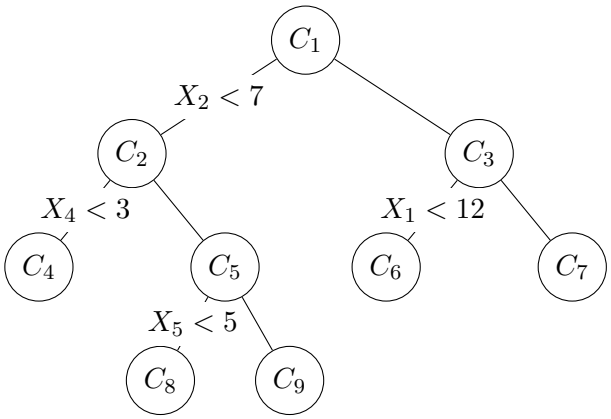


Figure 2.2.: Example of CART model. The set C_1 is step-by-step broken down to atomic sets.

The example in Figure 2.2 illustrates such a partition, where the classifier $h(x)$ is a mapping $h : \mathbb{R}^5 \mapsto \{0, 1\}$, say $h(x) = C_4 \mathbb{1}_{x_2<7, x_4<3} + C_8 \mathbb{1}_{x_2<7, x_4 \geq 3, x_5<5} +$

$C_9\mathbb{1}_{x_2<7,x_4\geq 3,x_5\geq 5} + C_6\mathbb{1}_{x_2\geq 7,x_1<12} + C_7\mathbb{1}_{x_2\geq 7,x_1\geq 12}$.

The most common splitting criterion is the *Gini impurity* ϕ:

$$\phi(\boldsymbol{p}) = \sum_{c=1}^{C} p_c(1-p_c) \tag{2.20}$$

where the $\boldsymbol{p} = (p_1, \ldots, p_c, \ldots, p_C)^T$ denotes the proportions of the C classes in a node. The function $\phi(\boldsymbol{p})$ is convex in $\boldsymbol{p}$. It assumes its maximum when all p_c are equal and its minimum when $\exists c$ s.t. $p_c = 1$.

The CARTs used in RFs differ slightly from the standard CART insofar that at each node only $q \ll Q$ randomly selected variables are used for the split and not the total number of variables Q. The number of variables to perform the split is reduced to turn the algorithm even faster. q is the same for all nodes of all trees in the forest. The tree is grown to its maximum extent and is not pruned. The RF combines the CARTs by performing a bagging (bootstrap aggregating) over all CARTs. This procedure is explained in the following.

Let $\mathcal{B}_0$ be the dataset of size M,

$$\mathcal{B}_0 = \{(\boldsymbol{x}_i; y_i)\}_{i=1}^{M} \tag{2.21}$$

where $\boldsymbol{x}_i$ is the i-th value of a p-vector of explanatory variables and y_i is the class label related to example $\boldsymbol{x}_i$.

From the original data set $\mathcal{B}_0$, one generates B bootstrapped sets $\mathcal{B}_b^*, 1 \leq b \leq B$, (i.e. B uniform samples of n data points in $\mathcal{B}_0$ with replacement). For any generated data set $\mathcal{B}_b^*$, an estimator of the classification function h_b is found by application of the CART procedure. So the bootstrap procedure provides B replications $h_b, b = 1, \ldots, B$ for the model (see Figure 2.3).

A *bootstrapped sample* $\boldsymbol{x}^{*b} = (\boldsymbol{x}_1^{*b}, \cdots, \boldsymbol{x}_M^{*b})$ is built by random drawing (with replacement) from the initial sample $\boldsymbol{x}$:

$$P_U(\boldsymbol{x}_i^{*b} = \boldsymbol{x}_j) = \frac{1}{M}, \qquad i, j = (1, \ldots, M), \tag{2.22}$$

where P_U is the uniform distribution on the original data set $\boldsymbol{x} = (\boldsymbol{x}_1, \cdots, \boldsymbol{x}_M)$.

To obtain a RF of T trees with M examples and Q variables we proceed as in Algorithm 1.

The classification for a new example is obtained by presenting the example to *all* trees in the forest. Each tree votes for the class of the new example and the random

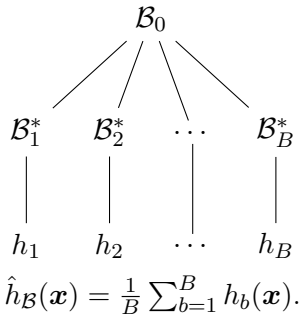


Figure 2.3.: Bagging procedure with CART models. Samples $\mathcal{B}_b^*$ are drawn from $\mathcal{B}_0$ and a predictor h_b for each sample generated.

Algorithm 1 Growing of random forests.

Require: number of trees T, $q \ll Q$ the number of variables that has to be tested at each node to choose the best split.

1: **repeat**
2: Draw a random sample of examples of size M with replacement.
3: Grow a CART with this sample of examples and calculate the best split between the variables with a random selection of q variables.
4: **until** T trees have been grown.

forest assigns the class that got the most votes (majority vote).

A very convenient property of RFs is that they provide a measure for prediction error without additional cross-validation. By drawing the sample for each tree at random with replacement while training the RF, several examples are drawn multiple times and others not at all. For a sample size of $n > 40$ the probability that an example appears in the sample is $1 - (1 - \frac{1}{n})^n \approx 0.632$. With a fraction of about 63.2 % of examples to be used in the growing of a tree there is a remaining fraction of $\approx 100 - 63.2 = 36.8\,\%$ examples that were not used to grow the tree. These examples are called out-of-Bag (ooB) examples. Presenting each tree in a sufficiently large forest its out-of-Bag (ooB) examples we obtain enough votes for each example to obtain a class prediction by majority vote as in the classification of new examples. The prediction error of the ooB examples is a good estimate for the predictive power of the RF.

2.2. Importance measurement using RFs

The RF classifier allows the calculation of the importance of the features involved in the decision. There are two commonly used importance measures in the RF frame-

work: the permutation importance and the Gini, or decrease of node impurity. The Gini was introduced by Breiman et al. [1984] and is used to perform the splits in the RF. But as an importance measure the Gini has been shown to be biased by Strobl et al. [2006]. Permutation importance was further analyzed in Strobl and Zeileis [2008] and Boulesteix et al. [2011]. As the permutation importance has no disadvantages over Gini impurity as importance measure we will further use the permutation importance.

The permutation importance measure is based on the assumption that the permutation of a predictor variable $\boldsymbol{x}_q$ with high importance will lead to a higher loss in classification accuracy than a less important predictor variable.

For each classifier h_b we consider the corresponding ooB sample and permute at random the values of the q-th variable of the sample. Then we compute the ooB error of h_k with these modified ooB data. The variable importance of the q-th variable is defined as the increase of ooB error after permutation. The more the increase of ooB error is, the more important is the variable.

The importance $\xi_q^{*(t)}$ of a variable $\boldsymbol{x}_q$ in tree t is defined by

$$\xi_q^{*(t)} = \frac{\sum_{i \in \mathrm{OOB}^{*(t)}} \delta(y_i, \hat{y}_i^{*(t)})}{M_{\mathrm{OOB}^t}} - \frac{\sum_{i \in \mathrm{OOB}^{*(t)}} \delta(y_i, \hat{y}_{i,\pi_q}^{*(t)})}{M_{\mathrm{OOB}^t}}, \tag{2.23}$$

where $\hat{y}_i^{*(t)}$ is the predicted class label for example i in tree t without any permutation and $\hat{y}_{i,\pi_q}^{*(t)}$ the predicted class label for example i after permuting the values of the q-th variable. M_{OOB^t} is the number of ooB examples in tree t and $\delta(\cdot,\cdot)$ is the Dirac delta-function which is 1 if its arguments are equal and zero in all other cases. While the theoretical upper bound of the variable importance $\xi_q^{*(t)}$ is 1 it is much more realistic to expect a maximum variable importance of 0.5. Assuming that the permutation of variable $\boldsymbol{x}_q$ would turn the before permutation perfect classifier (first term of Equation 2.23 equals 1) to a random guess (second term of Equation 2.23 equals 0.5) the resulting variable importance would be 0.5. Note that especially in RFs with low predictive power variable importance can assume negative values.

The overall importance of variable q is calculated as the mean over all trees:

$$\xi_q = \frac{\sum_{t=1}^{T} \xi_q^{*(t)}}{T}. \tag{2.24}$$

2.3. Creating an importance map using the RF classifier

Our approach consists in dividing the image in non-overlapping patches and calculating for each patch one feature. The features calculated from a set of examples are then fed to the RF classifier which provides us with the feature importance. Due to the direct link of the features to a well defined region in the image we obtain by calculating the feature importance as described in the preceding section at the same time information about the importance of the corresponding region in the image.

The RF classifier was chosen for this approach as it allows the calculation of feature importance and is capable to handle large numbers of features. It is at the same time fast and easy in handling. Furthermore, due to the internal re-sampling no cross-validation is necessary to obtain an estimate for the predictive power as detailed in subsection 2.1.2.

In order to create an importance map using the RF classifier we have first to ensure that all example images have the same size and that the motive has the same position and orientation in the image. The exact procedure of achieving these prerequisites depends on the type of image. However, a large amount of utilities for cropping, reorientation, interpolation, etc. suited for all kind of images is available which makes this step a possibly time consuming but easy task.

To transform the image in an array of non-overlapping patches we define the *non-overlapping patch transform*. With an application to volumetric images in mind we will define the transform on 3 dimensions. The two dimensional version can be obtained by omitting the 3$^{\text{rd}}$ dimension.

Definition 2.5 (non-overlapping patch transform)
Let $I(j,k,\ell)$ be an image with dimensions $J \times K \times L$ and j,k,ℓ the indices positioning the pixels in this image. Let further P be an array with dimensions $U \times V \times W$ where each entry is a tensor of dimensions $E \times Z \times H$. This array of tensors is denoted as $P(u,v,w;\epsilon,\zeta,\eta)$. Then the transformation of an image I to an array of tensors $P(u,v,w;\epsilon,\zeta,\eta)$ is called non-overlapping patch transform *if the transformation follows*

$$I(j,k,\ell) \rightarrow P(u,v,w;\epsilon,\zeta,\eta) \tag{2.25}$$

with

$$u = j \div E, \qquad v = k \div Z, \qquad w = l \div H \tag{2.26}$$

$$q = j \bmod E, \qquad r = k \bmod Z, \qquad s = l \bmod H \tag{2.27}$$

Each tensor in P is called a patch *and its dimensions $E \times Z \times H$ are called* patch size. *The number of patches is $M = (J \div E)(K \div Z)(L \div H)$.*

Non-overlapping and overlapping patches are often used in image processing and analysis. It is used for image restoration [Bertalmío et al., 2000], object recognition [Deselaers et al., 2005] and image editing [Cho et al., 2010] to give only a few examples.

The patch size has to be defined such that the calculation of features for small regions of the image is possible. For the selection of the patch size the size of the image as well as possibly available prior-knowledge about the feature size has to be considered. For a smaller patch size the spatial resolution of the obtained importance map will increase together with the number of patches and thus computational cost at the same time. For the RF classifier we suggest to stay well below 100 000 patches and consequently features. A balance between spatial resolution and discriminative power of the chosen feature has to be aspired.

Next we calculate the features for the defined patches forming the feature vector $\boldsymbol{x}_i$ for examples $1 \leq i \leq M$. The feature is a function f of the patches that does not depend on the u, v, w. We vectorize the array of patches

$$P(u, v, w; \epsilon, \zeta, \eta) \rightarrow P(m; \epsilon, \zeta, \eta) \tag{2.28}$$

$$\text{with} \qquad m = u + v(J \div E) + w(K \div Z) \tag{2.29}$$

and

$$\boldsymbol{x}_i = [f(P(1; \epsilon, \zeta, \eta)), \ldots, f(P(g, \epsilon, \zeta, \eta)), \ldots, f(P(G; \epsilon, \zeta, \eta))] \tag{2.30}$$

In this approach the feature can be chosen quite freely. The only restriction is that one feature per patch has to be produced and that the patches provide the information necessary to calculate the chosen feature. By restricting the number of features to one per patch we avoid to obtain multiple importance values per patch. We could for example choose a feature as simple as the mean intensity of the patch or its variance, but also more complex texture features. The calculated features of all available examples form the input data for the RF classifier.

Then the RF is grown. As we will calculate the feature importance the number of trees should be at least 500, better 1000 [Breiman, 2003]. Breiman proposes a default number of splits tried of $\sqrt{\#\text{ of features}}$. This value usually had not to be changed. The RF will return an estimate of prediction error. The lower the prediction error the higher our confidence in the obtained feature importance.

The feature importance is a vector $\boldsymbol{\xi}$ of the same length as the feature vector. For visual inspection and further use the importance values of each patch are attributed to all pixels belonging to the patch.

$$I_{\text{imp}}(j, k, \ell) = \boldsymbol{\xi}_{(j \div g) + (k \div g)(J \div g) + (l \div g)(K \div g)} \tag{2.31}$$

As the feature importance is not normalized we have to normalize the obtained map before display and/or further use. We normalize the map to an interval of $[0; 1]$ by min-max-normalization. In this way we obtain an importance image of the same size as the input images. An overview over the whole algorithm is presented in Algorithm 2. An application scheme for the extraction of maps with different patch sizes is shown in Figure 2.4.

Algorithm 2 Importance map generation procedure.

Require: Images of same size
1: define patch size
2: calculate feature for each patch
3: train RF classifier
4: check predictive power of the RF
5: **if** classification accuracy insufficient **then**
6: adjust patch size or chose other feature if necessary
7: **else**
8: obtain importance for each patch
9: transform patch importance to pixel importance
10: **end if**
Return: importance map

The purpose of the use of patches in this approach is to reduce the number of variables to an amount that can be treated by the RF classifier and at the same time conserving as much localization of the features of the image as possible. In case the resolution of the images is very low we could consider using the pixels directly as the input to the RF. The pixels could however be a less discriminative feature than an adequately chosen feature on small patches.

2.4. Practical example

To get a more precise impression of the presented method we will present a practical example of application. The MIT center for biological and computational learning (CBCL) provides free of charge a database of 2429 face images[1]. Images of 300 different persons are included in the database. For each person there are between 3 and 9 images. The images are 19×19 pixels gray scale provided in PGM format.

[1] CBCL Face Database # 1, MIT Center For Biological and Computation Learning, `http://www.ai.mit.edu/projects/cbcl`

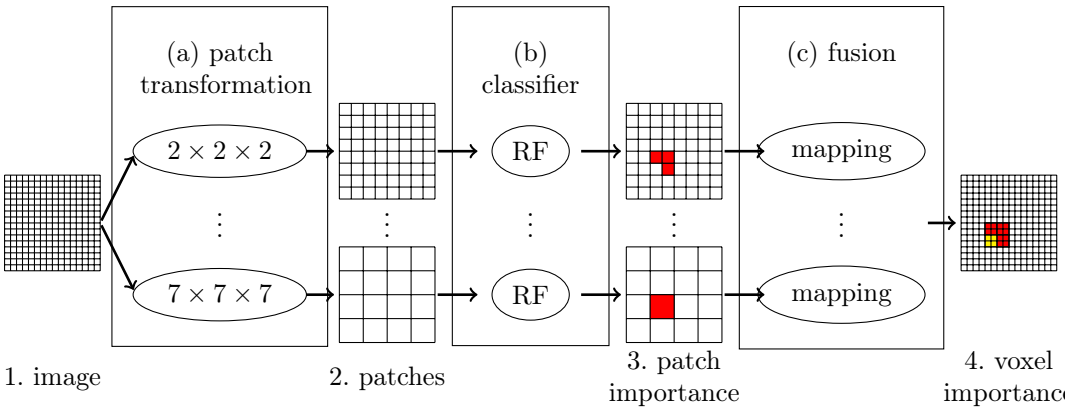
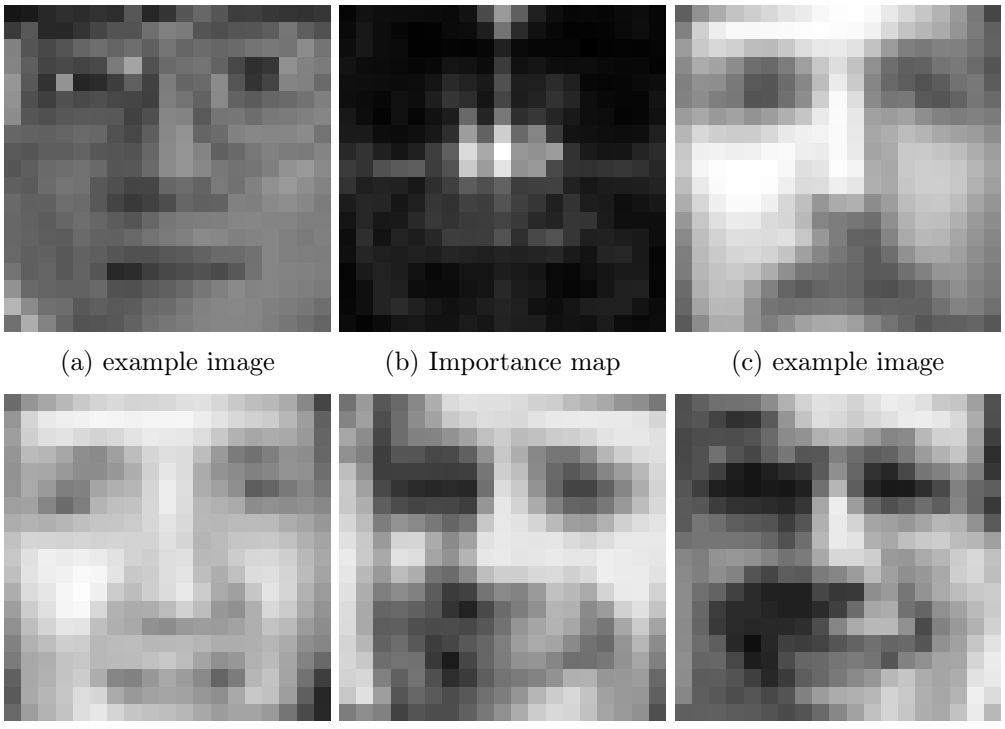


Figure 2.4.: Procedure for the extraction of importance maps at different patch sizes. The image is first transformed to a set of non-overlapping patches, then a feature for each patch is calculated and fed to the RF calssifier. The RF returns the classification and the feature importance. The feature importance corresponds to the importance of the patches. This patch importance can be mapped back to the image grid in order to obtain an importance map.

Our example consists of 210 persons with 6 images each. Given the low resolution of 19×19 pixels we consider the pixels already to be the mean intensity of patches. This leads to feature vectors $\boldsymbol{x}_i \in \mathbb{R}^{361}$. We train an RF classifier to classify the 1260 selected images into 210 classes, where each class corresponds to one person. Consequently the data matrix $\boldsymbol{X}$ fed to the RF has dimensions 1260×361.

The trained RF consisting of 1000 trees returns us the importance vector $\xi \in \mathbb{R}^{361}$. The ooB accuracy of the RF for this example was 97.8%. In Figure 2.5 we show the obtained importance map along with example images. The importance map shows that the area of the nose is most important for the classification. The eyes are of low importance in the classification. This might be caused by the low resolution which makes the eyes appear as dark spots without any distinctive features.

The importance map obtained from this simple example has a striking similarity with maps published by Peterson and Eckstein [2012]. The maps in this publication are generated by a Bayesian observer model and compared with the results of observation of human eye movement under stimulus by face images.



(a) example image (b) Importance map (c) example image

(d) example image (e) example image (f) example image

Figure 2.5.: Importance map obtained by the presented method on CBCL face images in the middle surrounded by example images.

2.5. Summary

In this chapter we introduced a procedure that uses the RF classifier as means to extract information about the spatial distribution of classification relevant information directly from image data. The image is sub-divided into patches and a feature calculated for each patch. The RF provides us with a variable importance measure which is used to produce a map picturing the distribution of information important to the performed classification.

As the map is directly related to the classifiers response to the presented examples and features it presents an information that is very specific to the problem. While a map generated using inter-class variance or a two-sample t-test would show only statistic differences for each patch individually our approach includes also the relation between the patches.

3. Non-negative tensor factorization (NTF)

Blind source separation (BSS) methods address the question how to decompose an array of quantitative data. The most known BSS techniques, principal component analysis (PCA) and independent component analysis (ICA), are performing a decomposition in the matrix domain. In the case of PCA the algorithm searches for an *uncorrelated* decomposition and in the case of ICA for a decomposition in *independent* components. The book of Comon and Jutten [2010] on BSS covers the whole family of BSS techniques and provides an excellent overview over this field.

Two families of techniques in the group of BSS techniques attract high interest in the last decade: the families of *non-negative* techniques and the family of *tensor* methods.

Non-negative methods apply a non-negativity constraint and are thus searching a decomposition into non-negative components. In consequence the input data also has to be non-negative as the mixing is strictly additive. This, however, is often the case in real-life applications like images or audio spectra. The non-negativity of all components facilitates visualization and interpretation.

Tensor methods extend matrix methods to arrays of higher order and allow therefore to represent the data in a more proper way than often possible with matrices. Especially in image applications where the usually applied vectorization of single images to store several examples in a matrix is an issue. The argument to justify vectorization matrix entries being independent is doubtable in the case of images as there is for sure a trivial relation of each pixel with its neighbors. Here tensor methods allow to keep the image in its original structure and thus conserving each pixels neighborhood relations.

This chapter of the thesis focuses on the combination of both non-negative and tensor methods to non-negative tensor methods called non-negative tensor factorization (NTF). NTF combines the advantages of both methods at the expense of high computational complexity. To make NTF applicable to real life problems with large data volume specialized algorithms are necessary. We will therefore present an algorithm based on a novel approach to reduce computational cost and memory consumption in the decomposition of large tensors.

Before presenting our novel NTF algorithm in section 3.4 we will provide an introduction to tensors (section 3.1). This includes the main definitions, the used tensor algebra and some tensor properties. Afterwards we present tensor decomposition models used in general tensor decomposition and NTF (section 3.2) and give an overview over related methods of the family of non-negative decompositions (section 3.3).

3.1. Introduction to tensors

In this section tensors are defined and the used tensor algebra is presented. Notations and definitions are largely adopted from Kolda [2006] and Cichocki et al. [2009]. The definitions of the less common matrix operations that are used can be found in Appendix A.

Definition 3.1 (Tensor)
A tensor of order N is a N-way array denoted by

$$\mathcal{Y} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N} \tag{3.1}$$

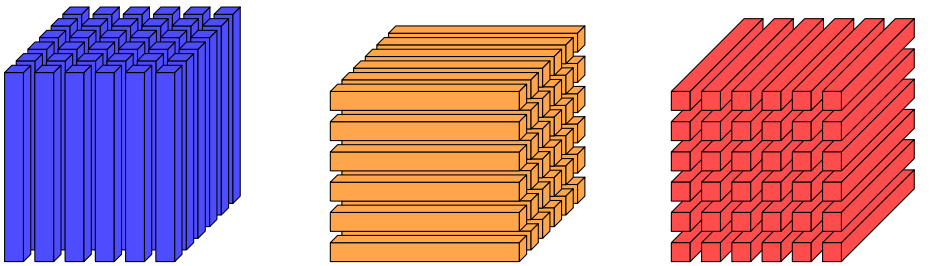
where $I_1, I_2, \ldots, I_N \in \mathbb{N}$ are the index upper bounds. Its elements are denoted by

$$y_{i_1, i_2, \ldots, i_N}. \tag{3.2}$$

Based on this definition of a tensor a matrix can be interpreted as a 2nd order tensor and a vector as a tensor of order 1. A scalar is consequently a tensor of order 0.

Definition 3.2 (Tensor fiber)
A tensor fibers is a one-dimensional fragment of a tensor, obtained fixing all but one index of the tensor. Or: A tensor fiber is a vector.

Following this definition the fibers of a 2nd order tensor, i.e. a matrix, are the columns or the rows of the matrix. Figure 3.1 shows the concept of tensor fibers on the example of a 3rd order tensor $\mathcal{Y} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$. Mode-1 fibers, for example, are obtained by fixing all but the first index of a tensor $\boldsymbol{y}_{:i_2 i_3}$. We use like in Matlab the : to denote a free index.

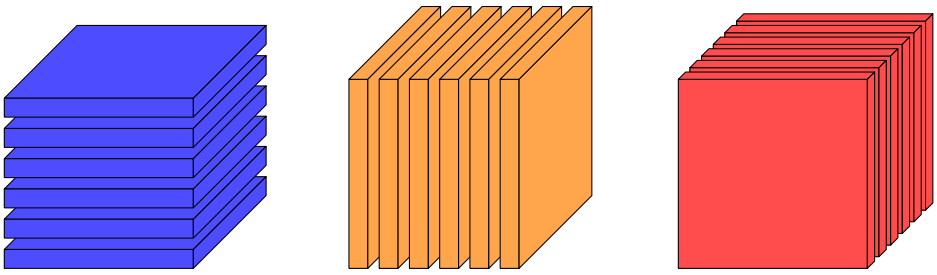


(a) mode-1 fibers, $\boldsymbol{y}_{:i_2 i_3}$ (b) mode-2 fibers, $\boldsymbol{y}_{i_1 : i_3}$ (c) mode-3 fibers, $\boldsymbol{y}_{i_1 i_2 :}$

Figure 3.1.: Fibers of 3rd order tensor $\mathcal{Y} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$.

Definition 3.3 (Tensor slice)
A tensor slice is a two-dimensional fragment of a tensor, obtained by fixing all but two indices of the tensor. Or: A tensor slice is a matrix.

To get a better idea of tensor slices Figure 3.2 shows the three possibilities to build slices in a 3rd order tensor.



(a) Horizontal slice, $\boldsymbol{Y}_{i_1::}$ (b) Lateral slice, $\boldsymbol{Y}_{:i_2:}$ (c) Frontal slice, $\boldsymbol{Y}_{::i_3}$

Figure 3.2.: Tensor slices of a 3rd order tensor $\mathcal{Y} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$.

3.1.1. Tensor algebra

Definition 3.4 (outer product)
The outer product of two tensors $\mathcal{Y} \in \mathbb{R}^{I_1 \times I_2 \times \ldots I_N}$ *and* $\mathcal{X} \in \mathbb{R}^{J_1 \times J_2 \times \ldots \times J_M}$ *is denoted by*

$$\mathcal{Z} = \mathcal{Y} \circ \mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N \times J_1 \times J_2 \times \ldots \times J_M} \tag{3.3}$$

where the elements of the resulting order NM *tensor* $\mathcal{Z}$ *are*

$$z_{i_1,i_2,\ldots,i_N,j_1,j_2,\ldots,j_M} = y_{i_1,i_2,\ldots,i_N} x_{j_1,j_2,\ldots,j_M} \tag{3.4}$$

In the case of two order one tensors, i.e. two vectors, $\boldsymbol{a} \in \mathbb{R}^I$ and $\boldsymbol{b} \in \mathbb{R}^J$ the result is a rank one matrix

$$\boldsymbol{A} = \boldsymbol{a} \circ \boldsymbol{b} = \boldsymbol{a}\boldsymbol{b}^T \in \mathbb{R}^{I \times J} \tag{3.5}$$

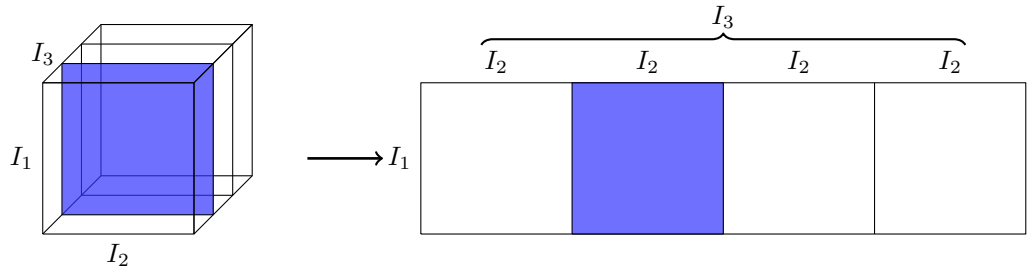
More details about the outer product of two vectors and the closely related Kronecker product of two matrices can be found in Appendix A.

Definition 3.5 (mode-n unfolding)
The mode-n unfolding of a tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times \ldots \times I_N}$ *will be denoted as* $\boldsymbol{X}_{(n)}$*. It arranges the mode-n fibers into columns of a matrix. The tensor element* $(i_1, \ldots, i_N)$ *is mapped to the matrix element* (i_n, j) *where*

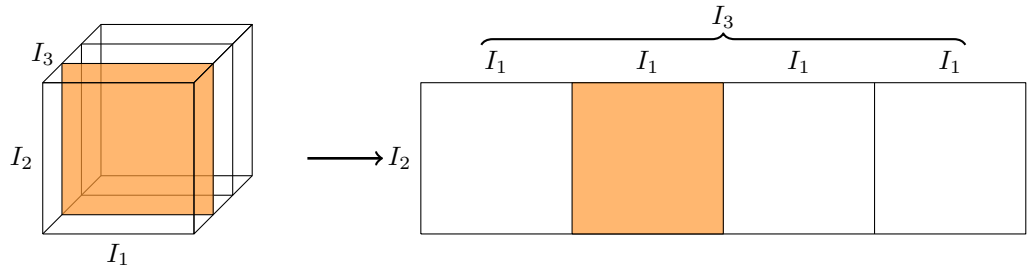
$$j = 1 + \sum_{p \neq n} (i_p - 1) J_p, \quad 1 \leq p \leq N$$

$$\text{with} \quad J_p = \begin{cases} 1 & \text{if } p = 1 \quad \text{or if} \quad p = 2 \quad \text{and} \quad n = 1, \\ \prod_{\substack{m=1 \\ m \neq n}}^{p-1} I_m & \text{otherwise} \end{cases} \tag{3.6}$$

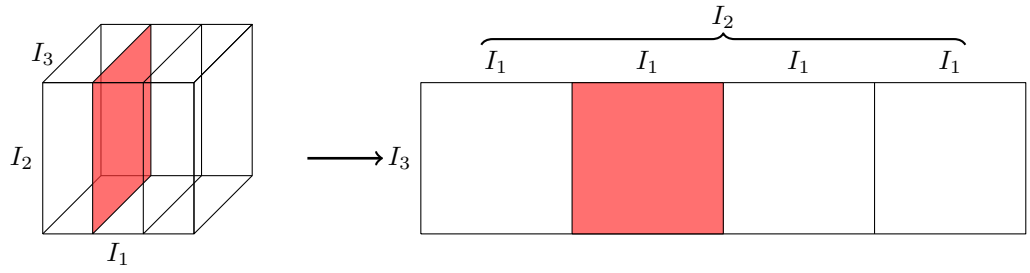
Mode-n unfolding is often also called mode-n matricization as the tensor is transformed to a matrix. Figure 3.3 gives a graphical example of this operation that is often used in tensor calculations.



(a) Mode-1 unfolding: $\boldsymbol{Y}_{(1)} \in \mathbb{R}^{I_1 \times I_2 I_3}$.



(b) Mode-2 unfolding: $\boldsymbol{Y}_{(2)} \in \mathbb{R}^{I_2 \times I_1 I_3}$.



(c) Mode-3 unfolding: $\boldsymbol{Y}_{(3)} \in \mathbb{R}^{I_3 \times I_1 I_2}$.

Figure 3.3.: Mode-n unfolding of a 3rd order tensor $\boldsymbol{\mathcal{Y}} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$.

Definition 3.6 (mode-n tensor matrix product)
The mode-n product $\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{G}} \times_n \boldsymbol{A}$ *of a tensor* $\boldsymbol{\mathcal{G}} \in \mathbb{R}^{I_1 \times \ldots \times I_n \times \ldots \times I_N}$ *and a matrix* $\boldsymbol{A} \in \mathbb{R}^{J \times I_n}$*, where* $1 \leq n \leq N$*, is a tensor* $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1 \times \ldots \times I_{n-1} \times J \times I_{n+1} \times \ldots \times I_N}$*, with elements*

$$x_{i_1,\ldots,i_{n-1},j,i_{n+1},\ldots,i_N} = \sum_{i_n=1}^{I_n} g_{i_1,\ldots,i_n,\ldots,i_N} a_{j,i_n}. \tag{3.7}$$

The mode-n tensor matrix product can be rewritten as matrix multiplication using the mode-n unfolding:

$$\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{G}} \times_n \boldsymbol{A} \Longleftrightarrow \boldsymbol{X}_{(n)} = \boldsymbol{A}\boldsymbol{G}_{(n)}. \tag{3.8}$$

where $\boldsymbol{G}_{(n)}$ denotes the mode-n unfolding of the tensor $\mathcal{G}$ as defined in mode-n tensor matrix product 3.5. Tensor matrix products along different modes are commutative. For instance, let $\mathcal{Y}$ be an order N tensor of dimensions $I_1 \times \ldots \times I_n \times \ldots \times I_m \times \ldots \times I_N$ and the matrices $\boldsymbol{A} \in \mathbb{R}^{J \times I_n}, \boldsymbol{B} \in \mathbb{R}^{K \times I_m}$ then

$$(\mathcal{Y} \times_n \boldsymbol{A}) \times_m \boldsymbol{B} = (\mathcal{Y} \times_m \boldsymbol{B}) \times_n \boldsymbol{A} = \mathcal{Y} \times_n \boldsymbol{A} \times_m \boldsymbol{B}, \qquad (m \neq n). \tag{3.9}$$

Definition 3.7 (Tucker operator)
Let $\mathcal{G} \in \mathbb{R}^{J_1 \times \ldots \times J_2 \times \ldots \times J_N}$ *and* $\boldsymbol{A}^{(n)} \in \mathbb{R}^{I_n \times J_n}$ *where* $1 \leq n \leq N$*. The Tucker operator* $[\![\cdot]\!]$ *is defined via the mode-n tensor matrix product as:*

$$\left[\!\left[\mathcal{G}; \boldsymbol{A}^{(1)}, \boldsymbol{A}^{(2)}, \ldots, \boldsymbol{A}^{(N)}\right]\!\right] \equiv \mathcal{G} \times_1 \boldsymbol{A}^{(1)} \times_2 \boldsymbol{A}^{(2)} \times_3 \ldots \times_N \boldsymbol{A}^{(N)} \tag{3.10}$$

There is an important equivalence of the Tucker operator with mode-n unfolding in combination with the Kronecker product.

$$\begin{aligned} \mathcal{X} &= \left[\!\left[\mathcal{Y}; \boldsymbol{A}^{(1)}, \ldots, \boldsymbol{A}^{(n)}, \ldots, \boldsymbol{A}^{(N)}\right]\!\right] \qquad \Longleftrightarrow \\ \boldsymbol{X}_{(n)} &= \boldsymbol{A}^{(n)} \boldsymbol{Y}_{(n)} \left(\boldsymbol{A}^{(N)} \otimes \ldots \otimes \boldsymbol{A}^{(n+1)} \otimes \boldsymbol{A}^{(n-1)} \otimes \ldots \otimes \boldsymbol{A}^{(1)}\right)^T \text{ for any } n \in \mathbb{N} \end{aligned} \tag{3.11}$$

This equivalence plays an important role in many tensor decomposition algorithms as tensor calculations can be transformed to matrix calculations using this equivalence.

Definition 3.8 (tensor rank)
The tensor rank $\operatorname{rank}(\mathcal{Y})$ *is defined as the smallest integer* r *such that the decomposition*

$$\mathcal{Y} = \sum_{i=1}^{r} \boldsymbol{a}_i^{(1)} \circ \boldsymbol{a}_i^{(2)} \circ \ldots \circ \boldsymbol{a}_i^{(N)} \tag{3.12}$$

of an order N *tensor* $\mathcal{Y}$ *holds exactly [Hitchcock, 1927a,b].*

Tensor rank is the generalization of matrix rank to higher orders. Regardless of order the $\operatorname{rank}(\mathcal{X}) = \operatorname{rank}(k\mathcal{X})$ for any scalar k and $\operatorname{rank}(\mathcal{X} + \mathcal{Y}) \geq \operatorname{rank}(\mathcal{X}) + \operatorname{rank}(\mathcal{Y})$.

While several properties of matrix rank, like the two just mentioned, are also true for tensor rank, there are also fundamental differences between matrix and tensor rank: There is no algorithm known capable to calculate the rank of a tensor with order ≥ 3, the rank of a real-valued tensor is in general not the same over $\mathbb{R}$ and $\mathbb{C}$, and the maximum possible rank of a tensor $\mathcal{Y} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ is unknown, while the maximum possible rank of a matrix $\boldsymbol{Y} \in \mathbb{R}^{I_1 \times I_2}$ is known to be

$$\operatorname{rank}_{\max}(\boldsymbol{Y}) = \min(I_1, I_2). \tag{3.13}$$

These are the most striking differences out of a longer list stated in Kruskal [1989].

3.1.2. Special tensors

Definition 3.9 (diagonal tensor)
A tensor $\mathcal{Y}$ *is called diagonal if only entries* $y_{i_1 \ldots i_N}$ *with* $i_1 = \ldots = i_N$*, i.e. entries on its diagonal, are different from zero. The tensor with diagonal entries equal one will be denoted as* $\mathcal{I}$*.*

$$(\mathcal{I})_{i_1 \ldots i_N} = \begin{cases} 1, & \text{if } i_1 = \ldots = i_N \\ 0, & \text{otherwise} \end{cases} \tag{3.14}$$

Definition 3.10 (non-negative tensor)
A tensor $\mathcal{Y}$ *is called non-negative if* all *its elements are non-negative*

$$y_{i_1,\ldots,i_N} \geq 0 \tag{3.15}$$

For non-negative tensors we use the short hands

$$\mathcal{Y} \geq 0 \qquad \text{and} \qquad \mathcal{Y} \in \mathbb{R}_{0+}. \tag{3.16}$$

Definition 3.11 (rank one tensor)
A tensor $\mathcal{Y} \in \mathbb{R}^{I_1 \times \ldots \times I_N}$ *of order* N *has rank one if it can be written as an outer product of* N *order one tensors, i.e.* N *vectors.*

$$\mathcal{Y} = \boldsymbol{a}^{(1)} \circ \boldsymbol{a}^{(2)} \circ \ldots \circ \boldsymbol{a}^{(N)} \tag{3.17}$$

where $\boldsymbol{a}^{(n)} \in \mathbb{R}^{I_n}$*.*

3.2. Tensor decomposition models

Models for the decomposition of tensors where first discussed by Hitchcock [1927a,b] in 1927. In the late 1960s tensor decompositions were rediscovered by Tucker [1966], Carroll and Chang [1970] and Harshman [1970] and promoted as Tucker decomposition, canonical decomposition (CANDECOMP) and parallel factor analysis (PARAFAC). These decompositions are higher-order generalizations of signular value decomposition (SVD). Tensor decompositions appear today in various fields including psychometrics, chemometrics, image processing, signal processing and others.

The main problem of tensor decompositions is introduced immediately: A rank reducing approximation of a tensor does not exist in general. For an overview over this issue see de Silva and Lim [2006]. As the same models are used in non-negative tensor decomposition we will present in the following, regardless of the just mentioned issue, tensor decomposition models and their advantages.

3.2.1. Canonical polyadic (CP) model

The CP model, also known as PARAFAC and CANDECOMP model, decomposes a tensor as a sum of outer-products of vectors. A given data tensor $\mathcal{Y} \in \mathbb{R}^{K \times L \times M}$ is approximated by three component, or factor matrices, $\boldsymbol{A} = [\boldsymbol{a}_1, \ldots, \boldsymbol{a}_J] \in \mathbb{R}^{K \times J}$, $\boldsymbol{B} = [\boldsymbol{b}_1, \ldots, \boldsymbol{b}_J] \in \mathbb{R}^{L \times J}$ and $\boldsymbol{C} = [c_1, \ldots, c_J] \in \mathbb{R}^{M \times J}$ as

$$\mathcal{Y} \approx \sum_{j=1}^{J} \boldsymbol{a}_j \circ \boldsymbol{b}_j \circ \boldsymbol{c}_j = \mathcal{I} \times_1 \boldsymbol{A} \times_2 \boldsymbol{B} \times_3 \boldsymbol{C} \tag{3.18}$$

or element wise

$$y_{klm} \approx \sum_{j=1}^{J} a_{kj} b_{lj} c_{mj}. \tag{3.19}$$

The tensor $\mathcal{I}$ is $J \times J \times J$ diagonal, with ones on its diagonal. The core tensor determines the interactions between the factor matrices. With the core tensor having only ones on its diagonal the interactions between the factor matrices are all equal. As the tensor is decomposed as sum of rank one tensors. Consequently, J is the rank of the decomposition. The CP model is visualized in Figure 3.4.

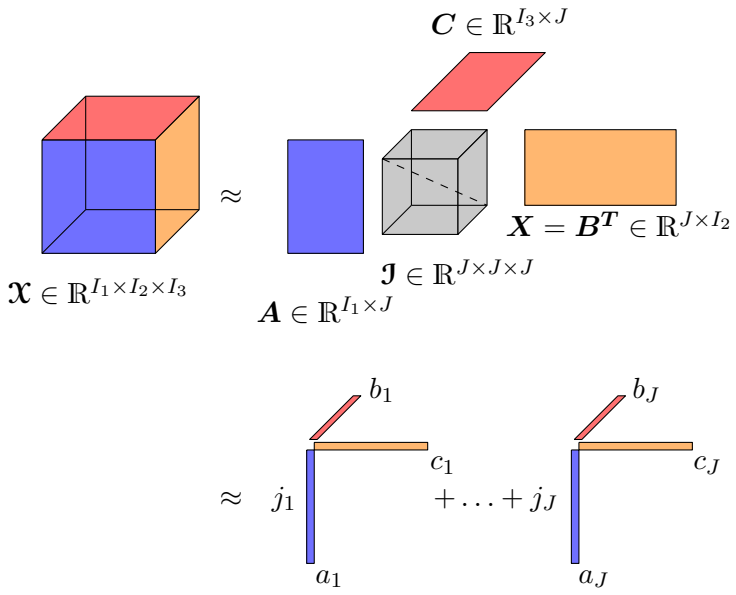


Figure 3.4.: Scheme of the CP model (Equation 3.18) in matrix and in vector form; The core tensor $\mathcal{J}$ is symmetric with entries 1 on the diagonal.

3.2.2. Tucker3 model

The Tucker3 model is a generalization of the CP model. The super-diagonal core tensor $\mathcal{J}$ is replaced by a general core tensor $\mathcal{G} \in \mathbb{R}^{J_1 \times J_2 \times J_3}$ with entries $g_{opq} \in \mathbb{R}_0^+$.

$$\mathcal{Y} \approx \sum_o \sum_p \sum_q g_{opq} \boldsymbol{a}_o \circ \boldsymbol{b}_p \circ \boldsymbol{c}_q = \mathcal{G} \times_1 \boldsymbol{A} \times_2 \boldsymbol{B} \times_3 \boldsymbol{C} \tag{3.20}$$

or element wise

$$y_{klm} \approx \sum_o \sum_p \sum_q g_{opq} a_{ko} b_{lp} c_{mq} \tag{3.21}$$

This allows every possible additive interaction between the factor matrices. Furthermore this generalization makes it possible to normalize all factor matrices and "absorb" the scale of the data into the core tensor. As for the CP model a graphical visualization of the Tucker3 model is provided in Figure 3.5.

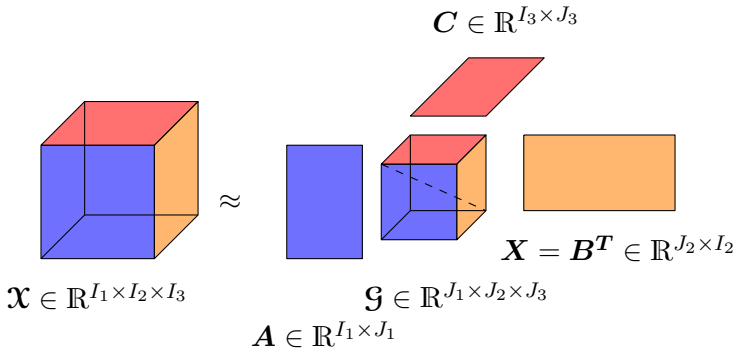


Figure 3.5.: Scheme of the Tucker model (Equation 3.20); The core tensor can assume any values and might change as well during the decomposition.

3.2.3. N-dimensional Tucker model

The N-dimensional Tucker model generalizes the Tucker3 model to N-dimensions and will be used in the remainder of the chapter to derive the new method of NTF. With a data tensor $\mathcal{Y} \in \mathbb{R}^{I_1 \times \ldots \times I_n \times \ldots \times I_N}$, the core tensor $\mathcal{G} \in \mathbb{R}^{J_1 \times \ldots \times J_n \times \ldots \times J_N}$ and I_n, J_n $(1 \leq n \leq N)$ the index upper bounds, the N-dimensional Tucker model is stated as

$$\mathcal{Y} \approx \mathcal{G} \times_1 \boldsymbol{A}^{(1)} \times_2 \boldsymbol{A}^{(2)} \times_3 \ldots \times_n \boldsymbol{A}^{(n)} = \left[\!\left[\mathcal{G}; \boldsymbol{A}^{(1)}, \ldots, \boldsymbol{A}^{(N)} \right]\!\right] \tag{3.22}$$

with the factor matrices $\boldsymbol{A}^{(n)} \in \mathbb{R}^{I_n \times J_n}$. The N-dimensional Tucker model gives us the freedom to use as many dimensions as desired to describe the data we want to decompose.

3.2.4. Advantages of tensor models

To understand the motivation for a tensor decomposition model let us first recall the data model usually used in BSS. The mixing model can be written in vector-matrix notation as

$$\boldsymbol{x} = \boldsymbol{A}\boldsymbol{s}, \tag{3.23}$$

where $\boldsymbol{x}$ is the vector of observations (measured the mixtures), $\boldsymbol{s}$ the vector of the underlying sources that can not be observed directly and $\boldsymbol{A}$ the mixing matrix.

This model is adapted to one-dimensional signals, e.g. a time series signal. But if we intend to search for underlying sources for a two-dimensional image, a three-dimensional brain scan or a set of images or measurements this kind of data model is no longer directly applicable. To make the data fit into the model it is common practice to *vectorize* the measured data, which means to concatenate the data to a vector. This vectorization is justified by the assumption that the $\boldsymbol{x}_i$ are independent which might not be strictly true.

Consider for example the image of a human face, the calculation of Eigen-faces is a quite common application of ICA, PCA, and other BSS techniques, or brain images acquired by MRI, single photon emission computed tomography (SPECT) or PET. It is evident that the pixels of the images are related to their neighboring pixels, thus the pixels are not independent as required to justify vectorization. Nevertheless, many real data applications do show good results using vectorization on face decomposition, brain imaging and other applications [Illán et al., 2011, Kodewitz et al., 2010, Turk and Pentland, 1991, Zafeiriou et al., 2006]. Literature provides to our knowledge no insight why the vectorization approach is successful even with the pixels being dependent.

In a tensor model on the contrary it is possible to maintain the data in its natural shape. A vectorization is not necessary and in the case of image data pixel neighborhoods remain intact as in a tensor model we can use one separate index for each dimension or measurement. In the case of a series of images, for example, we can reserve two indices for the image dimensions and one index for the different images which leads to a tensor of order 3. A time-series of CT for different patients would form a tensor of order 5.

In addition to this advantage that arises in practical use there is also a big theoretical advantage of tensor decompositions over BSS techniques in the matrix domain: The decomposition, if it exists, is unique. This property is examined along with the rank of the decomposition in Kruskal [1989]. Without uniqueness the decomposition algorithm could converge to several equivalent solutions, which makes the comparison of different decompositions impossible.

3.3. Non-negative decompositions

A non-negativity constraint facilitates the interpretation of the decomposition as input data as well as all components are constrained to be non-negative. The decomposition is therefore strictly additive and graphical display of non-negative data is more convenient. But even more important, the non-negativity constraint guaranties the existence of a rank reducing solution to the non-negative CP problem as proved in Lim and Comon [2009]. This proof holds as well for the matrix case (order 2) as for the tensor case (order ≥ 3).

3.3.1. Non-negative decomposition in the matrix domain

Non-negative matrix factorization (NMF), i.e a decompositions of the form

$$\boldsymbol{V} \approx \boldsymbol{W}\boldsymbol{H} \qquad \text{s.t. } \boldsymbol{W}, \boldsymbol{H} \geq 0, \tag{3.24}$$

became popular by an article on NMF by Lee and Seung [1999]. They put forward that the decomposition by PCA, not imposing any constraint to the sign of the decomposition, has the disadvantage of having a statistical but no visual interpretation. NMF on the contrary performs, due to its non-negativity constraint, a decomposition of the source into two matrices with positive entries. The source matrix contains necessarily also only non-negative entries. While the non-negativity constraint does not guarantee an unique solution, the resulting decomposition allows easy interpretation as the source on the non-negative domain is represented by a decomposition that is again on the non-negative domain.

The implementation by Lee and Seung used a multiplicative update rule that ensures non-negativity and cost-functions based on the Euclidean norm[1]

$$D(\boldsymbol{V}\|\boldsymbol{W}\boldsymbol{H}) = \|\boldsymbol{V} - \boldsymbol{W}\boldsymbol{H}\|_F^2 \tag{3.25}$$

$$\text{s.t. } \boldsymbol{W}, \boldsymbol{H} \geq 0 \tag{3.26}$$

where $\|\cdot\|_F^2$ denotes the Frobenius norm, or alternatively Kulback-Leibler divergence

$$D(\boldsymbol{V}\|\boldsymbol{W}\boldsymbol{H}) = \sum_{ij} \boldsymbol{V}_{ij} \log \frac{\boldsymbol{V}_{ij}}{(\boldsymbol{W}\boldsymbol{H})_{ij}} - \boldsymbol{V}_{ij} + (\boldsymbol{W}\boldsymbol{H})_{ij} \tag{3.27}$$

$$\text{s.t. } \boldsymbol{W}, \boldsymbol{H} \geq 0. \tag{3.28}$$

The paper by Lee and Seung was followed by a large amount of publications presenting other algorithms for NMF [Cao et al., 2007, Cichocki and Phan, 2009, Lee and

[1] The Euclidean norm is not used in Lee and Seung [1999] but in Lee and Seung [2000].

Seung, 2000] and many others. Several publications proposed to combine NMF with additional constraints, especially with a sparseness constraint [Hoyer and Dayan, 2004, Pascual-Montano et al., 2006, Peharz and Pernkopf, 2012]. The additional sparseness constraint enforces a *parts based* decomposition [Lee and Seung, 1999]. Furthermore, by adding this constraint the most sparse decomposition becomes the unique solution to the decomposition problem. Applications of NMF in various fields, like face recognition [Kotsia et al., 2007, Stouten et al., 2007, Yang and He, 2010, Zafeiriou et al., 2006], speech recognition Driesen et al. [2009] and medical image analysis [Lee et al., 2001].

3.3.2. Non-negative decomposition in the tensor domain

Non-negative CP is a multi-way generalization of NMF. It adds a non-negativity constraint to the CP model, as stated in Equation 3.18. In the literature it is mostly called NTF but we would prefer to reserve this term to general non-negative tensor decompositions without the implication of any underlying model. Non-negative CP is also known as *non-negative* PARAFAC, e.g. Paatero [1997].

The non-negative CP model has the form

$$\mathcal{Y} \approx \mathcal{I} \times_1 \boldsymbol{A}^{(1)} \times_2 \ldots \times_N \boldsymbol{A}^{(N)} = \left[\!\left[\mathcal{I}; \boldsymbol{A}^{(1)}, \ldots, \boldsymbol{A}^{(N)}\right]\!\right] \tag{3.29}$$

$$\text{s.t. } \boldsymbol{A}^{(1)}, \ldots, \boldsymbol{A}^{(N)} \geq 0 \tag{3.30}$$

where $\mathcal{Y} \in \mathbb{R}_{0+}^{I_1 \times I_2 \ldots \times I_N}$ is a non-negative N-way tensor, $\mathcal{I} \in \mathbb{R}_{0+}^{J \times J \times \ldots \times J}$ the diagonal core tensor of order N with entries one and $\boldsymbol{A}^{(n)} \in \mathbb{R}_{0+}^{I_n \times J}$ the factor matrices. Alternatively the NTF model can be written in vector form

$$\mathcal{Y} \approx \sum_{j=1}^{J} \boldsymbol{a}_j^{(1)} \circ \boldsymbol{a}_j^{(2)} \circ \ldots \circ \boldsymbol{a}_j^{(N)} \tag{3.31}$$

$$\text{s.t. } \boldsymbol{a}_j^{(1)}, \ldots, \boldsymbol{a}_j^{(N)} \geq 0 \tag{3.32}$$

Non-negative tucker decomposition (NTD) applies the Tucker model, as stated in Equation 3.20 or Equation 3.22, with an additional non-negativity constraint. In comparison to the NTF model the core tensor is no longer fixed and diagonal but can assume non-negative entries on all entries.

The non-negative Tucker decomposition (NTD) model takes the form

$$\mathcal{Y} \approx \mathcal{G} \times_1 \boldsymbol{A}^{(1)} \times_2 \ldots \times_N \boldsymbol{A}^{(N)} = \left[\!\left[\mathcal{G}, \boldsymbol{A}^{(1)}, \ldots, \boldsymbol{A}^{(N)}\right]\!\right] \tag{3.33}$$

$$\text{s.t. } \mathcal{G}, \boldsymbol{A}^{(1)}, \ldots, \boldsymbol{A}^{(N)} \geq 0 \tag{3.34}$$

where $\mathcal{Y} \in \mathbb{R}_{0+}^{I_1 \times I_2 \ldots \times I_N}$ is a non-negative N-way tensor, $\mathcal{G} \in \mathbb{R}_{0+}^{J_1 \times J_2 \times \ldots \times J_N}$ the core tensor and $\boldsymbol{A}^{(n)} \in \mathbb{R}_{0+}^{I_n \times J_n}$ the factor matrices.

Implementations of NTF as direct multi-way generalization to the Lee and Seung NMF algorithm were presented by Kim and Choi [2007], Lee et al. [2007], Mørup et al. [2008], Welling and Weber [2001]. These implementations use the same multiplicative update rule and the Euclidean norm or Kulback-Leibler divergence as cost-function as Lee and Seung. More specifically, an alternating minimization of the set of non-negativity constrained least squares (NNLS) problems

$$\boldsymbol{A}^{(n)} = \arg\min_{\boldsymbol{A}^{(n)}} \left\| \boldsymbol{Y}_{(n)} - \boldsymbol{A}^{(n)} \boldsymbol{Z}_{(n)} \right\|_2^2 \tag{3.35}$$

$$\text{s.t. } \boldsymbol{A}^{(n)} \geq 0 \tag{3.36}$$

or in case of the Kulback-Leibler divergence

$$\boldsymbol{A}^{(n)} = \arg\min_{\boldsymbol{A}^{(n)}} \sum_{i_1,\ldots,i_N} \mathcal{Y}_{i_1,\ldots,i_N} \log \frac{\mathcal{Y}_{i_1,\ldots,i_N}}{(\boldsymbol{A}^{(n)} \boldsymbol{Z}_{(n)})_{i,\ldots,i_N}} - \mathcal{Y}_{i_1,\ldots,i_N} + (\boldsymbol{A}^{(n)} \boldsymbol{Z}_{(n)})_{i_1,\ldots,i_N} \tag{3.37}$$

$$\text{s.t. } \boldsymbol{A}^{(n)} \geq 0 \tag{3.38}$$

where

$$\boldsymbol{Z}_{(n)} = \boldsymbol{G}_{(n)} (\boldsymbol{A}^{(N)} \otimes \boldsymbol{A}^{(N-1)} \otimes \ldots \otimes \boldsymbol{A}^{(n+1)} \otimes \boldsymbol{A}^{(n-1)} \otimes \ldots \otimes \boldsymbol{A}^{(1)})^T \tag{3.39}$$

is performed using the multiplicative update rule proposed by Lee and Seung. The update rule used in the case of the Euclidean norm is

$$\boldsymbol{A}^{(n)} \leftarrow \boldsymbol{A}^{(n)} \bullet \frac{\boldsymbol{Y}_{(n)} \boldsymbol{Z}_{(n)}}{\boldsymbol{A}^{(n)} \boldsymbol{Z}_{(n)} \boldsymbol{Z}_{(n)}^T} \tag{3.40}$$

and in the case of the Kullback-Leibler divergence

$$\boldsymbol{A}^{(n)} \leftarrow \boldsymbol{A}^{(n)} \bullet \frac{\frac{\boldsymbol{Y}_{(n)}}{\boldsymbol{A}^{(n)} \boldsymbol{Z}_{(n)}} \boldsymbol{Z}_{(n)}^T}{\boldsymbol{E}_{(n)} \boldsymbol{Z}_{(n)}} \tag{3.41}$$

where the bold dot $\bullet$ denotes the element wise product, the bold fraction line denotes the element wise division and $\mathcal{E}$ is a tensor of same dimensions as $\mathcal{Y}$ with all entries equal 1 and $\boldsymbol{E}_{(n)}$ its mode-n matricization.

While Welling and Weber [2001] and Lee et al. [2007] use a CP model, Kim and Choi [2007] and Mørup et al. [2008] use a Tucker model and propose also update rules for the core $\mathcal{G}$. In the case of the Euclidean norm

$$\mathcal{G} \leftarrow \mathcal{G} \bullet \frac{\left[\!\left[\mathcal{Y}, \boldsymbol{A}^{(1)^T}, \ldots, \boldsymbol{A}^{(N)^T}\right]\!\right]}{\left[\!\left[\hat{\mathcal{Y}}, \boldsymbol{A}^{(1)^T}, \ldots, \boldsymbol{A}^{(N)^T}\right]\!\right]} \tag{3.42}$$

and in the case of Kulback-Leibler divergence

$$\mathcal{G} \leftarrow \mathcal{G} \bullet \frac{\left[\!\left[\frac{\mathcal{Y}}{\hat{\mathcal{Y}}}, \boldsymbol{A}^{(1)^T}, \ldots, \boldsymbol{A}^{(N)^T}\right]\!\right]}{\left[\!\left[\mathcal{E}, \boldsymbol{A}^{(1)^T}, \ldots, \boldsymbol{A}^{(N)^T}\right]\!\right]} \tag{3.43}$$

where $\hat{\mathcal{Y}} = \left[\!\left[\mathcal{G}, \boldsymbol{A}^{(1)}, \ldots, \boldsymbol{A}^{(N)}\right]\!\right]$ is the approximation of the data tensor $\mathcal{Y}$.

Several other approaches were proposed to perform NTF: gradient descent [Hazan et al., 2005], fixed point alternating least squares and alternating interior-point gradient [Cichocki et al., 2007], alternating least squares (ALS) [Friedlander and Hatz, 2008], hierarchical alternating least squares (HALS) [Cichocki and Phan, 2009], preconditioned nonlinear conjugate gradient [Royer et al., 2011] and a block principal pivoting method [Kim and Park, 2012].

Besides Euclidean norm and Kulback-Leibler divergence, especially the more general alpha-divergence

$$D_A^{(\alpha)}(\boldsymbol{p}\|\boldsymbol{q}) = \frac{1}{\alpha(\alpha-1)} \sum_i \left(p_i^\alpha q_i^{1-\alpha} - \alpha p_i + (\alpha-1) q_i\right), \quad \alpha \in \mathbb{R} \tag{3.44}$$

and beta-divergence

$$D_B^{(\beta)}(\boldsymbol{p}\|\boldsymbol{q}) = \sum_i \left(p_i \frac{p_i^\beta - q_i^\beta}{\beta} - \frac{p_i^{\beta+1} - q_i^{\beta+1}}{\beta+1}\right), \quad \beta \in \mathbb{R} \setminus \{-1, 0\} \tag{3.45}$$

found attention in the implementation of NTF and NTD. These divergences are measures of distance between the two distributions p and q. The vectors p and q correspond to the data tensor $\mathcal{Y}$ and its approximation $\hat{\mathcal{Y}}$ respectively. Note that the $D_B^{(1)}$ is the squared euclidean distance and $D_B^{(0)}$ is the Kullback-Leibler divergence.

3.4. Non-negative sub-tensor-set factorization ($\mathsf{N_S T_S F}$)

In the search for faster NTF algorithms various optimization strategies have been pursued. In the decomposition of large scale tensors memory also becomes a major factor. Block wise processing of the data, a common approach to implement parallel processing, has however the drawback that the blocks of data have to be accessed multiple times. This creates a memory overhead we want to avoid witch our novel algorithm designed for the decomposition of large scale tensors with imbalanced dimensions.

Consider a tensor of order N, $\mathcal{Y} \in \mathbb{R}_{0+}^{I_1 \times I_2 \times \ldots \times I_n \times \ldots \times I_N}$, where the $I_n, n = 1, \ldots, N$ are the index upper bounds for the N ways. We will interpret the tensor $\mathcal{Y}$ as a set of sub-tensors of order $N-1$, $\{\mathcal{Y}_t \in \mathbb{R}_{0+}^{I_1 \times \ldots \times I_{N-1}}, t = 1, \ldots, I_N\}$. Intuitively we can call each tensor $\mathcal{Y}_t$ an *example*.

With this interpretation of the data tensor we approach the problem of decomposing "large" tensors sub-tensor by sub-tensor. Treating one sub-tensor at a time we proceed until we obtain a decomposition of the whole set of sub-tensors. Therefore, we call the algorithm non-negative sub-tensor set factorization ($\mathrm{N_S T_S F}$). In this context we call a tensor "large" if the size of at least one dimension is several hundreds. The algorithm we will present is specially adapted to tensors with unbalanced dimensions, which means that one dimension is much larger then the others $I_m \gg I_1, \ldots, I_n, \ldots, I_N, m \neq n$. In the derivation of our algorithm we will assume that the largest dimension I_m is dimension I_N. This can always be achieved by re-indexing the tensor.

The core tensor $\mathcal{G} \in \mathbb{R}_{0+}^{R_1 \times \ldots \times R_n \times \ldots \times R_N}$, determining the interactions between the factor matrices $\boldsymbol{A}^{(n)}$, is fixed to

$$g_{i_1, i_2, \ldots, i_N} = \begin{cases} 1, & \text{if } i_1 = i_2 = \ldots = i_N \\ 0, & \text{otherwise} \end{cases} \tag{3.46}$$

with $R_1 = R_2 = \ldots = R_N = R < I_1, \ldots, I_N$. This means we are "de facto" using the CP model but derive the update rule on the more general Tucker model. We profit insofar from this choice that an update of the core tensor is not necessary and the implementation becomes easier. As we do not use this property imposed on the core tensor in the derivation of the algorithm we can loosen this constraint on the core at later time without the need to repeat the whole derivation.

The factor matrices of the factorization of $\mathcal{Y}$ are $\boldsymbol{A}^{(n)} \in \mathbb{R}_{0+}^{I_n \times R}, 1 \leq n \leq N$ and for the corresponding $\mathcal{Y}_t$ the factor matrix $\boldsymbol{A}^{(N)}$ splits up into one way tensors, i.e. vectors, $\boldsymbol{A}^{(N)} = [\boldsymbol{a}_1, \ldots, \boldsymbol{a}_t, \ldots, \boldsymbol{a}_{I_N}]^T$. That means each sub-tensor is coded by a vector $\boldsymbol{a}_t$ and the basis matrices $\boldsymbol{A}^{(1)}, \boldsymbol{A}^{(2)}, \ldots, \boldsymbol{A}^{(N-1)}$.

The algorithm we intend to apply is outlined in Algorithm 3 (page 42). It consists of two phases: the update of the factors $\boldsymbol{a}_t$ (coding step) and the update of the factors matrices $\boldsymbol{A}^{(1)}$ through $\boldsymbol{A}^{(N-1)}$ (update of basis factors). The optimizations of both phases are repeated until "sufficient" convergence, where the optimizations of the second phase are executed in alternation. We use a multiplicative update rule similar to the update rule of Lee and Seung with the Frobenius norm as cost function.

Coding step In this step the coding vector $\boldsymbol{a}_t$ is calculated for one $\mathcal{Y}_t$ at a time. Using the regularization term $\lambda \|\boldsymbol{a}_t\|_1 = \lambda \sum_i |a_i|$ in Equation 3.47 a *sparse* coding is enforced. $\lambda \in \mathbb{R}$ is the regularization parameter controlling the "amount" of sparseness. This means that a solution with many small or zero entries are favored over a dense solution. The sparse coding vector will influence the update of the basis factor insofar as the changes in the factor matrices will be concentrated on the part of the factors that was relevant in the coding of the corresponding example.

The sparse non-negative updating of the coding vector has to be stopped according to a stopping criterion that is adjusted to the factors needed precision of the sparse coding vs. avoidance of over-fitting and time needed for the sparse coding. For simplicity it was chosen to stop the updates when the change of the cost function $\Delta_{\text{cost}} < 10^{-5}$. This value was learned by experiment and proved to perform well for various input tensors.

Update of basis factors In this step the factor matrices $\boldsymbol{A}^{(1)}$ through $\boldsymbol{A}^{(N-1)}$ are updated. These factor matrices constitute the basis for the reconstruction of all examples $\mathcal{Y}_t$. Therefore, we call this step "update of basis factors".

We will now derive the optimization problem for the update of the basis factors (Equation 3.50) from the cost function for the decomposition of the whole tensor $\mathcal{Y}$

$$D = \frac{1}{2} \left\| \mathcal{Y} - \left[\!\left[\mathcal{G}; \boldsymbol{A}^{(1)}, \boldsymbol{A}^{(2)}, \ldots, \boldsymbol{A}^{(N)} \right]\!\right] \right\|_2^2. \tag{3.51}$$

The cost function

$$D = \frac{1}{2} \sum_{t=1}^{I_N} \left\| \mathcal{Y}_t - \left[\!\left[\mathcal{G}; \boldsymbol{A}^{(1)}, \boldsymbol{A}^{(2)}, \ldots, \boldsymbol{A}^{(N-1)}, \boldsymbol{a}_t \right]\!\right] \right\|_2^2 \tag{3.52}$$

is the cost function for the update of the set of $\mathcal{Y}_t$ as we intend to interpret the tensor. This cost function in Equation 3.52 will serve us as a surrogate for the cost function of the whole tensor in Equation 3.51.

A direct calculation of the sum in this cost function at each iteration for the known $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_t$ and the $\boldsymbol{A}^{(n)}$ would be too costly to create a fast algorithm. Therefore

Algorithm 3 Outline of the NSTSF algorithm.

Require: stream of tensor slices $\mathcal{Y}_s$ of order $N-1$

1: **for** all slices s **do**

2: **Coding step:**

3: optimize with $\boldsymbol{A}^{(1)}$ through $\boldsymbol{A}^{(N-1)}$ fixed (stop when $\Delta_{\text{error}} < 10^{-5}$)

$$\begin{aligned} \boldsymbol{a}_s = \arg\min_{\boldsymbol{a}_s} \frac{1}{2} \left\| \mathcal{Y}_s - \left[\!\left[\mathcal{G}; \boldsymbol{A}^{(1)}, \boldsymbol{A}^{(2)}, \ldots, \boldsymbol{A}^{(N-1)}, \boldsymbol{a}_s \right]\!\right] \right\|_2^2 + \lambda \left\| \boldsymbol{a}_s \right\|_1 \\ \text{subject to } \boldsymbol{a}_s \geq 0 \end{aligned} \tag{3.47}$$

where $\lambda \in \mathbb{R}$ is a regularization parameter

4: **save the past information**

$$\boldsymbol{\alpha}_s^{(n)} = \boldsymbol{\alpha}_{s-1}^{(n)} + \boldsymbol{Z}_{s,(n)} \boldsymbol{Z}_{s,(n)}^T, \quad n = 1, \ldots, N-1 \tag{3.48}$$

$$\boldsymbol{\beta}_s^{(n)} = \boldsymbol{\beta}_{s-1}^{(n)} + \boldsymbol{Y}_{s,(n)} \boldsymbol{Z}_{s,(n)}^T, \quad n = 1, \ldots, N-1 \tag{3.49}$$

5: **repeat**

6: **Update of basis factors:**

7: **for** n=1,...,N-1 **do**

8: Calculate new $\boldsymbol{A}^{(n)}$ by *alternating* optimizations with all other factors fixed and starting with the latest $\boldsymbol{A}^{(n)}$

$$\begin{aligned} \boldsymbol{A}^{(n)} = \arg\min_{\boldsymbol{A}^{(n)}} \frac{1}{2s} \Big\{ \sum_{i=1}^{s} \Big[\operatorname{Tr}\left(\boldsymbol{Y}_{i,(n)}^T \boldsymbol{Y}_{i,(n)} \right) \Big] - 2 \operatorname{Tr}\left(\boldsymbol{A}^{(n)T} \boldsymbol{\beta}_s^{(n)} \right) \\ + \operatorname{Tr}\left(\boldsymbol{A}^{(n)T} \boldsymbol{A}^{(n)} \boldsymbol{\alpha}_s^{(n)} \right) \Big\} \\ \text{subject to } \boldsymbol{A}^{(n)} \geq 0 \end{aligned} \tag{3.50}$$

9: **end for**

10: **until** sufficient convergence ($\Delta_{\text{error}} < 10^{-6}$)

11: **end for**

12: **return** factor matrices $\boldsymbol{A}^{(1)}, \boldsymbol{A}^{(2)}, \ldots, \boldsymbol{A}^{(N-1)}$

we propose an approximation to avoid the calculation of the sum. We first rewrite the Tucker operator $[\![\cdot]\!]$ in Equation 3.54 using the equivalence in Equation 3.11 and shorten the equations by using the notation

$$\{\boldsymbol{A}^{(k)}\}_{k=N-1,k\neq n}^{\otimes 1} \equiv \boldsymbol{A}^{N-1} \otimes \ldots \otimes \boldsymbol{A}^{(n+1)} \otimes \boldsymbol{A}^{(n-1)} \otimes \ldots \otimes \boldsymbol{A}^{(1)}. \tag{3.53}$$

We then rewrite the L_2-norm by using the trace $\mathrm{Tr}(\cdot)$ (Equation 3.55).

$$\boldsymbol{A}_t^{(n)} = \underset{\boldsymbol{A}^{(n)}}{\arg\min} \frac{1}{2I_N} \sum_{i=1}^{t} \left\| \boldsymbol{\mathcal{Y}}_i - \left[\!\left[\boldsymbol{\mathcal{G}}; \boldsymbol{A}^{(1)}, \boldsymbol{A}^{(2)}, \ldots, \boldsymbol{A}^{(N-1)}, \boldsymbol{a}_i \right]\!\right] \right\|_2^2 = \tag{3.54}$$

$$= \underset{\boldsymbol{A}^{(n)}}{\arg\min} \frac{1}{2I_N} \sum_{i=1}^{t} \left\| \boldsymbol{Y}_{i,(n)} - \boldsymbol{A}^{(n)} \boldsymbol{G}_{(n)} \left(\boldsymbol{a}_i \otimes \{\boldsymbol{A}^{(k)}\}_{k=N-1,k\neq n}^{\otimes 1} \right)^T \right\|_2^2 = \tag{3.55}$$

$$\begin{aligned} = \underset{\boldsymbol{A}^{(n)}}{\arg\min} \frac{1}{2I_N} \sum_{i=1}^{t} \Bigg[& \mathrm{Tr}\left(\boldsymbol{Y}_{i,(n)}^T \boldsymbol{Y}_{i,(n)} \right) \\ & - 2\,\mathrm{Tr}\left(\boldsymbol{A}^{(n)T} \boldsymbol{Y}_{i,(n)} \left(\boldsymbol{a}_i \otimes \{\boldsymbol{A}^{(k)}\}_{k=N-1,k\neq n}^{\otimes 1} \right) \boldsymbol{G}_{(n)}^T \right) \\ & + \mathrm{Tr}\Bigg(\boldsymbol{A}^{(n)T} \boldsymbol{A}^{(n)} \boldsymbol{G}_{(n)} \left(\boldsymbol{a}_i \otimes \{\boldsymbol{A}^{(k)}\}_{k=N-1,k\neq n}^{\otimes 1} \right)^T \\ & \left(\boldsymbol{a}_i \otimes \{\boldsymbol{A}^{(k)}\}_{k=N-1,k\neq n}^{\otimes 1} \right) \boldsymbol{G}_{(n)}^T \Bigg) \Bigg] \end{aligned} \tag{3.56}$$

After rewriting the optimization problem in this way we are now replacing the sum over the i in Equation 3.56 by iterative variables of the form

$$\boldsymbol{\alpha}_t^{(n)} = \boldsymbol{\alpha}_{t-1}^{(n)} + \boldsymbol{Z}_{t,(n)} \boldsymbol{Z}_{t,(n)}^T \quad \in \mathbb{R}_{0+}^{R\times R} \tag{3.57}$$

$$\boldsymbol{\beta}_t^{(n)} = \boldsymbol{\beta}_{t-1}^{(n)} + \boldsymbol{Y}_{t,(n)} \boldsymbol{Z}_{t,(n)}^T \quad \in \mathbb{R}_{0+}^{I_n\times R} \tag{3.58}$$

with

$$\boldsymbol{Z}_{t,(n)} = \boldsymbol{G}_{(n)} \left(\boldsymbol{a}_t \otimes \{\boldsymbol{A}^{(k)}\}_{k=N-1,k\neq n}^{\otimes 1} \right)^T \in \mathbb{R}_{0+}^{R\times I_1 \cdots I_{n-1} I_{n+1} \cdots I_{N-1}} \tag{3.59}$$

so that Equation 3.56 simplifies the optimization problem to the form used in the algorithm

$$\begin{aligned} \boldsymbol{A}_t^{(n)} = \underset{\boldsymbol{A}^{(n)}}{\arg\min} \frac{1}{2I_N} \sum_{i=1}^{t} & \left\{ \mathrm{Tr}\left(\boldsymbol{Y}_{i,(n)}^T \boldsymbol{Y}_{i,(n)} \right) \right\} - 2\,\mathrm{Tr}\left(\boldsymbol{A}^{(n)T} \boldsymbol{\beta}_t^{(n)} \right) \\ & + \mathrm{Tr}\left(\boldsymbol{A}^{(n)T} \boldsymbol{A}^{(n)} \boldsymbol{\alpha}_t^{(n)} \right). \end{aligned} \tag{3.60}$$

By doing so we commit an error $\Delta^{(n)}$ between Equation 3.56 and Equation 3.60, as we implicitly transform $\{\boldsymbol{A}_{t-1}^{(k)}\}_{k=N-1,k\neq n}^{\otimes 1}$ to $\{\boldsymbol{A}_{i}^{(k)}\}_{k=N-1,k\neq n}^{\otimes 1}$ in Equation 3.56 by introducing α and β. The error committed is

$$\begin{aligned}\Delta^{(n)} = \sum_{i=1}^{t} \operatorname{Tr}\Bigg(& \boldsymbol{A}^{(n)T}\boldsymbol{A}^{(n)}\boldsymbol{G}_{(n)}\left(\boldsymbol{a}_i^T\boldsymbol{a}_i\right) \\ & \otimes \left(\left\{\boldsymbol{A}^{(k)T}\boldsymbol{A}^{(k)}\right\}_{\substack{k=N-1,\\ k\neq n}}^{\otimes 1} - \left\{\boldsymbol{A}_i^{(k)T}\boldsymbol{A}_i^{(k)}\right\}_{\substack{k=N-1,\\ k\neq n}}^{\otimes 1}\right)\boldsymbol{G}_{(n)}^T\Bigg) \\ & - 2\operatorname{Tr}\left(\boldsymbol{A}^{(n)T}\boldsymbol{Y}_{i,(n)}\boldsymbol{a}_i \otimes \left(\left\{\boldsymbol{A}^{(k)}\right\}_{\substack{k=N-1,\\ k\neq n}}^{\otimes 1} - \left\{\boldsymbol{A}_i^{(k)}\right\}_{\substack{k=N-1,\\ k\neq n}}^{\otimes 1}\right)\boldsymbol{G}_{(n)}^T\right)\end{aligned} \tag{3.61}$$

This replacement allows the update of the factor matrices for the whole set of tensor without knowing the whole set of tensor from the beginning. The information from the "past" tensor slices is stored in the variables $\boldsymbol{\alpha}_t^{(n)}$ and $\boldsymbol{\beta}_t^{(n)}$ with $n = 1, \ldots, N-1$.

3.4.1. Implementation

To implement our algorithm we use a multiplicative, positivity preserving update as proposed by Lee and Seung [1999] with additional sparseness constraint and factor normalization. We chose this kind of implementation for its simplicity in order to obtain quickly a prove of our approach's concept.

Considering a cost function $C(\theta)$ of non-negative variables θ the multiplicative update has the form

$$\theta_i \leftarrow \theta_i \bullet \left(\frac{\frac{\partial C(\theta)^-}{\partial \theta_i}}{\frac{\partial C(\theta)^+}{\partial \theta_i}}\right) \tag{3.62}$$

with $\frac{\partial C(\theta)^+}{\partial \theta_i}$ the positive and $\frac{\partial C(\theta)^-}{\partial \theta_i}$ the negative part of the derivative with respect to θ_i. The complete derivative of the cost function $C(\theta)$ is $\frac{\partial C(\theta)}{\partial \theta_i} = \frac{\partial C(\theta)^+}{\partial \theta_i} + \frac{\partial C(\theta)^-}{\partial \theta_i}$.

This multiplicative update rule can be explained as follows: In the case that the gradient is zero, i.e. $\frac{\partial C(\theta)^+}{\partial \theta_i} = \frac{\partial C(\theta)^-}{\partial \theta_i}$, θ_i remains unchanged. In the case that the gradient is positive, i.e. $\frac{\partial C(\theta)^+}{\partial \theta_i} > \frac{\partial C(\theta)^-}{\partial \theta_i}$, the update rule will decrease the entries of θ_i and vice versa in the case of a negative gradient.

The derivatives of Equation 3.60 are

$$\frac{\partial}{\partial \boldsymbol{A}^{(n)}} \operatorname{Tr}\left(\boldsymbol{A}^{(n)T}\boldsymbol{A}^{(n)}\boldsymbol{\alpha}_t^{(n)}\right) = 2\boldsymbol{A}^{(n)}\boldsymbol{\alpha}_t^{(n)} \tag{3.63}$$

$$\frac{\partial}{\partial \boldsymbol{A}^{(n)}} \operatorname{Tr}\left(\boldsymbol{A}^{(n)T}\boldsymbol{\beta}_t^{(n)}\right) = \boldsymbol{\beta}_t^{(n)} \tag{3.64}$$

and the resulting update rule for $\boldsymbol{a}_t$ (coding step) is

$$\boldsymbol{a}_t \leftarrow \boldsymbol{a}_t \bullet \left(\frac{\boldsymbol{Y}_{t,(N)}\boldsymbol{Z}_{t,(N)}^T}{\boldsymbol{a}_t\boldsymbol{Z}_{t,(N)}\boldsymbol{Z}_{t,(N)}^T + \lambda \left\|\boldsymbol{a}_t\right\|_1}\right) \tag{3.65}$$

and the update rule for the factor matrices $\boldsymbol{A}^{(n)}$ is

$$\boldsymbol{A}^{(n)} \leftarrow \boldsymbol{A}^{(n)} \bullet \left(\frac{\boldsymbol{\beta}_t^{(n)}}{\boldsymbol{A}^{(n)}\boldsymbol{\alpha}_t^{(n)}}\right) \tag{3.66}$$

At the beginning the factor matrices are initialized at random with each entry in the range $[0, 1]$. The iterative variables $\alpha_t^{(n)}, \beta_t^{(n)}$ are initialized 0. We also require all entries of the input tensor to be in the same range $[0, 1]$.

3.4.2. Factor normalization

If applying a sparseness constraint to only one factor matrix but not all, normalization of the remaining factors is essential to avoid growing of the non constrained factors. A cost function of the form

$$C = \left\|\mathcal{Y} - [\![\mathcal{G}; \boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}]\!]\right\|_F^2 + \lambda \left\|\boldsymbol{C}\right\|_1 \tag{3.67}$$

could be decreased by simply shrinking the entries of $\boldsymbol{C}$ and growing the factors $\boldsymbol{A}$ and $\boldsymbol{B}$ that do not obey the sparseness constraint.

To avoid this effect different kinds of normalization can be applied to the remaining factors. When using a multiplicative update rule like Lee and Seung a simple min-max normalization, i.e. renormalization to a maximum value of 1 and a minimum value of 0, would create values that are exactly zero. These values would, in consequence of the multiplicative nature of the update, not change any more. Common solutions are normalization of all columns of the factors to a sum of one

$$\boldsymbol{A}_t^{(n)} \leftarrow \boldsymbol{A}^{(n)} \bullet \left(\frac{\boldsymbol{\beta}_t^{(n)}}{\boldsymbol{A}^{(n)}\boldsymbol{\alpha}_t^{(n)}}\right) \bullet \mathbb{1}\left(\frac{1}{\boldsymbol{A}^{(n)}} \bullet \left(\frac{\boldsymbol{A}^{(n)}\boldsymbol{\alpha}_t^{(n)}}{\boldsymbol{\beta}_t^{(n)}}\right)\right) \tag{3.68}$$

or normalization of each column to a Frobenius norm of one

$$\boldsymbol{A}_t^{(n)} \leftarrow \boldsymbol{A}^{(n)} \bullet \left(\frac{\boldsymbol{\beta}_t^{(n)}}{\boldsymbol{A}^{(n)} \boldsymbol{\alpha}_t^{(n)}} \right) \bullet \left\{ \mathbb{1} \left(\frac{1}{\boldsymbol{A}^{(n)}} \bullet \left(\frac{\boldsymbol{A}^{(n)} \boldsymbol{\alpha}_t^{(n)}}{\boldsymbol{\beta}_t^{(n)}} \right) \right)^{\cdot 2} \right\}^{\cdot \frac{1}{2}} . \tag{3.69}$$

These are the normalizing update rules to be applied in the coding step.

3.4.3. Memory consumption

The approach to decompose a N-dimensional data tensor as a series of $N-1$-dimensional data tensors allows to keep less data in memory.

Intending to decompose a set of tensors $\{\mathcal{Y}_t \in \mathbb{R}_{0+}^{I_1 \times \ldots \times I_{N-1}}, t = 1, \ldots, I_N\}$ which is equivalent to the decomposition of a tensor $\mathcal{Y} \in \mathbb{R}_{0+}^{I_1 \times I_2 \times \ldots \times I_n \times \ldots \times I_N}$ with $I_N \geq I_{N-1} \geq \ldots \geq I_1 \geq R \geq N \geq 2$, where R is the reduced dimension, the proposed algorithm has a memory consumption of

$$M = 2 \prod_{i=1}^{N-1} I_i + R \prod_{i=1}^{N-2} I_i + R \sum_{i=1}^{N-1} I_i + R \sum_{i=1}^{N} I_i + (N-1)R^2 \tag{3.70}$$

matrix entries. The first summand is the space required to keep the example $\mathcal{Y}_t$ and its reconstruction in memory, the second for a $\boldsymbol{Z}_{(n)}, n \neq N$, the third for the $\boldsymbol{\beta}^{(n)}$ and the last for the $\alpha^{(n)}$.

In the case of a order 3 tensor this is: $M = 2R^2 + R(I_1 I_2 + 2I_1 + 2I_2 + I_3) + 2I_1 I_2$.

The memory consumption of the NsTsF algorithm is quadratic in R. Supposing cubic tensors to be decomposed, i.e. $I_n = I_m = I, \forall n, m$, we can also determine the memory consumption of the algorithm to be proportional to $I^{(N-1)}$ and exponential in N.

3.4.4. Algorithmic cost

The numerical complexity can help to compare the performance of different algorithms. For the proposed algorithm the computational cost for the two steps are different. The coding step has a computational cost of

$$\mathcal{O}\left((2R+1) \prod_{n=1}^{N-1} I_n + R \right) \tag{3.71}$$

and the basis update step

$$\mathcal{O}\left(R \sum_{i=1}^{N-1} I_n + (N-1)(2R+1) \prod_{n=1}^{N-1} I_n \right). \tag{3.72}$$

For an order 3 tensor, this cost is: $\mathcal{O}((2R+1)I_1I_2 + J)$ for the coding step and $\mathcal{O}(2(2R+1)I_1I_2 + R(I_1+I_2))$ for the basis update step.

The cost of the ordinary matrix product $\boldsymbol{AB}$, with $\boldsymbol{A} \in \mathbb{R}^{I\times J}$ and $\boldsymbol{B} \in \mathbb{R}^{J\times K}$ is assumed to be $\mathcal{O}(IJK)$. The computational cost of the element-wise product $\boldsymbol{A} \bullet \boldsymbol{B}$ is $\mathcal{O}(IJ)$.

The algorithmic cost of both basis update and coding step are linear in R. Supposing the tensor to decompose to be cubic the algorithmic cost of both steps is of the order $I^{(N-1)}$ in I and exponential in N.

3.4.5. Further properties

The factors $\boldsymbol{a}_t$ that are available after decomposition should be updated before the assessment of reconstruction error. Especially the factors of the first $\boldsymbol{a}_t$'s are not perfectly adopted for the final basis factors $\boldsymbol{A}^{(1)}, \ldots, \boldsymbol{A}^{(N-1)}$ because they were optimized for an early version of the basis factors.

With this decomposition approach we have the possibility to start the decomposition of a tensor *before* all of its sub-tensors $\mathcal{Y}_t$ are know. In cases where the whole tensor $\mathcal{Y}$ we want to decompose is known from the beginning we can present the sub-tensors in arbitrary order. This simple change of the order makes the algorithm more robust to decomposition problems originating from a strong order in the sub-tensors, e.g. the sub-tensors belong to several different classes of examples.

Our algorithm decomposes the data tensor along the largest dimension. By reindexing the tensor we can decompose also along every other direction using our algorithm. In terms of algorithmic cost and memory consumption it might be of no advantage to decompose along any other dimension but the larges one, but there might be circumstances that make it necessary to choose this route.

3.4.6. Performance evaluation and comparison

Often the L_2-error is used to compare the decomposition quality using different parameters or algorithms. The L_2-error however constitutes an *absolute* error which depends on the number of entries of the tensors.

$$C_{\text{abs.}} = \left\| \mathcal{Y} - \hat{\mathcal{Y}} \right\|_2 \tag{3.73}$$

In our evaluation we also want to investigate the influence of the size and the order of the tensor that is to be decomposed on the reconstruction error. Decomposing

different tensors the L_2-error will not allow a comparison as the L_2-error changes with the number of tensor elements and their scale. Therefore, we will use instead the *relative* L_2-error

$$C_{\text{rel.}} = \frac{\left\| \mathcal{Y} - \hat{\mathcal{Y}} \right\|_2}{\|\mathcal{Y}\|_2}, \tag{3.74}$$

where $\mathcal{Y}$ is the tensor to be decomposed and $\hat{\mathcal{Y}}$ the reconstructed tensor. This error measure is independent of size and scale of the tensor to be decomposed and allows therefore the comparison of results obtained for different input data.

Algorithms and Data used

For comparison a selection of 4 algorithms will be used: The original NMF algorithm by Lee and Seung [2000], the direct extension of the same algorithm to NTD by Mørup et al. [2008] (higher order non-negative matrix factorization (HONMF)), a preconditioned nonlinear conjugate gradient (cgP) algorithm [Royer et al., 2011] and a alternating non-negative least squares (ANLS) algorithm with block principal pivoting (BPP) (ANLS-BPP) [Kim and Park, 2012]. All algorithms are used in a pure Matlab® implementation. NMF is a 2-way method, i.e. designed for 2^{nd} order tensors, therefore the examples have to be vectorized when decomposing a 3rd order tensor. The preconditioned nonlinear conjugate gradient (cgP) is an implementation specialized to 3rd order tensors, consequently a vectorization becomes necessary if the order of the input tensor exceeds 3.

To evaluate the performance of the proposed algorithm we will use the same data set as in section 2.4. The face images of the CBCL database present a typical decomposition task and have been used in the evaluation of several publications dealing with matrix and tensor factorizations [Friedlander and Hatz, 2008, Hazan et al., 2005, Hoyer and Dayan, 2004, Lee and Seung, 1999].

This data set forms a 3rd order tensor of dimensions $19 \times 19 \times 2429$ which we will use to analyze the performance of our new algorithm and to compare it to established algorithms. As the data set contains several images of the same person we have to assure by shuffling their order that they are not grouped together in the data tensor. To also perform tests on a 4^{th} order tensor we selected 211 persons with at least 6 images available and formed a 4^{th} order tensor of dimensions $19 \times 19 \times 6 \times 211$, where the first two ways correspond to the pixels, the third indexes the different images of one person and the fourth way corresponds to the different persons.

Performance evaluation

We will first examine the response of the algorithm to the parameters sparseness, rank R of the decomposition and the method of factor normalization by decomposing the 3rd and 4^{th} order data tensor formed from CBCL images. Afterwards we compare the performance of $\text{N}_\text{S}\text{T}_\text{S}\text{F}$ with the selection of algorithms introduced before.

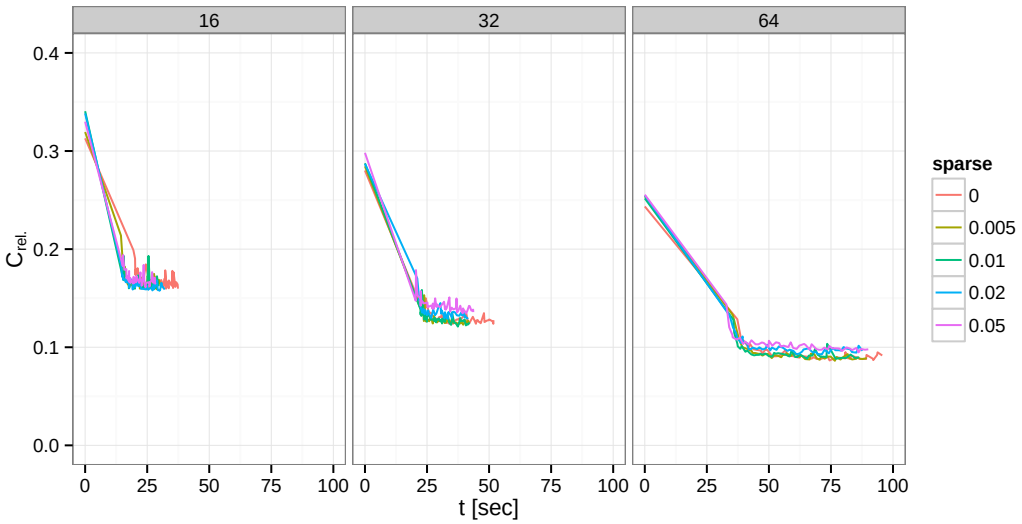
The reconstruction error plot in Figure 3.6a for a decomposition without factor normalization shows that omitting factor normalization can, but not necessarily must, cause problems in the decomposition. In this case the algorithm showed instabilities only in the case of $R = 16$. Repeating the same decomposition with a normalization of the factors columns to $\sum_j \boldsymbol{A}_{ij}^{(n)} = 1$, i.e. all columns of the factor matrices sum up to one, we do not experience any instabilities (see Figure 3.6b). The factor normalization to $\sum_j {\boldsymbol{A}^{(n)}}_{ij}^2 = 1$ as implemented by Mørup shows slightly higher stability then the normalization to $\sum_j \boldsymbol{A}_{ij}^{(n)} = 1$.

We compare the methods of factor normalization using the same regularization parameters λ for sparseness as before at a fixed basis dimension of $R = 32$ in Figure 3.7. This comparison shows that the factor normalization does not influence the final reconstruction error of the algorithm. However decomposition time is significantly lower (by approx. 25 seconds) with normalization to $\sum_j \boldsymbol{A}_{ij}^{(n)} = 1$ than with the method of Mørup et al. [2008]. The normalization of the factors to a sum of squares of one ($\sum_j {\boldsymbol{A}^{(n)}}_{ij}^2 = 1$) by our direct implementation failed totally and is not shown.

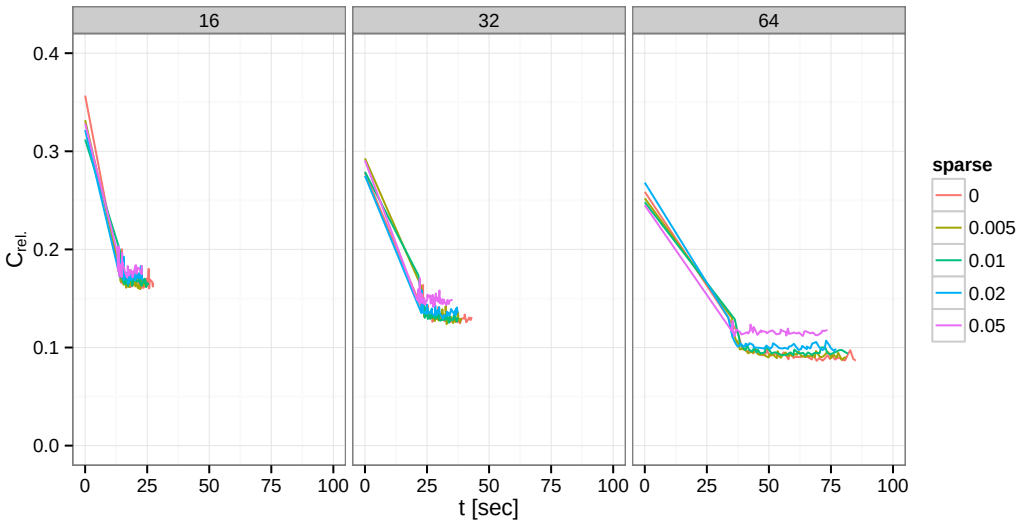
The evolution of the reconstruction error when varying the rank R of the decomposition is as expected. The relative error decreases with increasing R while the computation time increases. For decreasing R we observe opposite behavior. When varying the regularization parameter for sparseness λ we observe no significant change in the convergence behavior of the algorithm. The reconstruction error has however a minimum for $0.005 < \lambda < 0.01$ and is most influenced when applying a normalization of the factors to $\sum_j \boldsymbol{A}_{ij}^{(n)} = 1$.

Reconstructed face images from decomposition by $\text{N}_\text{S}\text{T}_\text{S}\text{F}$ are shown alongside with reconstructions of the other algorithms in Figure 3.10. The reconstruction basis is as interesting as the reconstruction. In the connection with face images this basis is often called the Eigenfaces when decomposing with PCA. In the case of our non-negative algorithm these images are no Eigenfaces so we will call them *basis faces*. The basis faces obtained in the previous test by $\text{N}_\text{S}\text{T}_\text{S}\text{F}$ are visualized in Figure 3.8. Besides some basis images that are harder to interpret, the majority of the basis images correspond to nose, mouth chin, cheeks, eyes and eye-brows.

For further testing using CBCL face images we use the 4^{th} order tensor of dimensions $19 \times 19 \times 6 \times 211$. The tensor is decomposed in its original 4^{th} order shape and with

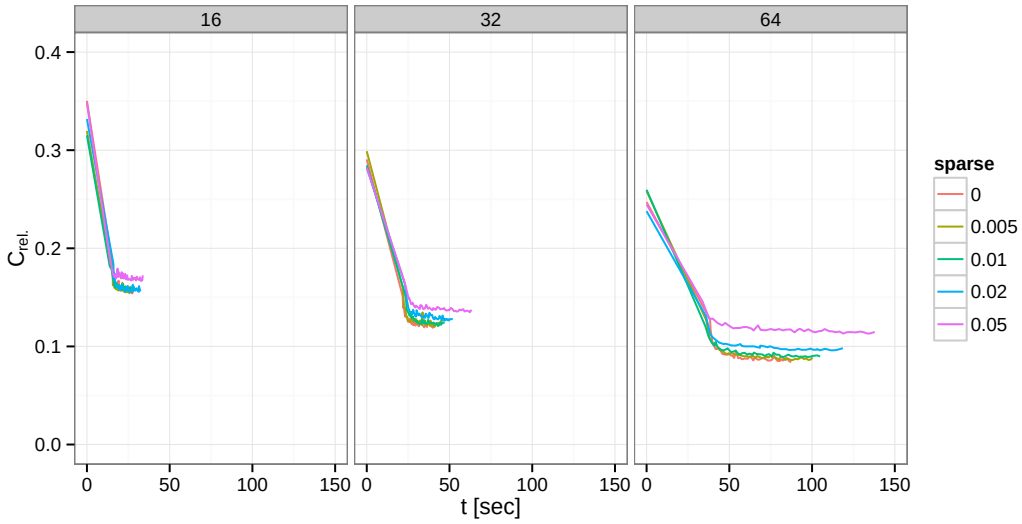


(a) Without factor normalization.



(b) The used factor normalization is $\sum_j \boldsymbol{A}_{ij}^{(n)} = 1$.

Figure 3.6.: Comparison of error development in decompositions with different ranks $R = 16, 32$ and 64. Relative error $C_{\text{rel.}}$ plotted over the time.



(c) The used factor normalization is proposed by Mørup.

Figure 3.6.: Comparison of decompositions with $R = 16, 32$ and 64.

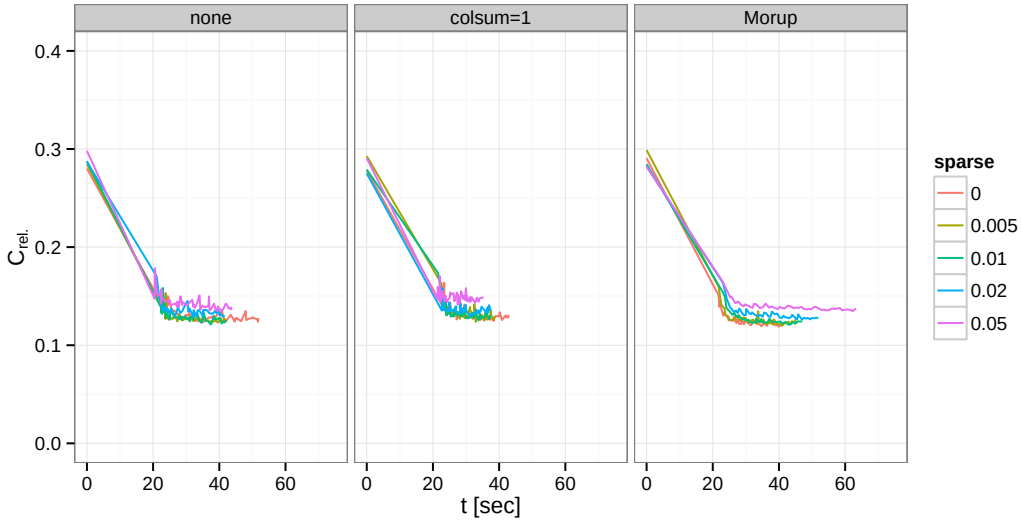


Figure 3.7.: Comparison of factor normalization techniques: none, $\sum_j \boldsymbol{A}_{ij}^{(n)} = 1$, Mørup. $R = 32$. The vertical line marks where the data was completely presented for the first time. The data point at $t = 0$ corresponds to $C_{\text{rel.}}$ obtained with random initialized basis factors and only the coding factor optimized.

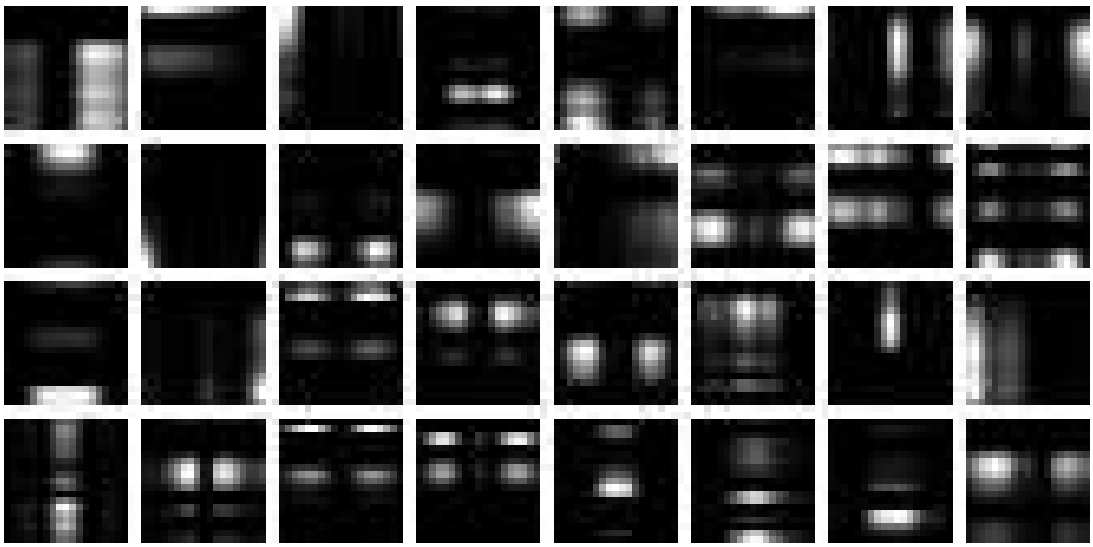


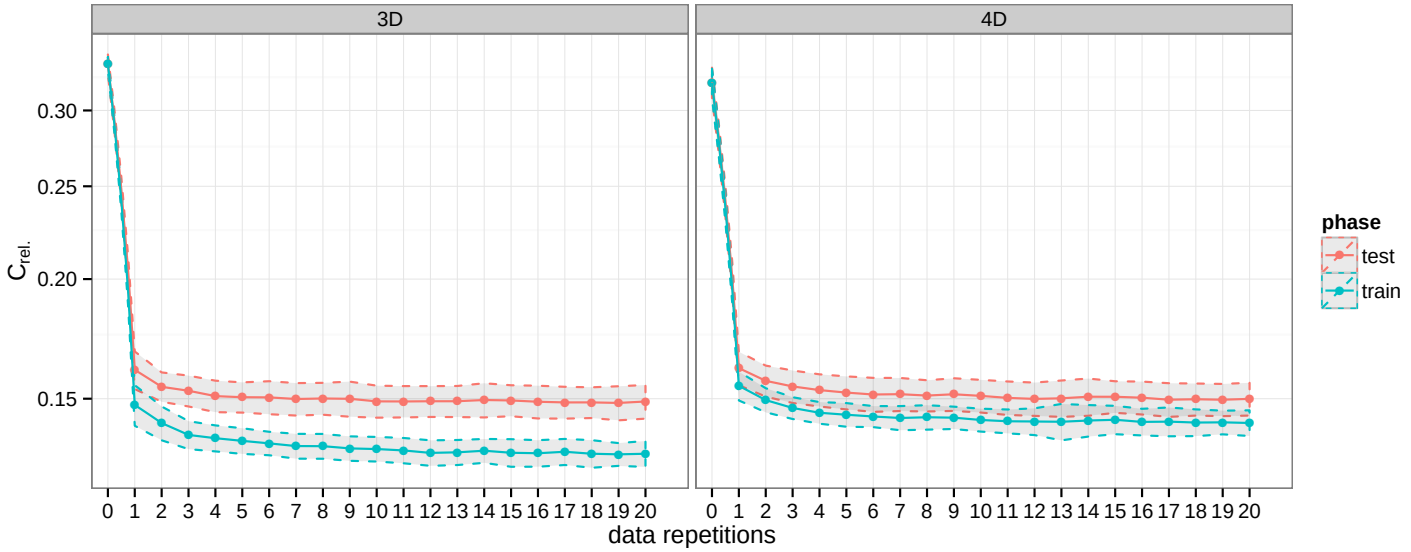
Figure 3.8.: 32 basis faces obtained by N_STsF decomposing 2429 CBCL face images. Images corresponding to nose (3^{rd} row, 7^{th} column), mouth (1^{st} row, 4^{th} column), chin (3^{rd} row, 1^{st} column), cheeks (3^{rd} row, 5^{th} column), eyes, eye-brows or combinations can be found. Others are harder to interpret.

each image vectorized, i.e. as a 3^{rd} order tensor of dimensions $361 \times 6 \times 211$. As the special structure of the N_STsF algorithm imposes that the number of iterations is equal to the number of presented examples, in the case of the tensor used in this test 211, we were interested in question whether a repeated presentation of examples in randomized order would enhance decomposition accuracy.

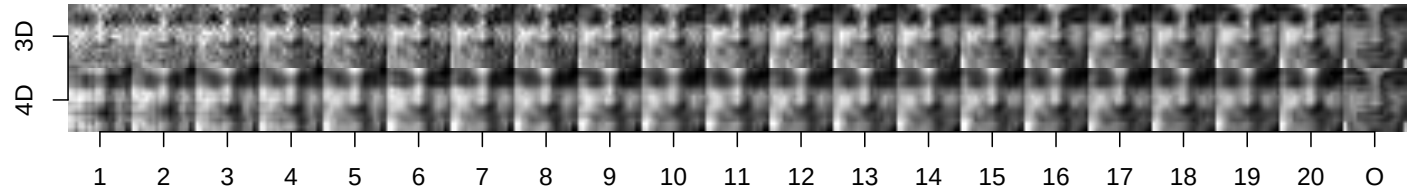
To learn in the same experiment about the generalization capability of the calculated basis factors a bootstrap scheme was applied. A bootstrap sample was drawn out of the 211 persons. On this sample the basis factors where calculated (training) and the reconstruction error on the sample measured. Afterwards the ooB examples were sparse coded in the just obtained basis factors (testing) and again the reconstruction error measured. This procedure was repeated 50 times. Usually 100 or more repetitions should be performed, but the results for 50 repetitions already had a acceptable standard deviation consuming less computation time. For basis factors that are generalizing well the motive of the image only a small difference between the error in training and testing will be obtained.

Figure 3.9 plots the results of this experiment. We observe that the repeated presentation of the examples helps to obtain a better reconstruction error in both cases, the 3^{rd} order tensor and the 4^{th} order tensor. However the gain is higher for the 3^{rd} order

tensor. The curve representing the test error is always higher than the curve representing the training error as expected. In the case of the 4^{th} order tensor the difference is always in the order of one standard deviation. In the case we vectorized to a 3rd order tensor the difference between the training and the testing reconstruction error is two standard deviation for the first two repetitions and grows to three standard deviations subsequently. This growth of difference in reconstruction error shows that further repetitions of the data are not capable to significantly improve the generalization of the learned basis factors.



(a) L2-error with standard deviation per image.



(b) Reconstruction of testing images of a person with average L2-error.

Figure 3.9.: Decomposition of tensors formed out of the CBCL face data set. The sub-tensors associated with the different persons are repeated in a random order. Error rates are calculated by 50 fold .632 bootstrap. (a) is a plot of the relative error over the data repetitions. At repetition zero the error of a reconstruction with random basis factors was measured. (b) shows a corresponding reconstructed image followed by the original image (O).

Performance comparison

Using the same input data we compared N_STsF with other NTF algorithms. Figure 3.10 shows 5 different examples of reconstruction selected over the whole range of the tensor as well as decomposition time and relative error of the decomposition. The visual appearance of the reconstructed face image corresponds to the relative reconstruction error. NMF shows the best reconstruction closely followed by ANLS-BPP and N_STsF.

time	-	31sec	48sec	34min	5h30	2min30
rel. error	-	0.1042	0.1180	0.1791	0.2679	0.1120
$\frac{\text{rel. error}}{\text{image}}$	-	0.11 ± 0.04	0.13 ± 0.05	0.14 ± 0.05	0.28 ± 0.09	0.12 ± 0.05

Figure 3.10.: Reconstruction after decomposition into 32 basis images; relative L_2-error is given as error $\pm$ standard deviation. Computation time raising from left to right.

In figure 3.11 we plot the evolution of the relative error for the selection of NTF algorithms over time during the decomposition. This plot shows that the ANLS-BPP algorithm is, decomposing the example data, as fast as the N_STsF algorithm but stopped later due to the applied stopping criterion.

This first test shows that the proposed algorithm and NMF, as well as ANLS-BPP, are comparable in reconstruction error and time consumption on this simple task, while

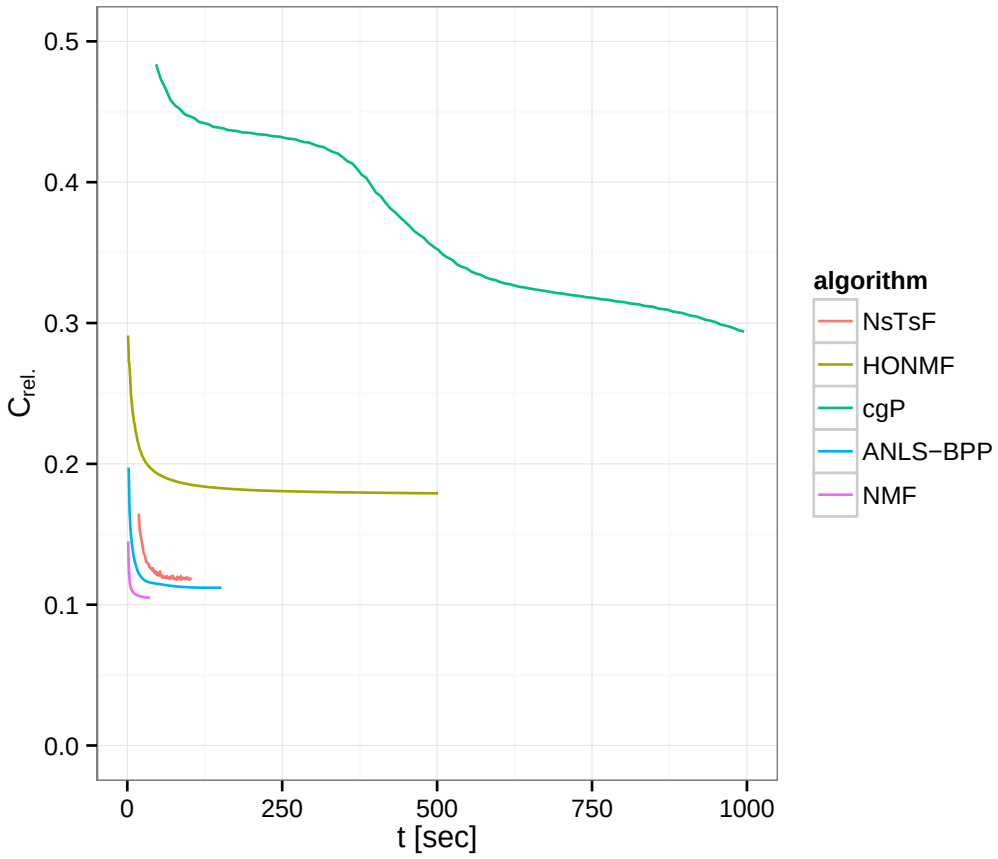


Figure 3.11.: Evolution of relative error over time for several NTF algorithms when decomposing 3rd order CBCL face image tensor. Basis images $R = 32$. Time in seconds.

HONMF and cgP take significant time to calculate a decomposition with higher reconstruction error. In visual examination of the reconstructed images shown in figure 3.10 the reconstruction provided by NMF is found to be slightly superior while there is no significant difference between those provided by NsTsF and ANLS-BPP.

3.5. Discussion

One shortcoming of our algorithm is that an approximation is necessary to derive the basis update step (see Equation 3.61). Unfortunately we were not (yet) able to

establish upper bounds for this error. However, the numeric examples showed that the error of our algorithm is comparable with the error of state of the art algorithms. We ascribe this to the coding step which does not need an approximation and thus can equilibrate the committed error.

Another way to approach large scale decompositions are block wise approaches, e.g. Kim and Park [2012] and Cichocki et al. [2009, section 1.3.1]. In these approaches parts of the input tensor are selected randomly to be processed. $N_S T_S F$ on the contrary does process parts of the tensor in a regular way. One might suspect that this leads to a better representation of the information processed last in the decomposition but as the past information is stored in the basis update step we did not observe any dependence of the reconstruction error on the location of the data in the input tensor what so ever. Additionally $N_S T_S F$ avoids the need to access the same part of the tensor multiple times which considerably reduces memory overhead and contributes to the speed of the algorithm. Furthermore, block wise algorithms need the parts of the tensor to be selected according to an uniform distribution in order to obtain a decomposition with uniform reconstruction error over the whole tensor. In consequence additional precautions are necessary to be able to start the decomposition of a tensor before all entries of the tensor are known. With our $N_S T_S F$ algorithms it is possible to add new examples at any time and the algorithm will incorporate the new information available.

After finishing the presented implementation we also investigated whether it is possible to transfer our decomposition approach directly to an implementation using the Kullback-Leibler divergence as cost criterion. The use of the Kullback-Leibler divergence, however, did not allow us to derive the iterative variables $\alpha_t^{(n)}$ and $\beta_t^{(n)}$, which are essential to circumvent the complete recalculation of the sum in the cost function (Equation 3.52) in every update.

3.6. Summary

In this chapter we presented a novel algorithm for NTF and sparse NTF adapted to the decomposition of tensors order ≥ 3 with one large dimension. Algorithms for NTF presented in the past were often restricted in the size of tensors that can be decomposed. Algorithms designed to overcome this size restriction, for example based on block wise decomposition, require a frequent access to partitions of the whole data of the tensor. The presented algorithm on the contrary requires only a minimum of data access and is even capable to start a decomposition before the whole tensor is know. In comparative tests the algorithm has proved to be competitive with state of the art algorithms.

Part II.

Application: Early detection of Alzheimer's disease

4. About Alzheimer's disease and FDG PET

In 2010 there are worldwide 35.6 million people estimated to be affected by dementia. With the growth of the elder population this number is expected to raise to about 115 million by 2050 while there is no reliable early diagnosis method and effective treatment at hand. The most common cause of dementia is Alzheimer's disease (AD). Dementias do not only affect the patient himself but also his social environment. In many cases relatives cannot handle daily life without external help. Hence, dementias also charge health care systems with huge amounts of money (estimated to 604 billion US dollars in 2010) Wimo and Prince [2010].

The World Alzheimer Report 2011 Prince et al. [2011] is focused on early diagnosis and early treatment of Alzheimer's disease (AD). It states that more than 50 % of cases of dementia remain undiagnosed in high income countries and therefore do not receive appropriate medical treatment. According to this report early diagnosis allows patients to maintain higher quality of life due to better possibilities of treatment.

With positron emission tomography (PET), nuclear medicine provides a non-invasive, three-dimensional functional imaging method that measures the metabolism in the brain. The scans used in this work use the tracer fluorodeoxyglucose (FDG) that accumulates in the brain cells according to their metabolic rate to do so. AD typically causes bilateral hypo-metabolism in the temporal and parietal lobes, posterior cingulate gyri and precunei, as well as frontal cortex that is imaged by the PET scan [Perani and Cappa, 2001]. Hypo-metabolism has been correlated to dementia severity by comparing measures of cognitive function, such as Mini-mental State Exam (MMSE) and Clinical Dementia Rating (CDR), and cerebral metabolic rate of glucose [Braak and Braak, 1991, Hoffman et al., 2000].

4.1. Medical background

In 1906 Alois Alzheimer described in a short communication the case of a 56 years old patient suffering from dementia. After pathological examination Alzheimer states that the patient was suffering from a disease that was not described so far. He presented his findings at a conference of psychiatrists in Tübingen, Germany, in November 1906 under the title "On the peculiar disease process of the cerebral cortex" (Über eine eigenartige Erkrankung der Hirnrinde; Alzheimer 1906). In a textbook of Kreapelin,

appearing in 1910, the described disease is first named Alzheimer's disease. In 1911 Alzheimer published a comprehensive article [Alzheimer, 1911] describing clinic and pathology of the disease. This very article is included in English translation in Möller and Graeber [1998] which revised Alzheimer's findings.

Today several non-Alzheimer's dementias are known and have to be considered in differential diagnosis. In the modern conception of AD the case presented by Alzheimer in his 1911 publication is of the "plaque-only" AD type as there were no intracellular neurofibrillary tangles (NFT) found.

4.1.1. Clinical diagnosis

In 1984 recommendations for the clinical diagnosis of AD were published by a common work group of NINCDS and ADRDA [McKhann et al., 1984]. In this recommendations, that should stay unrevised for more than 25 years, biomarkers[1] took, due the early stage of development, only a secondary role. In 2011 revised recommendations for the diagnosis of AD were published. Three separate publications for AD [McKhann et al., 2011], mild cognitive impairment (MCI) due to AD [Albert et al., 2011] and preclinical stages of AD [Sperling et al., 2011] provide new guidelines for AD diagnosis and research. These articles take advances of research concerning AD and other dementia of the past decades into account. Diagnosis is still based primarily on clinical examination, but biomarkers underwent significant valorization especially in the diagnosis of mild cognitive impairment (MCI) and preclinic AD. Biomarkers are separated into two major groups: biomarkers of amyloid β (Aβ) accumulation and biomarkers of neuronal degeneration or injury. The first comprise abnormality in amyloid PET imaging and low cerebrospinal fluid (CSF) Aβ_{42}, the latter elevated CSF tau, decreased FDG uptake on PET in a specific topographic pattern involving temporoparietal cortex and atrophy on structural magnetic resonance - again in a specific topographic pattern - involving medial, basal, and lateral temporal lobes and medial and lateral parietal cortices [Jack et al., 2011]. All three publications agree that a further validation of these biomarkers is necessary to integrate them in standard clinic diagnosis.

Diagnosis of AD usually starts with memory complaints of the patient himself or, more typical for AD dementia, with complaints by a relative realizing memory loss or problems in daily activities of the patient. Neurological assessment will be used by the clinician to determine whether the patient's memory declines as part of normal aging or if the patient suffers from a dementia disease. One of the first test that are usually performed is the Mini-mental State Exam (MMSE). It consists of 30 questions examining orientation, registration, attention and calculation, recall, language and praxis.

[1] The term biomarker will be used, though slightly counter intuitive, for both fluid analytes and imaging measures.

The resulting score is sensitive to dementia typic memory decline. Due to the fixed questionnaire and evaluation scheme the MMSE can also be performed by nurses or social workers. An other dementia score, the CDR, is in contrary based on a more subjective experience of the patient by a trained medical doctor. CDR ranges from 0 to 3, where 0 is non-demented and 3 severe demented. A CDR of 0.5 is called MCI which is characterized by very mild memory problems that are not sufficient to conclude the patient to be affected by AD. After the diagnosis of a present dementia the patient is diagnosed probable AD after excluding other causes of dementia like depression, fronto-temporal dementia (FTD), dementia with Lewy bodies (DLB), Parkinson's disease with dementia (PDD), vascular dementia (VaD) and others. In the differential diagnosis the patients medical history and present medication is examined [Gauthier, 2006, chapter 4 to 6]. magnetic resonance imaging (MRI), or, if not available or contra-indicated, computed tomography (CT) can be used to examine possible brain lesions and atrophy [Gauthier, 2006, chapter 7]. Due to high cost and expertise needed for functional MRI, single photon emission computed tomography (SPECT) and PET, these methods are only in major hospitals routinely applied in the diagnosis of AD.

The progress of AD is staged by the medical examiner taking into account the results of neurophysiological testing, family history and education. In table 4.1 we try an approximate comparison of the neurological scores for dementia mentioned. As the comparison table shows MMSE is not capable to differentiate between normal control group (NC) and MCI.

Label	CDR	MMSE	Symptoms
NC	0	>24	none
MCI	0.5	>24	very mild
AD	1	21-26	mild
AD	2	10-20	moderate
AD	3	≤ 9	severe

Table 4.1.: Approximate correspondence between the different scales for neurological assessment of dementia.

4.1.2. PET in the diagnosis of Alzheimer's disease

PET is a tracer based tomographic imaging method. That means in order to obtain the image the patient has to be administered a radioactive substance, the so called tracer, that accumulates predominantly in the areas that are desired to be visualized. In the case of PET imaging the tracer is a positron emitter. The positrons are produced in a β-decay of the tracer substance. The positrons are detected via the two photons

emitted in the positron-electron annihilation $e^+ + e^- \rightarrow 2\gamma$. The angle between the emitted photons is $(180 \pm 3)°$. The variation originates from the positrons momentum. Due to the movement of the positron before annihilation and the deviation from the 180° angle the maximum physical possible resolution of a PET scanner is $2 - 3$mm depending on the positron energy.

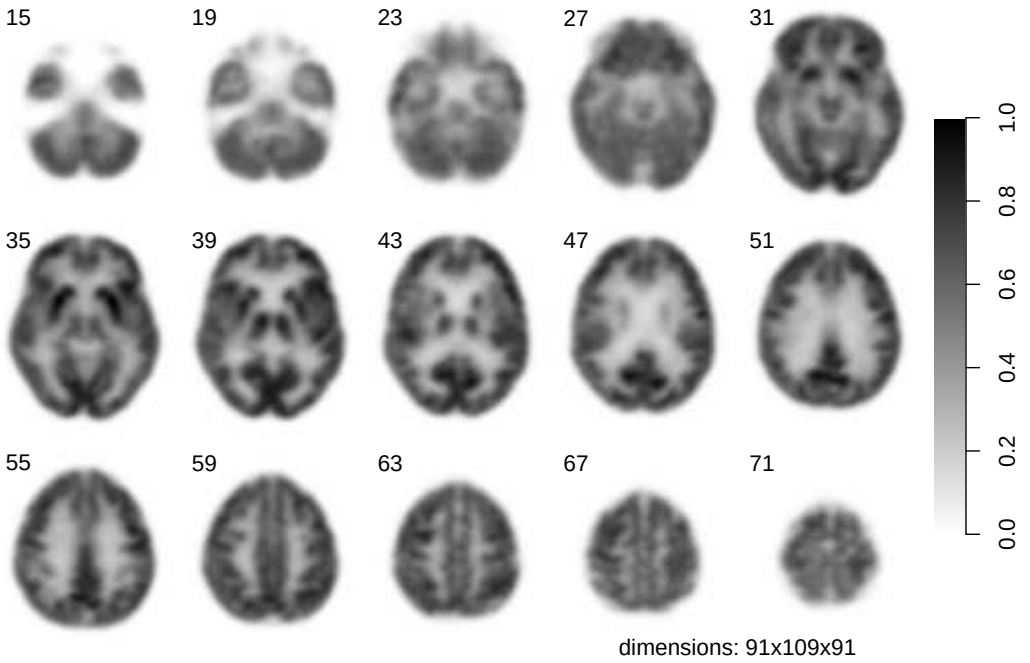
The capabilities of PET to provide an image of blood flow and metabolism of the brain is used by the clinician to differentiate AD from other dementias, such as FTD, VaD and DLB [Ercoli and Small, 2009][Gauthier, 2006, chapter 8]. To discriminate those entities the clinician searches for disease typic or atypic patterns of hypo-metabolism. Therefore the reading of a PET scan in this context remains a difficult and time consuming task.

Usually PET scans are aligned to the AC-PC line prior to visual inspection. This rigid body transformation facilitates the orientation in the brain and comparison with other imaging modalities or patients. Grayscale, inverse gray-scale and diverse color scales are used for the reading of the scans. While gray-scale representation inhibits the advantage of the linearity of the scale, color scales can emphasize small metabolic differences. Only few computational utilities are used, such as contrast adjustment, tools for quantitative FDG uptake measurement as well as tools for volumetric measurement [Silverman, 2009, chapter 2].

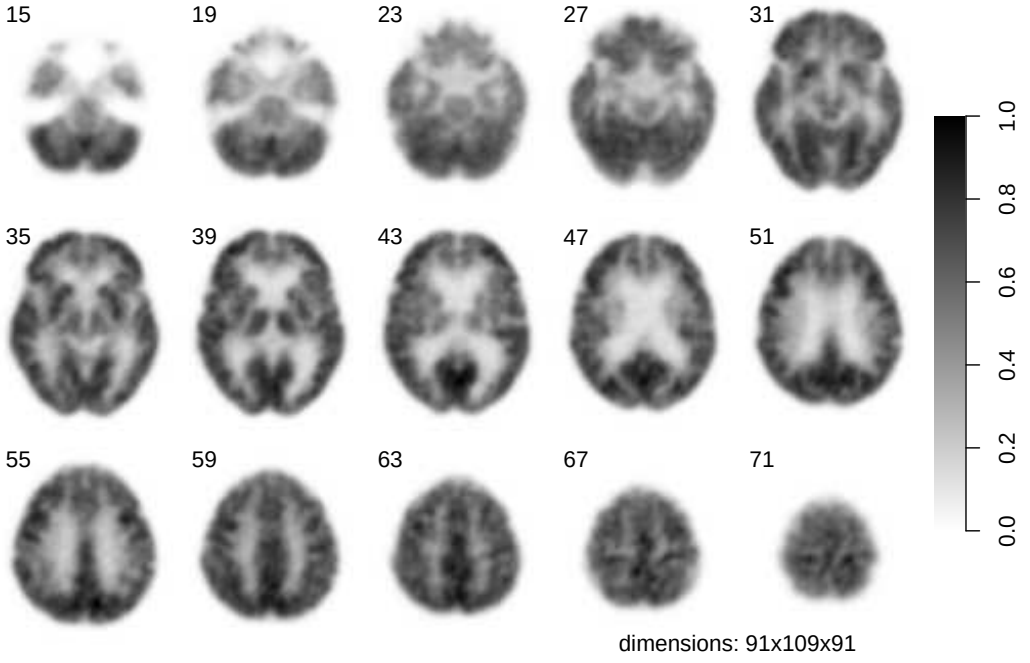
Example FDG PET scans of NC and AD subjects using inverse gray-scale are shown in figure 4.1 (page 65). At first sight we have the impression that the AD example image has a lower over all intensity. But this is not sufficient to deduce a dementia because there are other possible explanations for this difference: *1)* The two patients might have a different level of glucose metabolism and therefore a different FDG uptake. *2)* The images are normalized differently (which is not the case here).

In order to conclude from PET scans to the presence of AD the brain has to be examined for regions that have an FDG uptake that is below normal level compared to other regions in the brain that are usually not affected by AD. Furthermore the scan has to be examined for signs of other diseases that might as well cause the patients symptoms. Ishii [2002] reviews clinical differential diagnosis by FDG PET using statistical parametric mapping (SPM) and 3D stereotactic surface projection (3D-SSP).

The direct detection of the AD biomarkers Aβ and NFT using the radioligand Pittsburgh compound B (PIB) is object of fundamental research. First trials on humans have been published by Klunk et al. [2004]. Due to its very short half-live time of ≈ 20 minutes and the complexity of its production, the handling of this radioligand is very complicated and costly. Therefore, its use is restricted to a small number of research facilities.



(a) Normal control group: CDR=0, MMSE=29



(b) Alzheimer's disease: CDR=1, MMSE=20

Figure 4.1.: Transversal slices of FDG PET scans after alignment to the AC-PC line and intensity normalization. At first sight one remarks a reduced intensity in the AD image but this difference can have various other explanations. A detailed analysis is necessary.

4.2. State of the art in computerized analysis of PET in the diagnosis of Alzheimer's disease

After a series of articles appeared in the 1990's proving statistical difference between NC and AD PET scans using several different approaches, a focus on single patient classification started in the following years. This, by no means exhaustive, summary of publications depicts an overview over the advances of the last decade in this specific field. Unfortunately the obtained classification results are not absolutely comparable, as it is not in all cases clear at which stage of the disease the used scans were acquired and some publications use populations of only 50 subjects or less.

Herholz et al. [2002] report a classification rate of 84 % for very mild AD using a voxel wise analysis based on t-values. Scarmeas et al. [2004] complemented a voxel wise t-value analysis with a scaled subprofile model (SSM) of 45 predefined regions of interest (ROIs) and report a sensitivity of 76 % at a specificity of 63 % using the presented methodology. Nobili et al. [2008] present a classification based on the combination of a principal component analysis (PCA) of PET scans and a clinical memory score obtained from a selective reminding test (SRT). The accuracy achieved for a classification MCI versus MCI to AD converters is at 80-90 %. Using the information obtained from the PET scans only the classification rate is approximately at 80 %. Using a voxel wise approach combined with the regional distributions of gray matter (GM), white matter (WM) and CSF applied to both MRI and PET of MCI patients Fan et al. [2008] claim a classification rate of 93 %. Markiewicz et al. [2009] implement a PCA approach with Fisher discriminant analysis (FDA) classifier. They use re-sampling techniques to asses the robustness of their methods to misclassified labels and obtain a classification rate of 95 % of a group of normal controls and AD patients. Illán et al. [2011] express the PET scans as linear combinations of the voxel wise within class average of the images to reduce the dimensionality. The images are then classified by a support vector machine (SVM) attaining a classification rate of 88 %. Herholz et al. [2011] present a PET score based on t-values [Herholz et al., 2002] sensible to the progression of normal aging to MCI and to AD. Coupé et al. [2011] perform a segmentation and gradings of hippocampus and entorhinal cortex by means of a supervised algorithm called scoring by non-local image patch estimator (SNIPE). The achieved classification rate is at 90 %.

Though not providing a diagnosis of single patients Drzezga et al. [2003] shows a group difference between patients with stable MCI and patients that converted from MCI to AD within a 1 year follow-up period. To establish the difference a two-sample t-test and analysis of variance (ANOVA) was used.

4.3. Tools for PET analysis

There are several software tools with capabilities for the analysis of PET data available. The only proprietary software used in the preparation of this thesis was Matlab® . This section will introduce the utilities that have been used for data preparation and analysis. Particularly, it will give an introduction to brain coordinate systems and templates, and to the Matlab® toolboxes SPM and WFU PickAtlas.

4.3.1. Brain coordinates and templates

To be able to describe a position in the brain a Cartesian coordinate system has to be defined. The usual convention is that the anterior positions in the brain, the forehead, corresponds to $+y$-coordinates and posterior positions, the back of the head, to $-y$-coordinates. Superior positions (upper part of the brain) correspond to $+z$-coordinates and inferior positions to $-z$-coordinates respectively. For the right hemisphere $+x$-coordinates are used and for the left hemisphere $-y$-coordinates.

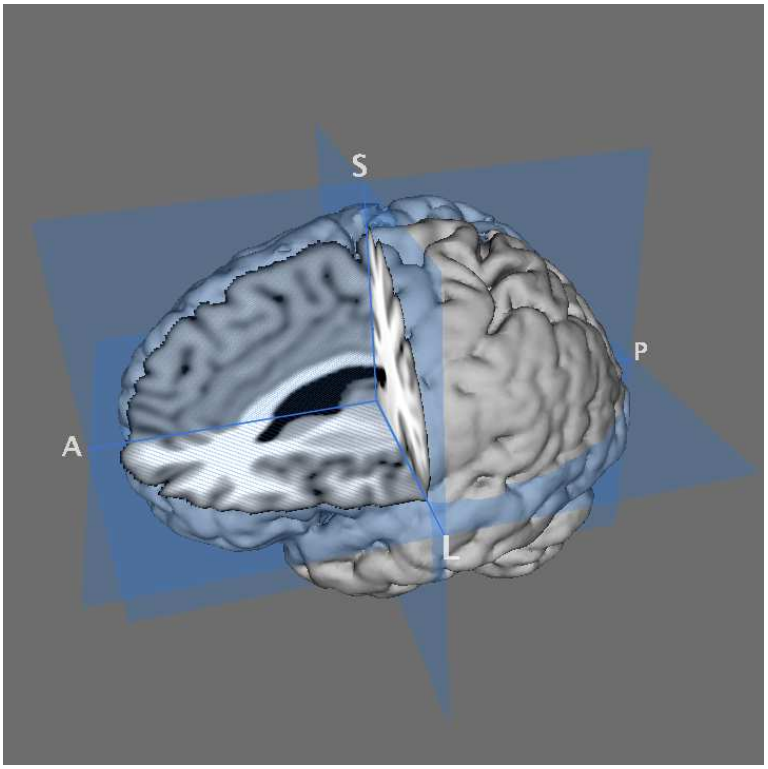


Figure 4.2.: The brain coordinate system. A: anterior, P: posterior, L: left hemisphere, S: superior. A transversal, coronal and sagittal plane is shown.

Three planes through the brain are possible. xy-planes are called transversal planes, xz-planes are called coronal planes and yz-planes are called sagittal planes. These planes are visualized in Figure 4.2. In the remainder of this thesis usually a representation as transversal planes is used to visualize the brain. The brain is oriented with anterior to the top of the figure and inferior slices first.

In 1988 Jean Talairach and Pierre Tournoux published their Co-Planar Stereotaxic Atlas of the Human Brain. This atlas comprises the photographies of the dissected brain of one person, sketches of the dissections and tomographic images. The pictures are annotated with the names of the brain tissues shown. Furthermore important landmarks in the brain like the line connecting anterior commissure and posterior commissure (AC-PC), the vertico-frontal line (VCA line), a line traversing the posterior margin of the anterior commissure vertically and the inter-hemispheric plane (midline). Those landmarks are used to achieve a uniform orientation of different brain images. The volume spanned by the Talairach space has dimensions $170 \times 200 \times 210$mm. Unfortunately it origins from one single female brain and thus is rather small and not representative of the population. The Talairach Daemon [Lancaster et al., 1997] is a software that allows the conversion of coordinates in Talairach space to brain tissue labels and vice versa. This brain atlas is useful to identify the different brain tissues in clinical use. The Talairach Daemon allows even the non-specialist to retrieve the name of the tissue corresponding to a given location in the brain.

The current brain template of the Montreal Neurological Institute (MNI) and the International Consortium for Brain Mapping (ICBM) is the ICBM152. This template is the average of 152 normal MRI scans of young persons that have been registered to the older MNI305 template by a 9 parameter affine transformation. The ICBM152 is also the standard template in SPM starting with version SPM99 (see figure 4.3).

The MNI305 template is the average of 305 normal MRI scans of young persons. Of those scans 250 have been registered manually to the Talairach atlas using landmarks. The remaining 55 scans have been registered to the 250 scans by a automatic 9 parameter affine transformation. Details are described in Evans et al. [1993].

The SPM PET template is the average of images from 12 subjects spatially normalized to ICBM152. Images were first registered to the T1-weighted MRI images, and spatially transformed using the same transformation. Images were acquired on a Siemens ECAT HR+ at the FIL, using Oxygen-15 labeled water. The averaged images are smoothed using 8mm full width at half maximum (FWHM) Gaussian. This PET template is used to perform a spatial normalization of the images.

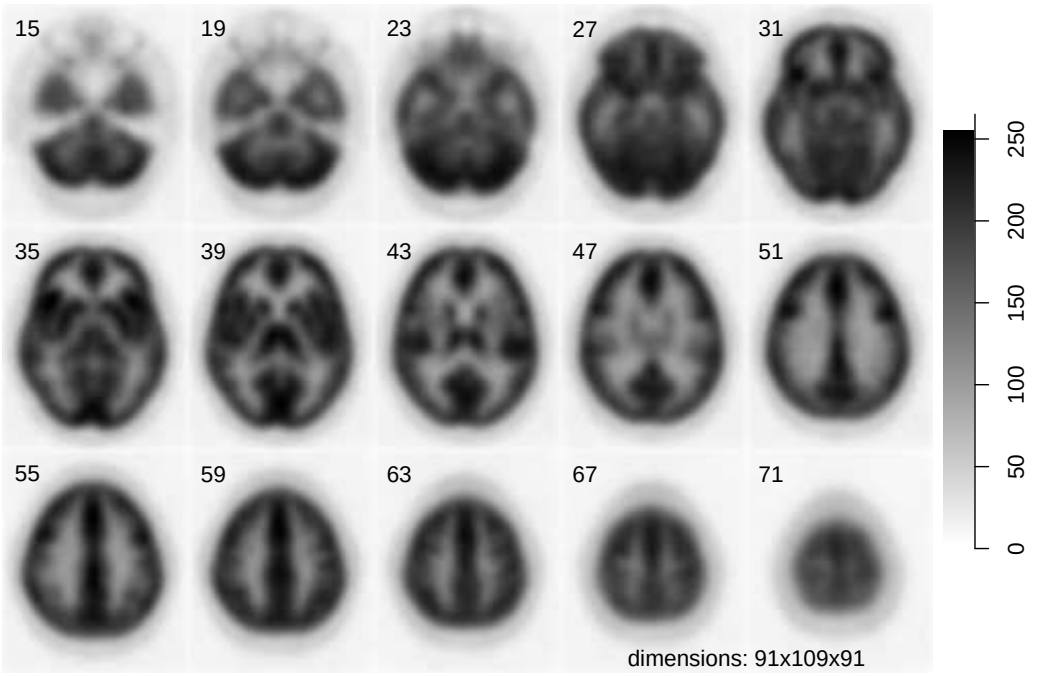


Figure 4.3.: Transversal slices of the ICBM152 brain template as shipped with SPM99 and later. Slices are numbered from bottom to top of the brain. Plotted on a linear gray scale. Averaging over many scans makes the image blurry.

4.3.2. Statistical parametric mapping (SPM)

This Matlab® toolbox allows the user to pre-process and analyze three types of data: PET & SPECT data, MRI & functional magnetic resonance imaging (fMRI) data and electroencephalography (EEG) data. The pre-processing tools for PET allow a spacial pre-processing of the data.

There exist several tools for the spatial normalization of brain scans. The probably most spread tools are automated image registration (AIR) developed by M.D. Roger P. Woods, NEUROSTAT developed by the Departments of Radiology and Bioengineering, Washington National Primate Research Center, University of Washington, Seattle, WA, U.S.A. and SPM developed by by members & collaborators of the Wellcome Trust Centre for Neuroimaging.

Kiebel et al. [1997] compare SPM and AIR in the co-registration of simulated PET images to MRI images. Both algorithms perform the task robust and precise. Ishii et al. [2001] compared SPM and NEUROSTAT in the real life situation of co-registration of

FDG PET images of AD patients to a brain template. They focused the analysis on probable artifacts when standardizing the atrophied brains. The results of their experiments show deformation artifacts with both methods and a considerable difference in location of the peak hypo-metabolism as well as a change in extent and severity between the two methods. Neither of the studies shows considerable advantages of one of the three utilities. It was decided to use SPM as it is an open source Matlab® implementation and therefore easy to modify if necessary.

The most important tools for pre-processing included in SPM are the realign, the co-registration and the normalization tool. Besides these three tools there exist also tools for smoothing and the segmentation of gray matter, white matter and cerebro-spinal fluid. As the only tool used in the preparation of our data set is the normalize tool, we will provide a short description of this tool and refer for further information to the manual of SPM which is available at `http://www.fil.ion.ucl.ac.uk/spm/`.

Normalise is a tool to *spatially* normalize scans to a template. It rotates and stretches the brain to match the template. To do so the tool calculates first the optimum 12-parameter affine transformation and then estimates non-linear deformations, whereby the deformations are defined by a linear combination of three dimensional discrete cosine transform (DCT) basis functions. Spatial normalization is used to establish voxel wise comparability of different scans and also of different patients.

While the spatial normalization allows a comparative analysis of a set of scans it also introduces a source of error in the analysis. The performed transformation is deforming the brain and the deformation models are based on assumptions that might hold or not. But even with a deformation model based on correct assumptions the deformation might cause a degradation of image quality or a loss of the information that is contained in the specific shapes. However, this risk has to be taken in order to be able to compare the scans of different individuals.

The brain templates included in the SPM toolbox are conform to the space defined by the ICBM, National Institutes of Health (NIH) P-20 project and approximate that of the space described in the atlas of Talairach and Tournoux [1988]. For more details about the template see section 4.3.1.

4.3.3. WFU PickAtlas

The Matlab® toolbox WFU PickAtlas makes it possible to generate ROI maps based on the Talairach daemon database (see Lancaster et al. [2000]). This daemon includes hemispheres, lobes, Brodmann areas, anatomic labels and tissue types. Tissue types include gray matter, white matter, cerebro spinal fluid and anatomic labels include the most common names that are used to describe a certain brain region. The atlases have been corrected and extended to the vertex in MNI space (see Maldjian et al. [2004]).

This tool makes it possible to select desired zones by their name, combine them and save them as indexed mask in MNI space.

5. Image set for Alzheimer's disease early recognition

The basis for the training of a performant classifier is a proper selection of the input data. It has to represent the variety of the problem and has to be as clean from false labeled data as possible. Moreover the amount of available data plays also an important role. Small numbers of examples or unbalanced datasets make specialized analysis techniques and more sophisticated evaluation necessary.

The needs in a clinical study differ slightly from the needs of a computer based image analysis. For visual analysis a spatial normalization is not necessary and a unified orientation of all scans is not obligatory but simplifies the analysis. For a computer based analysis this is absolutely necessary. The choice of the color or gray scale is important for visual examination, as explained in Silverman [2009, chap. 2], whereas computer analysis does not rely on the scale but on a stable intensity normalization. Therefore, the images have to be specifically prepared and selected for computerized analysis.

5.1. ADNI-Database

The Alzheimer's Disease Neuroimaging Initiative (ADNI) was launched in 2003 by the National Institute on Aging (NIA), the National Institute of Biomedical Imaging and Bioengineering (NIBIB), the Food and Drug Administration (FDA), private pharmaceutical companies and non-profit organizations, as a $60 million, 5-year public-private partnership. The primary goal of ADNI has been to test whether serial MRI, PET, other biological markers, and clinical and neuropsychological assessment can be combined to measure the progression of MCI and early AD. Determination of sensitive and specific markers of very early AD progression is intended to aid researchers and clinicians to develop new treatments and monitor their effectiveness, as well as lessen the time and cost of clinical trials.

The Principal Investigator of this initiative is Michael W. Weiner, M.D., VA Medical Center and University of California - San Francisco. ADNI is the result of efforts of many co-investigators from a broad range of academic institutions and private corporations, and subjects have been recruited from over 50 sites across the U.S. and Canada. The initial goal of ADNI was to recruit 800 adults, ages 55 to 90, to participate in the

research – approximately 200 cognitively normal older individuals to be followed for 3 years, 400 people with MCI to be followed for 3 years, and 200 people with early AD to be followed for 2 years. For up-to-date information see `www.adni-info.org`.

The images of the ADNI database are available via internet and can be downloaded for brain scans common file formats without charge[1]. Both raw scans and preprocessed scans are available for download. Furthermore, the meta data containing information about the patient and the scan itself are obtainable via the ADNI homepage.

5.1.1. Key eligibility criteria

All participants of the study must be between 55-90 (inclusive) years of age and fulfill several criteria for general health. In the following only the criteria that are of interest for the scan selection are described and a complete list of all criteria can be found in the ADNI Procedures Manual available on the ADNI homepage. The selection of the participators class, NC, MCI or AD, is based on several examinations of the cognitive functions: The MMSE, the CDR and interviews with the patients itself and one of its relatives. The results must achieve several conditions for admittance of the patient to the study. 200 patients each for both NC and AD and 400 patients for MCI constitute the population of the study.

- NC subjects: MMSE between 24-30 (inclusive), CDR of 0, non-depressed, non MCI, and non-demented.
- MCI subjects: MMSE between 24-30 (inclusive), a memory complaint, CDR of 0.5, absence of significant levels of impairment in other cognitive domains, essentially preserved activities of daily living, and an absence of dementia.
- AD subjects: MMSE between 20-26, CDR of 0.5 or 1, and meets NINCDS-ADRDA criteria for probable AD.

Patients fulfilling these criteria in the screening examination will undergo clinical examinations, MRI, FDG- and PIB-PET scans in fixed intervals. FDG-PET scans are taken for 50% of the study population and PIB-PET scans for 12% of the study population. The neuro-imaging timetable for the selected patients is shown in Table 5.1.

5.1.2. Raw PET image data

The scans in the ADNI database are administered in different sites with different types of scanners, but each site has passed a qualification test by the ADNI and the quality control is centralized. All PET scans in the ADNI database are acquired using one of three different protocols:

[1] registration is necessary

	Screening	Baseline	Month 6	Month 12	Month 18	Month 24	Month 36	Month 48
MRI (1.5T)	●■◆		●■◆	●■◆	●	■◆	●■	●■
MRI (3T)		●■◆	●■◆	●■◆	●	●■◆	●■	●■
PET		●■◆	●■◆	●■◆	●	●■◆	●■	

● NC ■ MCI ◆ AD

Table 5.1.: Timetable of the image acquisitions in the ADNI study. Brain imaging is performed according to this schedule. Screening is performed to determine whether the person meets the enrollment criteria. Baseline corresponds to month 0, the first examination in the program.

1. dynamic: 6 five-minute frames are acquired within 30 minutes starting 30 minutes after FDG injection and ending 60 minutes after FDG injection.
2. static: a single 30 minutes frame is acquired starting 30 minutes after FDG injection and ending 60 minutes after FDG injection.
3. quantitative: 33 frames are acquired starting at injection and continuing for 60 minutes. That can be used to calculate the absolute glucose metabolic rate from the radioisotope input function measured in the carotid arteries.

The majority of the scans in the ADNI database were acquired with the dynamic protocol. The static protocol is a backup protocol for those scanners that are not capable to acquire a dynamic scan.

5.1.3. Processed PET image data

To make these raw images more easily usable in a study the database provides also preprocessed scans. The goals of this preprocessing is to make the scans more uniform and to minimize the differences between the different scanner systems. The ADNI database provides four different types of processed data.

1. *Co-registered Dynamic*: Raw PET images from all acquisition sites are downloaded for quality control at University of Michigan. Raw images are converted to a standard file format. A single frames are extracted from the raw image file for registration purposes. Each extracted frame is co-registered to the first extracted frame of the raw image file. The base frame and the co-registered frames are recombined into a co-registered dynamic image set. These image sets have the same image size and voxel dimensions and remain in the same spatial orientation as the original PET image data. This is called "native" space. Logically only PET scans acquired under protocol 1 or 3, will have a processed image set of this type (see item 5.1.2). Example slices of one frame are shown in Figure 5.1a.

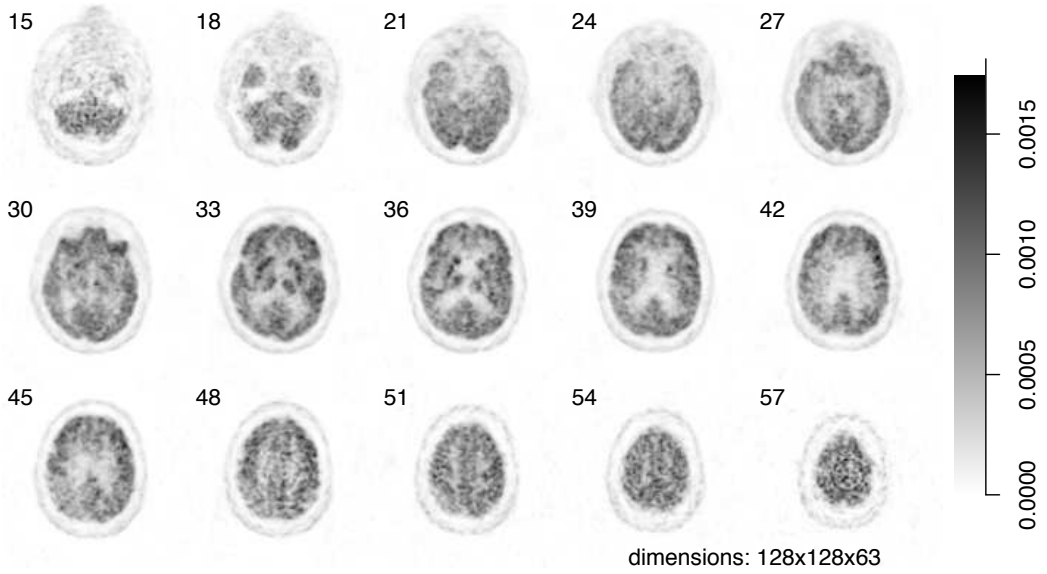
2. *Co-registered averaged*: This processed image set is generated simply by averaging the 6 five-minute frames (or the last 6 frames from the quantitative studies) of the "Co-registered Dynamic" image set described above. This creates a single 30 minute PET image. As above, this kind of processed data is only available for scans acquired under protocol 1 or 3 (see item 5.1.2). Example slices are shown in Figure 5.1b.

3. *Co-reg., Avg., Standardized Image and Voxel Size*: Each subject's *baseline* PET scan is then reoriented into a standard $160 \times 160 \times 96$ voxel image grid, having 1.5mm cubic voxels. This image grid is oriented such that the anterior-posterior axis of the subject is parallel to the AC-PC line. This is referred to as "AC-PC" space. This standardized image then serves as a *reference* image for all PET scans on that subject. By doing the co-registration from the original raw image data to a standardized space in a single step, only one interpolation of the image data is required, and thus resolution degradation by interpolation is kept to a minimum, and is the same for all scans. An average image is generated from the AC-PC co-registered frames and then intensity normalized using a subject-specific mask so that the average of voxels within the mask is exactly one. Both the spatial orientation (AC-PC) and the intensity normalization of the image are intended as a starting point for subsequent analyses. With a standardized image matrix, PET data from different scanner models can be compared more easily. It should be noted that in these images sets only spatial re-orientation and intensity normalization of scans has occurred. No non-linear warping or even linear scaling of the brains dimensions has been applied to the images. Example slices are shown in Figure 5.1c.

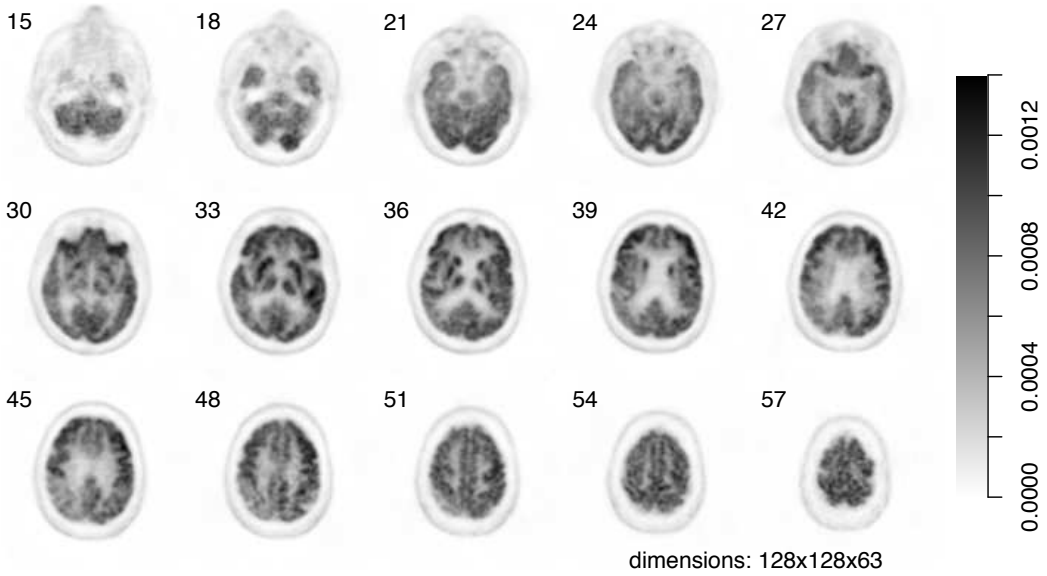
4. *Co-reg., Avg., Std. Image and Voxel Size, Uniform Resolution*: These images are the result of smoothing of the above-mentioned images. Each image set is filtered with a scanner-specific filter function to produce images of a uniform isotropic resolution of 8mm FWHM, the approximate resolution of the lowest resolution scanners used in ADNI. Image sets from higher resolution scanners obviously have been smoothed more than image sets from lower resolution scanners. The specific filter functions were determined from the Hoffman 3-D Brain Phantom PET scans that were acquired during the certification process. Example slices are shown in Figure 5.1d.

Examples of the preprocessing steps are shown in Figure 5.1. A scheme of the whole ADNI data preparation is shown in Figure 5.2. The numbers of included patients and available scans are shown in Table 5.2.

Comparing with other image analysis data sets a total of 404 patients and roughly 4 scans per subject might seem few. But high cost and ethic reasons limit the acquisition of PET scans. Compared to other PET and SPECT studies e.g. Drzezga et al. [2003],

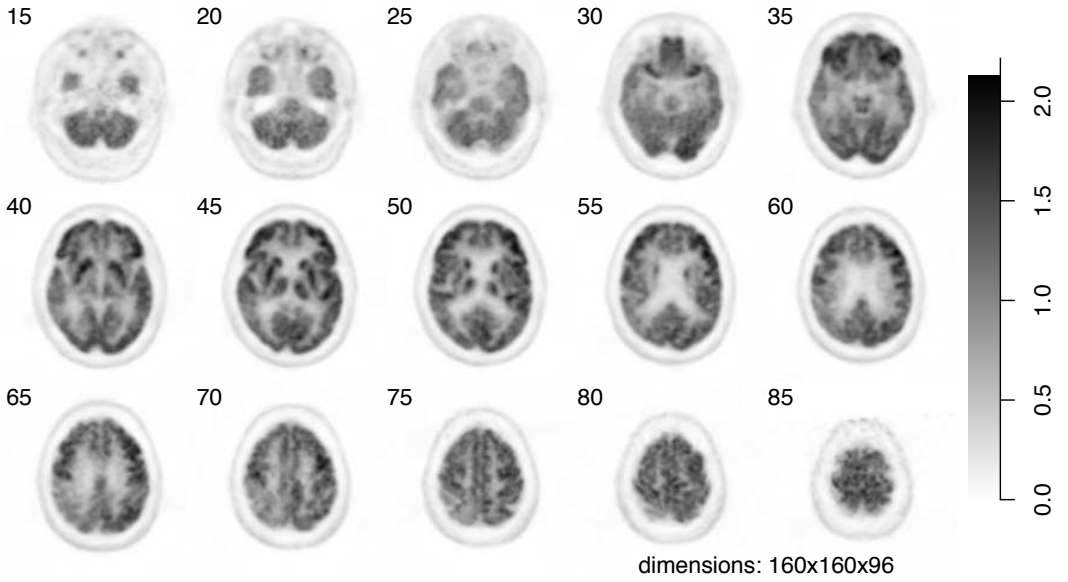


(a) co-registered, dynamic, one frame

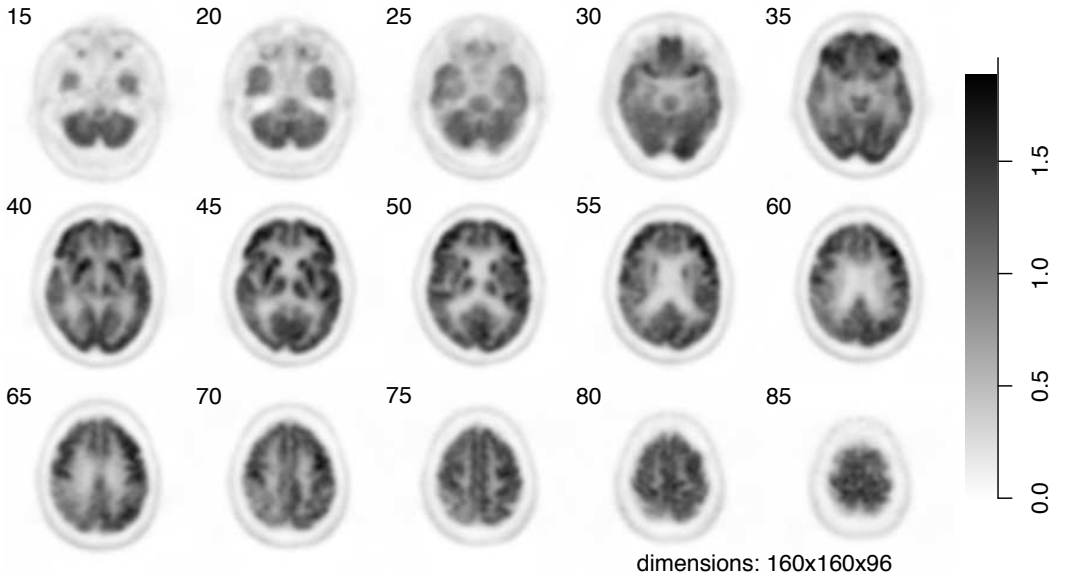


(b) co-registered, averaged

Figure 5.1.: Transversal slices of the preprocessing stages that are available in the ADNI database; Slices are numbered from bottom to top; Alzheimer's disease patient with CDR 2; asymmetric hypo metabolism is visible in the parietal lobe



(c) co-registered, averaged, standardized voxel and image size



(d) co-registered, averaged, standardized voxel and image size, uniform resolution

Figure 5.1.: Transversal slices of the preprocessing stages that are available in the ADNI database; Slices are numbered from bottom to top; Alzheimer's disease patient with CDR 2; asymmetric hypo metabolism is visible in the parietal lobe

	NC	MCI	AD	$\sum$
Patients	102	207	95	404
Scans	433	956	313	1702

Table 5.2.: FDG-PET preprocessed scans available form the ADNI database.

Ishii et al. [2001], Saxena et al. [1998] the ADNI provides a rather large number of images.

5.2. Needed properties for the intended methods

Image sets for computational analysis have to fulfill some additional criteria in comparison to visual analysis by physicians. Those criteria depend on the methods that are intended to be applied.

In this thesis we want to extract information that discriminates the healthy population form the population affected by AD directly from the data set. To be capable to do this extraction we have to compare the brains of the individuals and need therefore the best inter-brain comparability possible. As every brain is different a spatial normalization of the brain scans is necessary. Several tools for spatial normalization are available (see section 4.3).

In addition to spatial comparability we need also comparability on the intention domain. The PET raw data contain the reconstructed count rates for each voxel. So a priory there is no fixed range of voxel values. Ideally there would be a reference in each image that corresponds to the same metabolic activity that can be used for normalization. Unfortunately there is no such reference and intensity normalization has to assume a brain region that has the same metabolism in every person or be reference free. Usually the images are not normalized to 8bit unsigned integer $[0, 255]$ because the intensity resolution of the raw data is higher than 8bit. Original images obtained from the PET machine contains usually the count rate as voxel value and the range is not fixed.

A further important point in the preparation of a medical data set is the confidence in the class labels. For the diagnosis of early stage AD medical doctors estimate the diagnosis to be correct in about 85% of the cases. While modeling of measurement noise is an intensively studied field in classification modeling of label noise is not studied in big extent [Markiewicz et al., 2009, Quost and Denœux, 2009, Thiel, 2008].

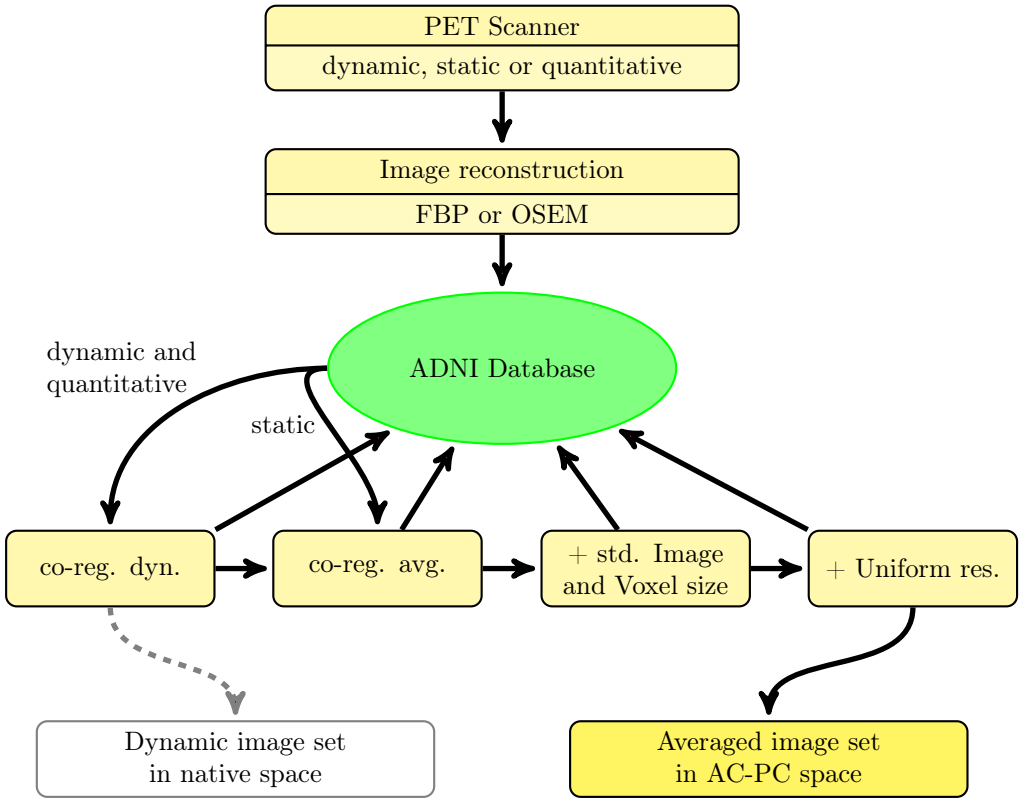


Figure 5.2.: Scheme of ADNI imaging workflow. Processing steps are displayed in yellow.

5.3. Existing image set: University of Granada (Spain)

One image set for computer analysis consisting of ADNI scans was created by a research group of the University in Granada Spain. Its images are

1. spatially normalized to the MNI space
2. have a $79 \times 95 \times 69$ voxel grid and
3. a voxel size of 2mm^3.

The normalization of the images was performed with a method proposed by Saxena et al. [1998] which is adopted to the image type. The number of scans and patients is about half the amount of patients and scans in the ADNI database. The statistics of patients and scans are shown in Table 5.3. This image set was used in publications by Illán [2009], Illán et al. [2011], Kodewitz et al. [2010], Ramírez et al. [2009].

	NC	MCI	AD	$\sum$
Patients	52	114	53	219
Scans	176	448	157	781

Table 5.3.: ADNI FDG-PET scans selected and preprocessed at the University of Granada.

The voxel grid of the image set corresponds to the default settings of the SPM spatial normalization but not to the voxel grid of the MNI template which has dimensions $91 \times 109 \times 91$ voxels. The ROI respectively volume of interest (VOI) maps that are created with SPM toolboxes have exactly this size. In consequence it is necessary to do a spatial normalization of those maps to use them with the image set. As we might use many different maps this additional processing step should be avoided as it introduces an additional source of possible errors and would consume also considerable time.

The labels in the images set are attributed per patient. This means, all scans of a patient that was classified by the medical doctors in the initial examination in a certain group remain in this group no matter the results of later examinations. Therefore, this image set contains several patients whose scans should have been attributed different labels according to the ADNI inclusion criteria.

5.4. Creation of a new image set

The problems listed in the end of the proceeding section 5.3 and the lack of an other image set of sufficient size called for the preparation of a new ADNI image set. Its creation is described in the following.

Among the several available stages of pre-processing that are available in the ADNI database, the one that matches most the needs of computer analysis had to be selected. The most pre-processed ones, the "co-registered, averaged, standardized image and voxel size, uniform resolution" pre-processed images provides the most standardization. Especially the manual alignment to the AC-PC is favorable as a visual control of an automatic alignment would require visual control by a clinician. We therefore chose this pre-processing step to be the starting point for our image set.

To create a set of images suitable for the intended methods of feature extraction and classification the following protocol is followed:

- Spatial normalization of all images to MNI space with same voxel grid

- Creation and application of a mask for the background area
- Visual verification of the image preprocessing
- Intensity normalization with a method that assures a constant normalization over all images

These steps are described in detail in the following.

5.4.1. Revision of meta data

The ADNI database provides only labels per patient. This means the class label is attributed to the patient after the screening tests and remains the same during the follow-up time of the study, which is up to 3 years. However, mental state tests are performed during the follow-up temporally close to each brain scan. We therefore prefer to attribute to each *scan* the label corresponding to the class label rules of the study using the mental state test. This makes it possible that the class label changes for one patient between different scans.

The rules ADNI rules for label attribution (see page 74) do not allow unique label attribution in cases where the patient has a CDR of 0.5 and a MMSE score between 24 and 26 (inclusive). To resolve these ambiguities the following additional rules are applied: For the baseline scan the label attributed to the patient is used, as the patients class is chosen by a specialized medical doctor. In the rare cases that the MMSE score is 26 or higher and CDR is 1 the CDR indicated label AD is attributed as the CDR is based on the experience of the medical doctor whereas the MMSE is a simple questionnaire that does not take the education of the patient into account. For patients in AD or MCI group that have ambiguous scans but no scan with differing label the scans are labeled according to the patients group. For the AD patients this rule is based on the fact that the recovery of patients once identified as AD by a medical doctor to normal brain function can be practically excluded. In the MCI case we can, applying this assumption, finally separate a group of patients that clearly degrades to AD. Those patients have a considerable decline in the MMSE and the CDR becomes 1 in the end of the follow-up period. This group of mild cognitive impairement converting to Alzheimer's disease ($\mathrm{MCI_{AD}}$) patients contains, as expected, round 15% of the total amount of MCI patients. These scans are of high value for the testing of early detection methods.

The rules used for the selection or exclusion of patients and scans can be summarized as follows:

- Patients with one scan only are not accepted.
- For NC patients one scan with label MCI is allowed.

- MCI and AD patients with a NC result in the follow-up exams are excluded.
- All scans of MCI and AD patients that are ambiguous are labeled according to the patients group.
- Scans that are not labeled according to the patients group are excluded.
- Patients in MCI group that have multiple exams in the end of the follow-up period that indicate AD are separated and form the MCI_{AD} group.

The application of these label attribution and patient selection rules results in an image set with 335 subjects. This corresponds to an exclusion of 17% of the initial 404 available patients. The numbers each class can be found in Table 5.4 and the demographics of the set in Table 5.5.

	NC	MCI	MCI_{AD}	AD	$\sum$
Patients/baseline	84	124	25	82	335
MMSE	28 ± 2	27 ± 2	26 ± 1	23 ± 2	

Table 5.4.: Scans in new ADNI PET data set.

	NC		MCI		MCI_{AD}		AD		Total	
	mean	sd	mean	sd	mean	sd	mean	sd	mean	sd
Male	75.4	4.8	76.1	7.1	77.0	6.0	75.3	6.9	75.7	6.5
Female	76.9	5.3	74.1	7.4	74.6	6.1	75.2	7.0	75.2	6.8
All	76.0	5.0	75.4	7.2	76.6	6.0	75.3	6.9	75.5	6.6

Table 5.5.: Demographics of the baseline scans forming the new dataset. Mean and standard deviation (sd) of the patients age.

5.4.2. Spatial normalization (co-registration)

After selecting the subjects all scans are spatially normalized to a template. To grant voxel-to-voxel comparability and make the use of several available tools like, e.g. tools for ROI selection as easy as possible, the images are normalized to the MNI space using the SPM toolbox. To do so the origin of the original ADNI images has to be moved from the corner of the image to the center. For this step the reorient function of SPM is used.

For the actual normalization a batch template is generated to automatize the spatial normalization. Using such a template it is possible to co-registrate a whole set of images

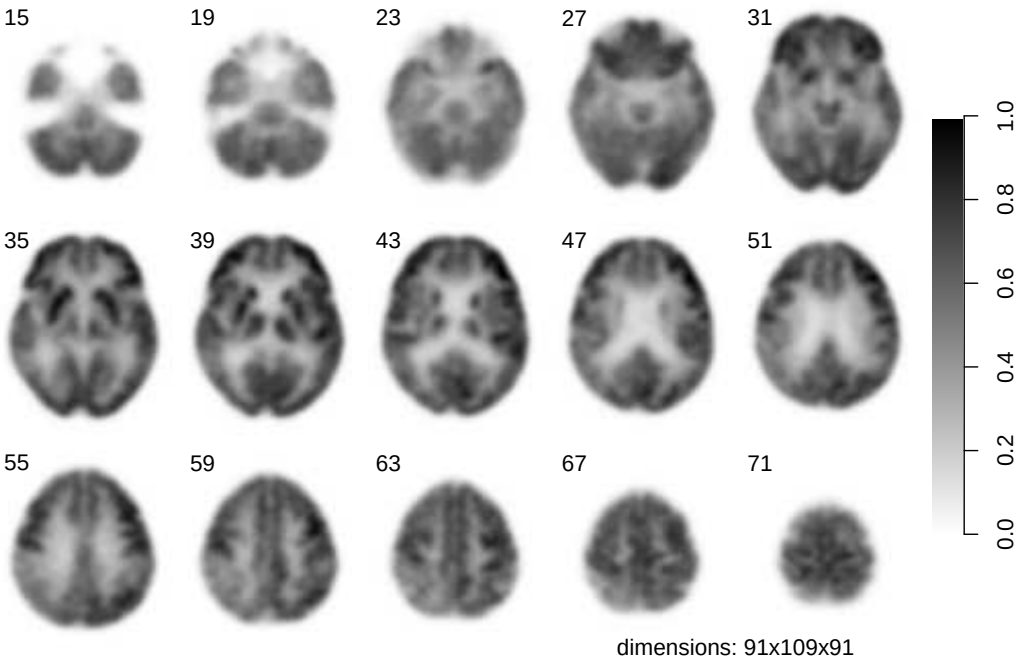


Figure 5.3.: Slices of same scan as in Figure 5.1 but normalized to the MNI standard brain; The scull is stripped by application of the created mask; The normalization has also caused a slight degradation in image sharpness.

at once. The used batch template differs only in two points from the default template. The source image smoothing was disabled as the ADNI images are already smoothed with 8mm FWHM and the bounding box was changed ([-89 -127 -73; 92 90 108]) to obtain in the end an image that has the same size as the MNI template. The voxel size is set to 2mm cubic voxel as this is the minimal spatial resolution of scanners used in the ADNI study.

After spatial normalization all images were visually inspected and eventual normalization failures corrected or if not possible the image excluded from the data set. The spatial normalization has caused a slight degradation of image sharpness that can be seen comparing Figure 5.1d and Figure 5.3.

5.4.3. Mask generation

The mask for background voxels is derived from the SPM PET template by thresholding. The applied threshold was determined deterministically. It was varied until the desired stripping of the scull was achieved. Masking the background prevents all further analysis to respond to the background which can heavily depend on the different types of scanner. We discovered that without masking the background a classifier can actually be misguided to distinguish different sites of acquisition.

5.4.4. Intensity normalization

The preprocessed images of the ADNI database are normalized to a average voxel intensity of one. In the baseline scans however 92 scans with negative voxel values have been found. Negative voxel values can be involuntarily inserted to PET scans by spatial normalization errors and the use of filtered backprojection (FBP) for the reconstruction. The scans containing negative voxels were reconstructed using FBP, while the other scans not containing negative voxels were reconstructed be more modern iterative algorithms, such as ordered subset expectation maximization (OS-EM). FBP creates negative values in areas of low count rate [Razifar, 2005], which means outside of the brain. In our case we found only a very small number (< 100) of negative voxels inside the brain volume. All values were larger than -0.1. Those values are replaced with their absolute value $I(x, y, z) = |I(x, y, z)|$ to obtain purely positive pixel values. Afterwards all voxels that are outside of the brain are set to zero using the generated mask.

5.5. Basic analysis of the new ADNI data set

To get a first impression of the brain scans and the differences between scans of NC and AD patients it is interesting to have a look on simple voxel-wise statistics. The images shown here are obtained by a simple stratification over the classes.

Figure 5.4, for instance, shows the voxel-wise mean intensity of all ADNI baseline scans. This figure shows us where to expect which level of metabolism. It allows to determine hypo- and hyper-metabolism compared to the whole population by simply comparing the intensity.

Figure 5.5 shows the voxel-wise variance between the classes (inter-class variance), so this is a first indicator where to expect differences between NC and AD patients. The graphics show that the ADNI dataset has a large inter-class variance in the back part of the brain. These areas are part of precuneus, posterior cingulate and inferior parietal lobe. Finding with a new method differences in brain regions with low inter-class variance must raise doubt in the new findings.

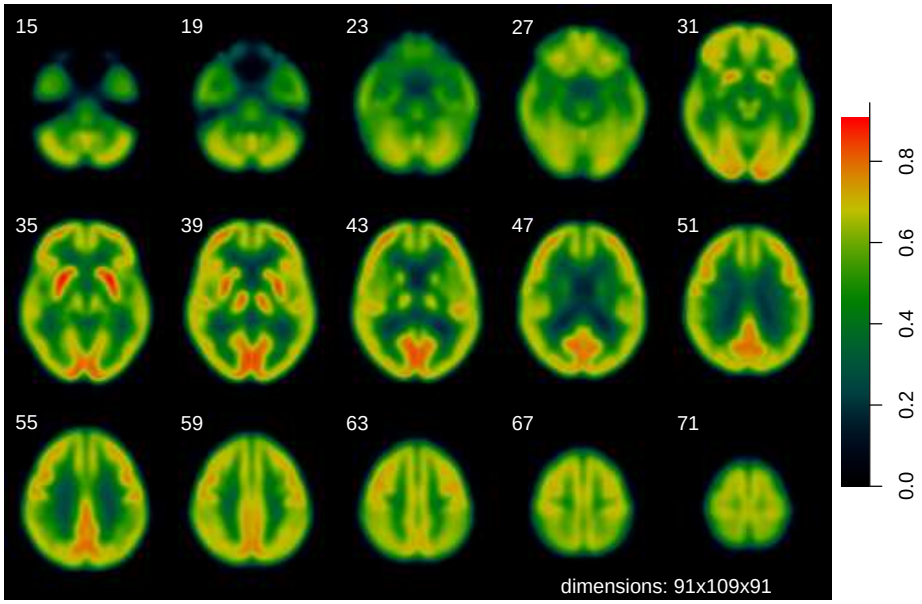


Figure 5.4.: Voxel-wise mean off all baseline scans in the image set; Transversal slices numbered from bottom to top; Color scale provided on the right.

Figure 5.6 shows the voxel-wise variance in the classes (intra-class variance). The intra-class-variance indicates the significance for the differences pointed by the inter-class variance. Class differences found in areas of low intra-class variance are expected to be more robust.

The results of this very basic analysis will be used as a reference to check for consistency. Detailed voxel-wise analysis can be performed by software packages like SPM using ANOVA, analysis of co-variance (ANCOVA) and its multivariate correspondents. These methods were subject of numerous publications with application to CT, MRI and EEG. SPM was used to analyse AD for example by Ishii et al. [2001], Drzezga et al. [2003], Scarmeas et al. [2004] and Yakushev et al. [2009].

As we intend to apply tensor decomposition methods to the prepared images the rank of the images is also of interest. As there is no algorithm to calculate the rank of tensors of order ≥ 3 we can only investigate the rank of slices. The calculation of the tensor rank is in general NP-hard and even the calculation of upper- and lower-bounds is challenging [Alexeev et al., 2011]. In Figure 5.7 the column rank of slices is plotted together with the number of columns that is occupied by the brain in the same slice.

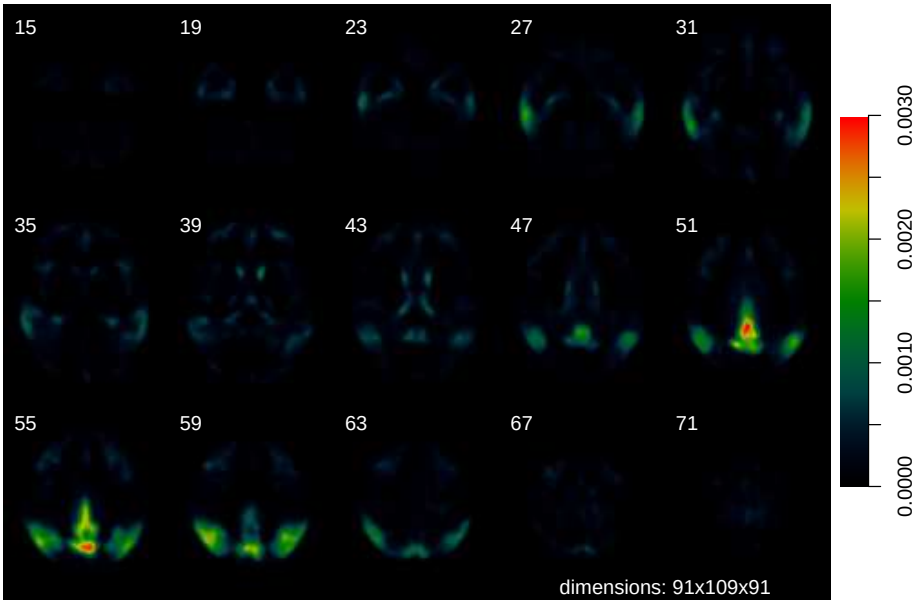


Figure 5.5.: Voxel-wise variance between NC and AD, i.e. inter-class-variance.

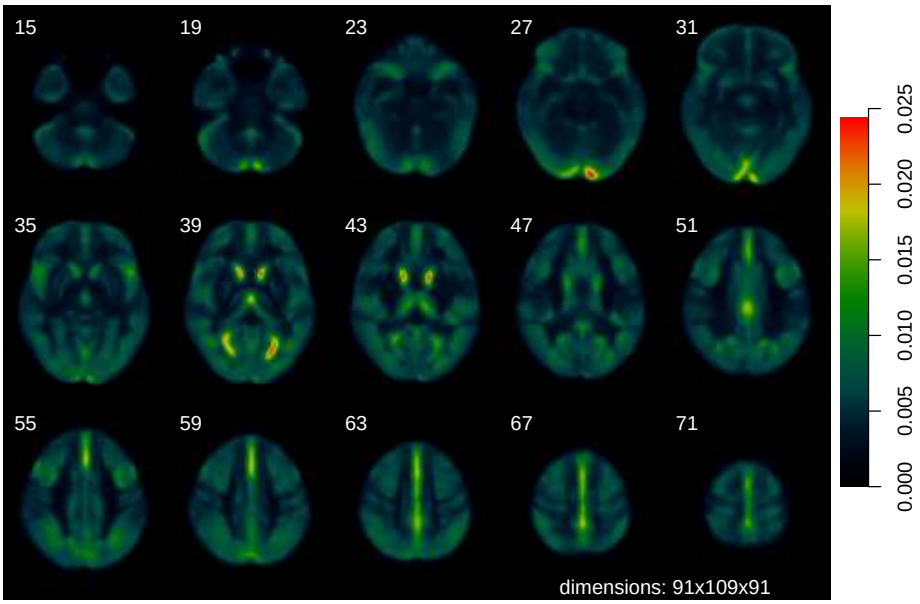


Figure 5.6.: Voxel-wise variance in NC and AD, i.e. intra-class-variance.

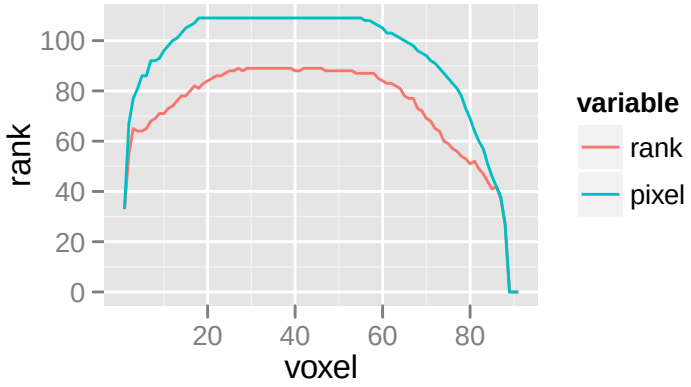
These plots show that the rank of each slice is more or less the same than the number of pixels that is occupied by the brain. That means that the brain slices are all close to full rank.

As a reference we present the classification results obtained with the methods presented in Kodewitz [2009, section 6.2.2] and Kodewitz et al. [2010]. The same single transversal slice intersecting parietal and temporal lobe of the images is selected and vectorized. In the two before mentioned publications the ADNI dataset prepared as the University of Granada (see 5.3), which has different dimensions as the new dataset, was used. The slice in the new data set corresponding to slice 42 in the Granada dataset is slice 56, which we use therefore as best guess in the determination of the slice to be used. The precise index of the slice is determined heuristically. The slices of all images form a matrix with dimensions # images $\times 91 \cdot 109$. This matrix is decomposed using non-negative matrix factorization (NMF)[2] and the obtained weights used as input for the a SVM and a random forest (RF) classifier. For the training we use 60 images per class and the unused images as testing set. The number of basis vectors in the previous publications was $R = 12$. As the image size has changed this parameter might not be optimal. Therefore, we performed a sweep to find the best number of basis vectors.

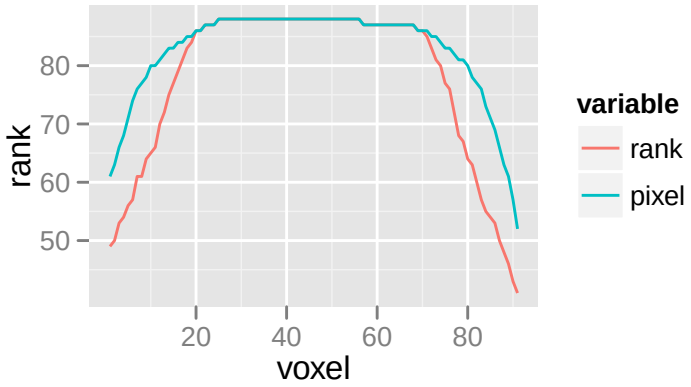
As best slice we found slice number 57 and the use of $R = 16$ basis vectors produced best results. Results tables of classification using RF and SVM are shown in Table 5.6. Classification accuracy of the SVM is superior to the accuracy of the RF. Both classifiers are less performant on the new data set. Classification accuracy for NC vs. AD is at 88.6% for the SVM and at 74.6 % for the RF. Besides good performance in classifying NC vs. AD using the SVM, classification accuracy is also good (85.6 %) for NC vs. $\mathrm{MCI}_{\mathrm{AD}}$. Classification accuracy for MCI vs. $\mathrm{MCI}_{\mathrm{AD}}$, the class combination most interesting for AD early detection, is also at 81.8 %.

We will measure the success of the classification methods presented in the following sections on the here presented results. They were obtained using previously published methods and the data set presented in section 5.4.

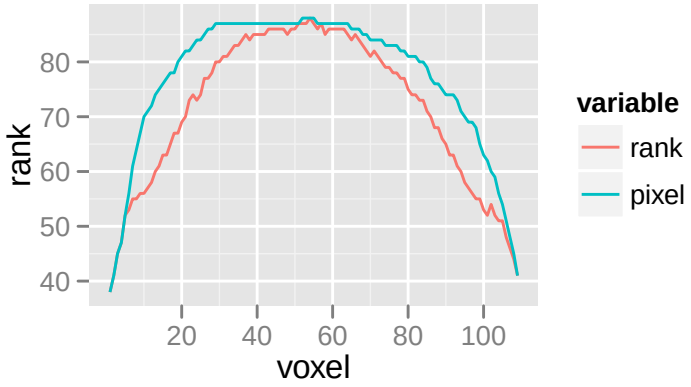
[2] We used the NMF:DTU Toolbox implementation of the Lee and Seung algorithm `http://cogsys.imm.dtu.dk/toolbox/nmf/index.html`



(a) Transversal: slice 1 (bottom most) to 91 (top most)



(b) Sagittal: slice 1 (left most) to 91 (right most)



(c) Lateral: slice 1 (back most) to 109 (front most)

Figure 5.7.: Comparison of the rank and the columns covered by brain for the three possible slice directions.

	oob-err	FPR	FNR
NC / AD	**0.114**	0.097	0.129
NC / MCI	0.225	0.245	0.207
NC / MCI_{AD}	**0.144**	0.129	0.161
MCI / AD	0.171	0.129	0.211
AD / MCI_{AD}	0.222	0.215	0.233
MCI / MCI_{AD}	**0.182**	0.177	0.185

(a) Classification by SVM using NMF decomposition.

	oob-err	FPR	FNR
NC / AD	**0.254**	0.222	0.287
NC / MCI	0.433	0.461	0.406
NC / MCI_{AD}	0.408	0.466	0.350
MCI / AD	0.348	0.361	0.335
AD / MCI_{AD}	0.533	0.470	0.596
MCI / MCI_{AD}	0.449	0.476	0.422

(b) Classification by RF using NMF decomposition.

Table 5.6.: Classification rates with NMF feature extraction method described in master's thesis. 16 basis vectors were extracted from transversal slice 57.

6. Learning the importance of brain areas

To learn about the exact location of AD related changes we will apply the procedure for the extraction of an importance map proposed in chapter 2 to the ADNI data set.

For this approach we have to chose patch size and a feature to be calculated on each patch. Taking into account the dimensions of the images we considered unreasonable to examine patches bigger than $7 \times 7 \times 7$ voxels as this size already corresponds to $\approx 7.5\%$ of the image size and we do not expect good results with a sub-division rougher than that. As voxel-wise analysis, as in SPM, has produced good classification results based on the intensities only, we will use the mean intensity of each patch as feature. This has the advantage that we can compute extensive statistics as the mean is very cheap in calculation.

6.1. Extraction of importance maps

We transformed the images in sets of non-overlapping patches with the patch size in a range from $2 \times 2 \times 2$ to $7 \times 7 \times 7$ voxels and generated an importance map for each of those patch sizes. The mean intensity of all patches was calculated and passed to a RF as features for classification and calculation of feature importance. The RFs used to calculate the feature importance contain 500 trees and at each split $\sqrt{\text{number of features}}$ splits were tried to find the best split. We repeated the training of the RFs 50 times to be able to asses the variance of the classification result. Minimum classification error rates of 17% where achieved using $2 \times 2 \times 2$ patches and augmenting constantly $\sim 0.5\,\%$ for each next bigger patch size. For all patch sizes the variance of the RFs' classification error was at $(1.5 \pm 0.5) \times 10^{-4}$. The classification error of the RFs used to extract the importance map can be found in Table 6.1. Note that even though we use a quite simple method the classification error achieved on this specific classification task is lower than with the reference method (see Table 5.6b).

Figure 6.1 visualizes the mean decrease in accuracy obtained from the RF classification mapped back to the original image size. The map is presented as a colored overlay to a gray scale version of the reference map shown in Figure 6.3 to facilitate interpretation of the extracted importance map. The feature importance had for all patch sizes in a range from $2 \times 2 \times 2$ to $7 \times 7 \times 7$ very similar distribution over the brain.

We then created a fusion of all maps that is shown in Figure 6.2. As classification rates obtained in the measurement of the importance showed significant differences for

the different patch sizes (see Table 6.1) we preferred the weighted mean over the generic mean to combine the single maps. So we combined the single maps, after mapping back to the original image size, by a weighted mean with the corresponding classification rate as weight:

$$\xi(n) = \frac{\sum_{k=1}^{K} w_k \xi_k(n)}{\sum w_k} \tag{6.1}$$

with K the number of maps or images to be combined and w_k the classification rate of the RF that computed the importance ξ_k. n is again the voxel index.

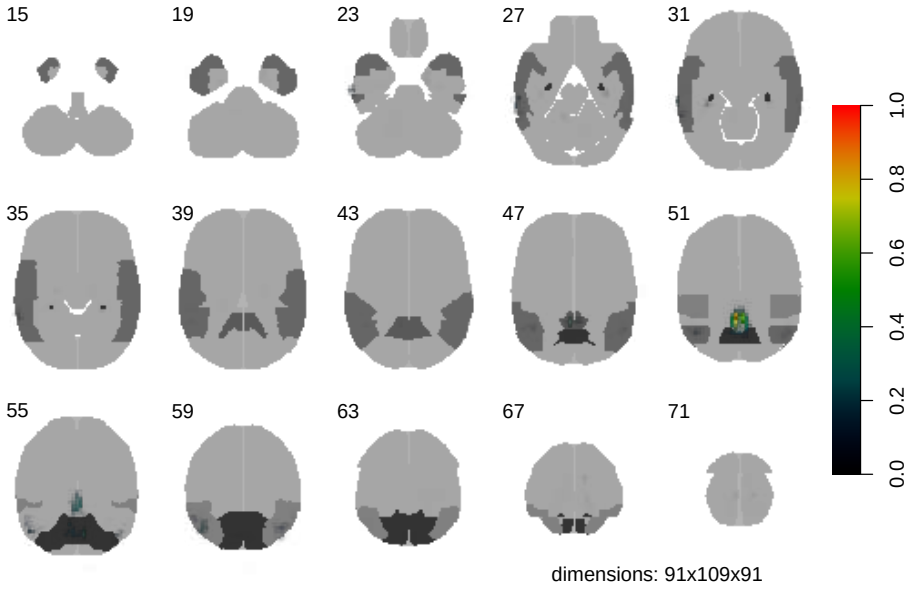
cube size	classification rates: mean ± std-error		
	err. rate	FPR	FNR
$2 \times 2 \times 2$	$\mathbf{0.170 \pm 0.009}$	0.161 ± 0.014	0.179 ± 0.014
$3 \times 3 \times 3$	0.178 ± 0.011	0.169 ± 0.013	0.187 ± 0.018
$4 \times 4 \times 4$	0.185 ± 0.010	0.176 ± 0.012	0.194 ± 0.019
$5 \times 5 \times 5$	0.188 ± 0.010	0.176 ± 0.012	0.201 ± 0.018
$6 \times 6 \times 6$	0.205 ± 0.012	0.194 ± 0.016	0.218 ± 0.018
$7 \times 7 \times 7$	0.210 ± 0.011	0.202 ± 0.014	0.219 ± 0.018

Table 6.1.: Classification results obtained by RFs trained on the whole brain volume. The sole feature calculated for each patch is the mean intensity. These classifiers where used to calculate the feature importance. After transformation of the patch importance to the voxel importance the importance can be visualized like in Figure 6.1.

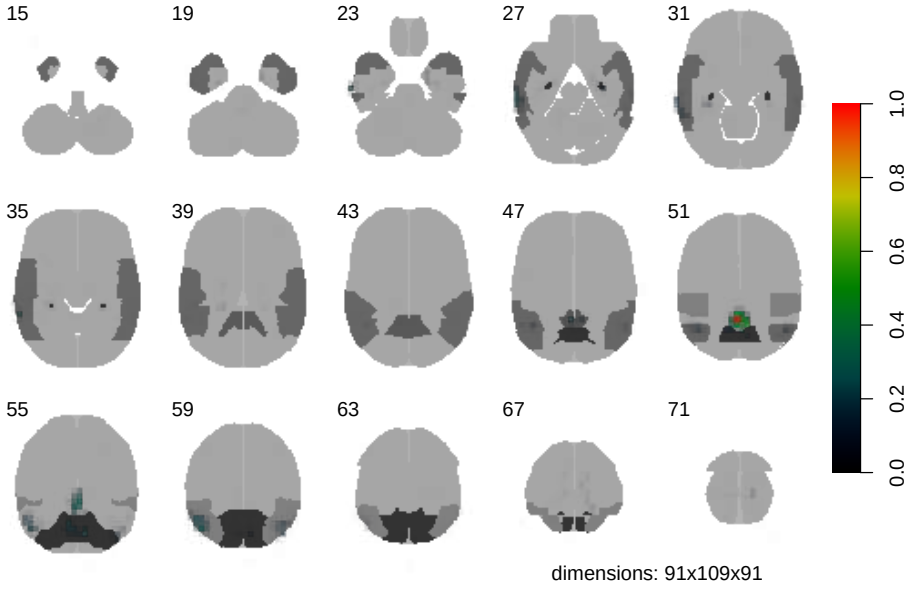
6.2. Evaluation of learned importance map

For a statistical evaluation of the learned importance map a reference map of all brain areas relevant for the diagnosis of AD was created using WFU pickatlas. This map serves as reference in visual examination and is used to calculate statistics describing the distribution of important voxels across the different brain ares. The map created after AD literature [Braak and Braak, 1991, Delacourte et al., 1999, Herholz et al., 2002, Minoshima et al., 1997, Mosconi et al., 2007] consists of 8 ROIs: hippocampus, inferior and superior parietal lobe, inferior, middle and superior temporal lobe, posterior cingulate cortex and precuneus. It is visualized in Figure 6.3.

The high localization of RF importance is clearly visible in the histograms in Figure 6.4. To obtain more detail in the tail of the distribution we have zoomed in on the y-axis and do not show the lowest bin (0 to 0.05). All histograms besides those of inferior parietal lobe, posterior cingulate and precuneus have a very heavy lowest bin.

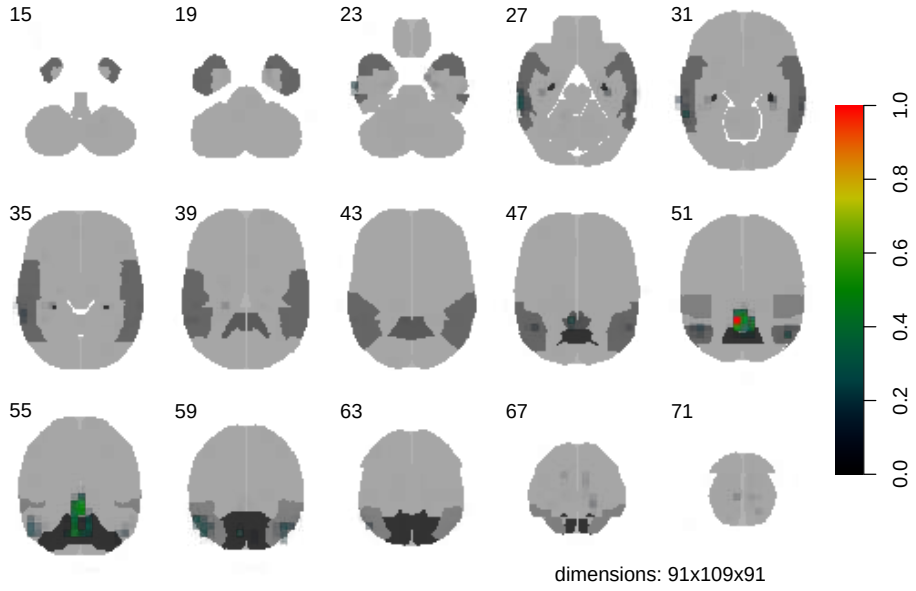


(a) patch size $2 \times 2 \times 2$ voxels

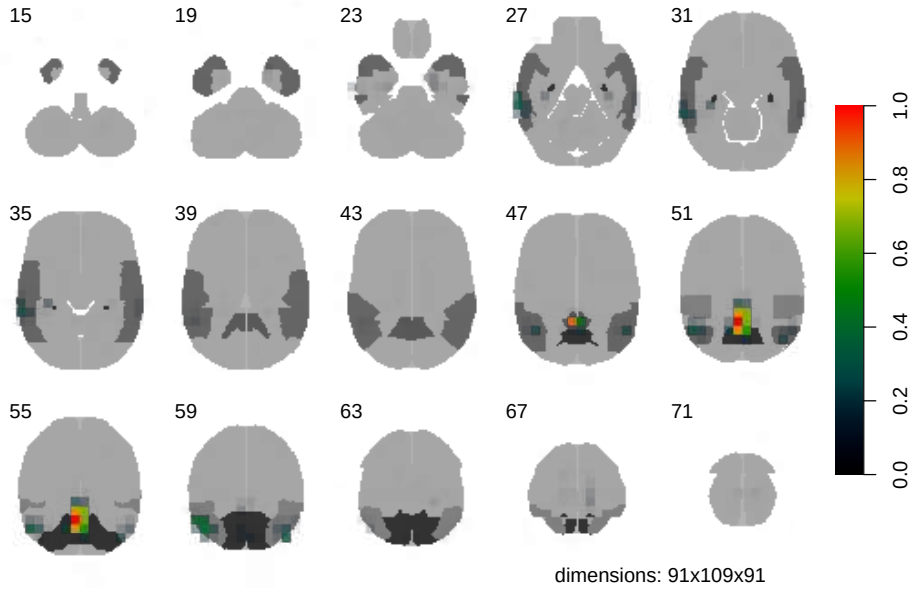


(b) patch size $3 \times 3 \times 3$ voxels

Figure 6.1.: Overlay of feature importance in AD vs. NC classification obtained by RF; A standard color scale is used: red indicates highest importance, blue indicates low importance; importance below 0.1 is transparent.

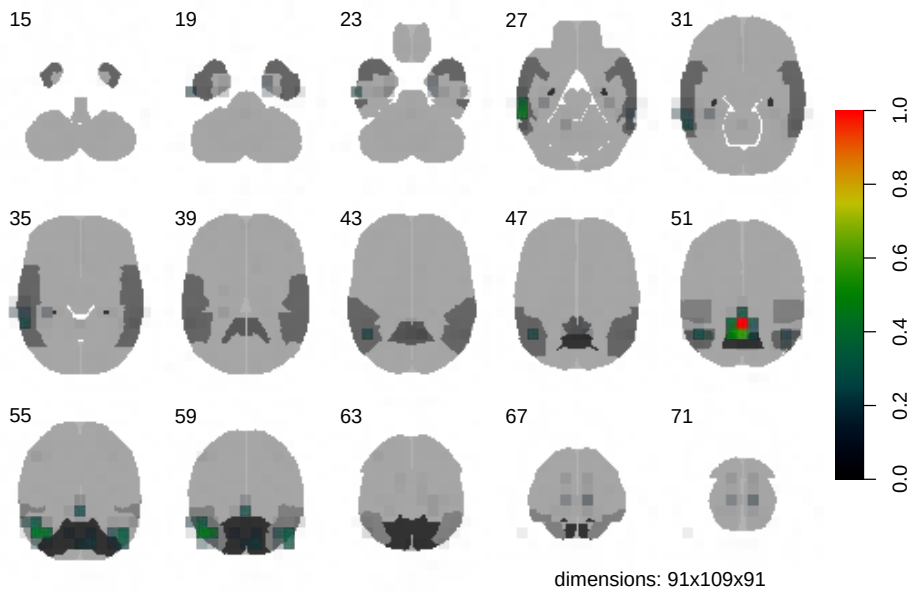


(c) patch size $4 \times 4 \times 4$ voxels

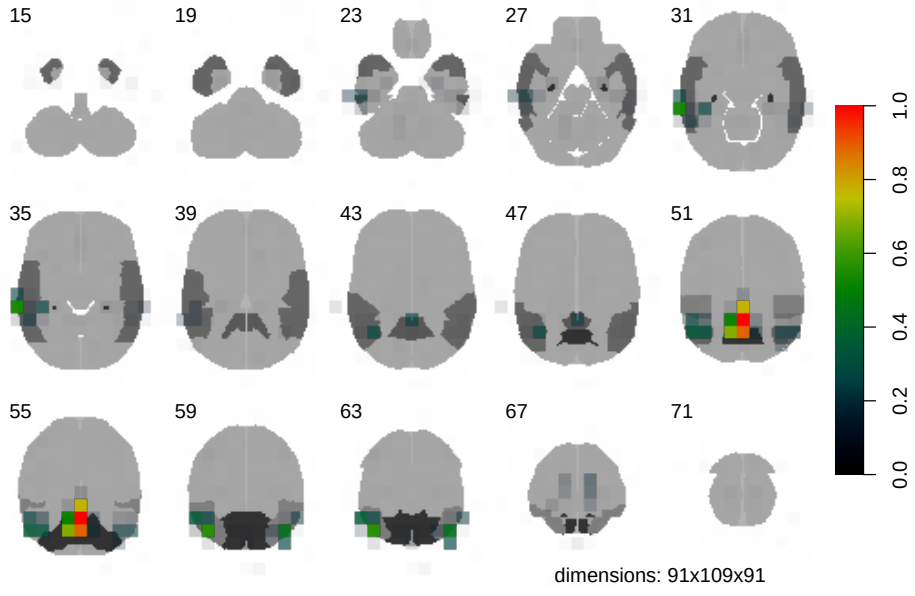


(d) patch size $5 \times 5 \times 5$ voxels

Figure 6.1.: Overlay of feature importance in AD vs. NC classification obtained by RF; red indicates highest importance, importance below 0.1 is transparent.

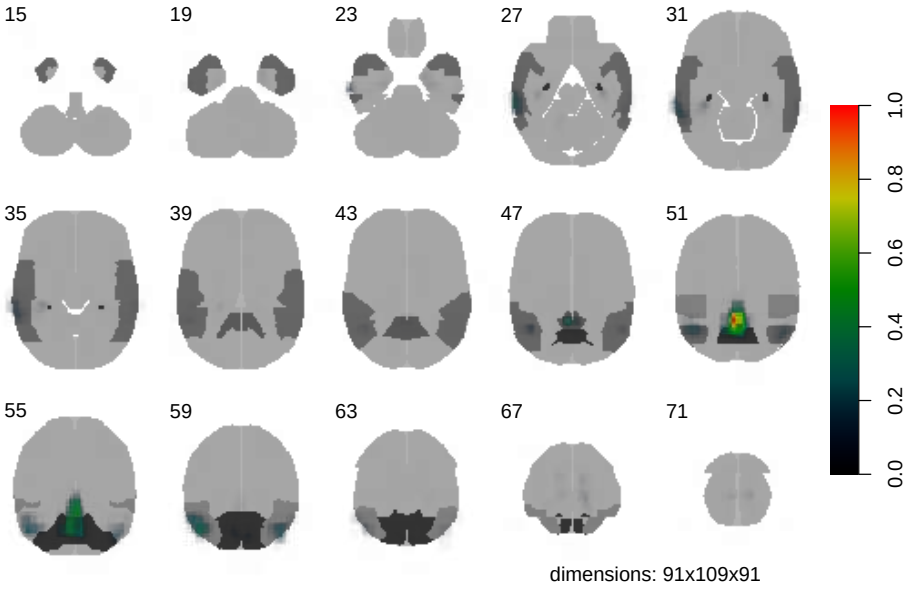


(e) patch size $6 \times 6 \times 6$ voxels

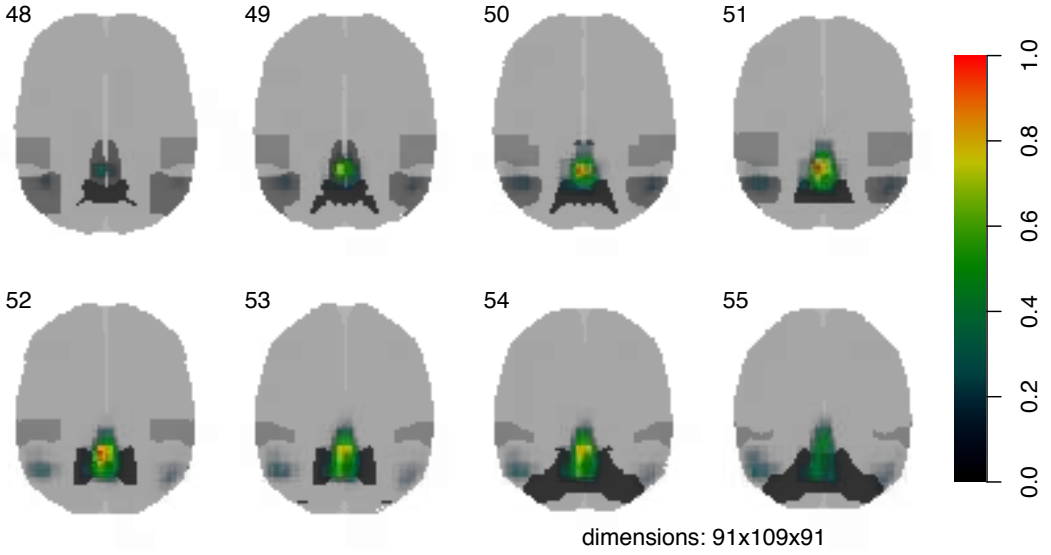


(f) patch size $7 \times 7 \times 7$ voxels

Figure 6.1.: Overlay of feature importance in AD vs. NC classification obtained by RF; red indicates highest importance, importance below 0.1 is transparent.



(a) weighted mean of importance images of patch size $2 \times 2 \times 2$ to $7 \times 7 \times 7$



(b) Detail of most important slices of weighted mean of importance images (Figure 6.2a). Slices 50 and 51 contain the voxels of high importance that are not contained in a ROI. High importance voxels in neighboring slices 49 and 51 are contained in ROIs.

Figure 6.2.: Importance map created by fusion of all maps shown in Figure 6.1 using a weighted mean; the transversal slices are numbered from the bottom to the top of the brain.

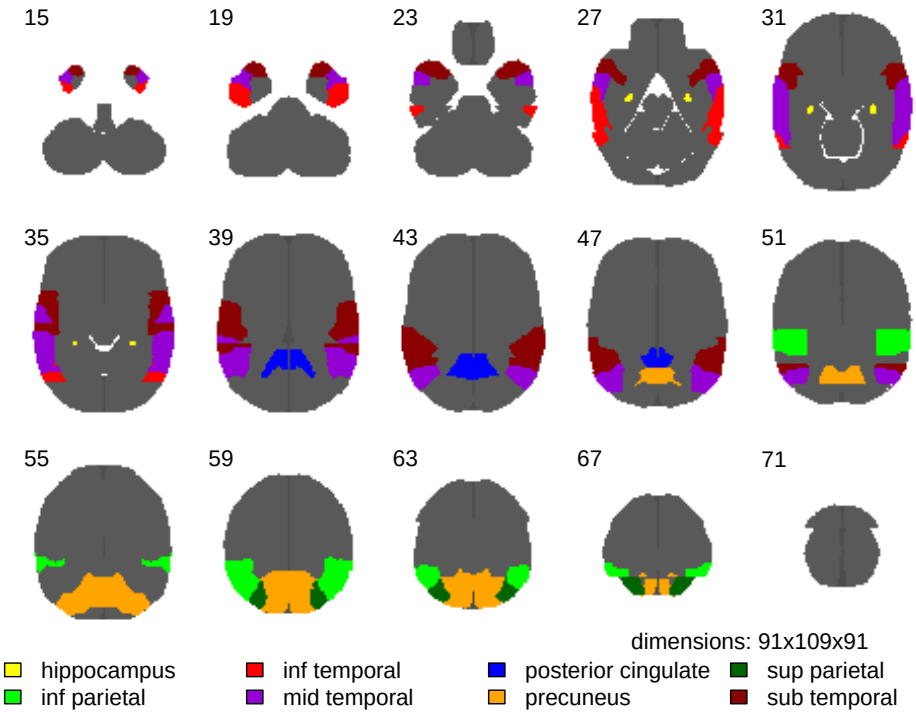


Figure 6.3.: Detailed atlas for Alzheimer's disease brain zone evaluation. 8 zones mentioned in AD related literature.

That means most of the voxels forming the ROI are not important at all. In Table 6.2 it becomes even more visible that the RF map points posterior cingulate and precuneus as ROIs of high interest in the classification of AD vs. NC. As visible in Table 6.2 all voxels exceeding 75% of maximum importance are situated in posterior cingulate, precuneus or outside the defined ROIs. The amount of voxels of high importance not contained in a ROI is at least a magnitude lower than the amount of voxels in posterior cingulate or precuneus. A specific lookup of those voxels not contained in a ROI revealed that those pixels are located in close vicinity to posterior cingulate and/or precuneus. These pixels are also visible in the detail view of the map in Figure 6.2b. Superior temporal lobe, superior parietal lobe and, surprisingly, hippocampus do not contain any voxels exceeding a RF importance of 0.25.

A comparison with the inter-class variance (see Figure 5.5) shows that the areas considered important by our algorithm are areas of high inter-class variance, but of smaller extent than the areas of high inter-class variance. In the numeric evaluation of the brain areas (see Table 6.3) it becomes evident that the inter-class variance

is more distributed over the brain areas. Especially the precuneus contains a large number of voxels with high inter-class variance, but is in our importance map of similar importance as the posterior cingulate, which contains far less voxels of high inter-class variance.

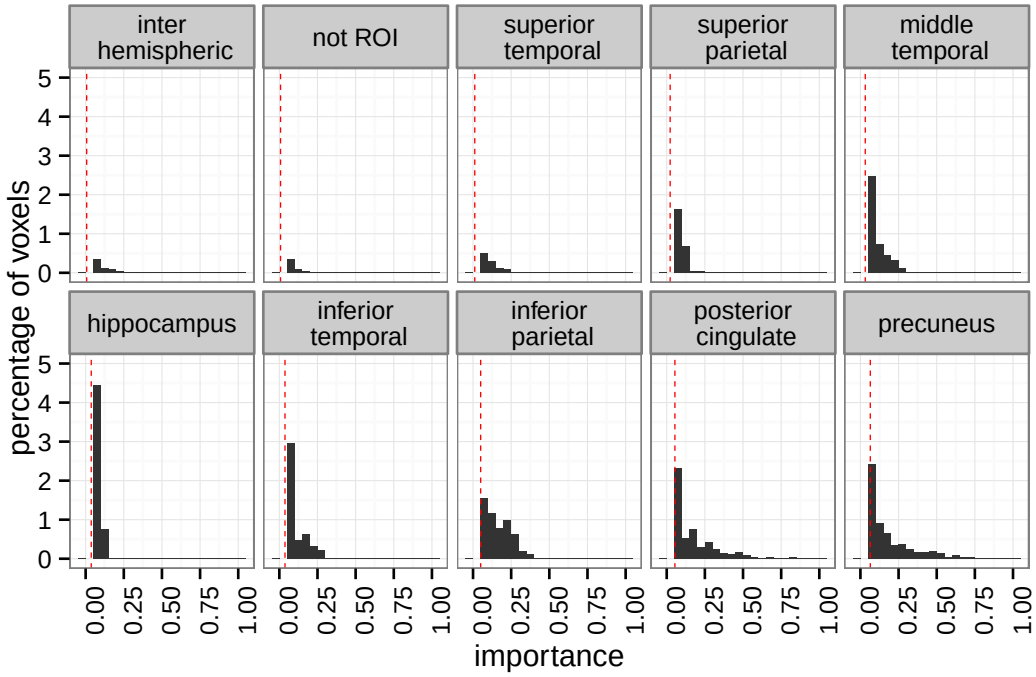


Figure 6.4.: Distributions of voxel importance in the ROIs defined in Figure 6.3. The red dashed lines indicate the average importance of the region. The numeric values of the average importance, along with other numeric information about the distribution of the importance, can be found in Table 6.2.

6.3. Discussion

The maps for the importance of different brain areas that were obtained by our machine learning approach using the RF classifier indicated very similar areas of interest over all patch sizes. Those areas are much smaller than expected after literature study, but, in accordance with literature, localized mainly in parietal and temporal lobe. The areas of highest importance have very similar location as the areas of highest inter-class variance (see Figure 5.5).

As regions of highest interest for the distinction of NC and AD using FDG PET

	importance		% voxels in bin		
	mean	max	.25 to .5	.5 to .75	>.75
background voxels	1.13e-03	0.31	0.0025	0.0	0.0
not ROI	7.96e-03	1.00	0.24	0.078	0.032
inter hemispheric	6.87e-03	0.50	0.16	0.0	0.0
hippocampus	3.66e-02	0.13	0.0	0.0	0.0
inferior parietal	5.00e-02	0.40	4.6	0.0	0.0
inferior temporal	3.71e-02	0.28	1.0	0.0	0.0
middle temporal	3.18e-02	0.31	0.6	0.0	0.0
posterior cingulate	5.35e-02	**0.83**	**5.4**	**0.8**	**0.3**
precuneus	6.49e-02	**0.92**	**5.8**	**1.6**	**0.14**
superior parietal	2.24e-02	0.21	0.0	0.0	0.0
superior temporal	1.14e-02	0.24	0.0	0.0	0.0

Table 6.2.: Comparison of the importance map obtained by RF (see Figure 6.2a) and the map of ROIs (see Figure 6.3). Given are the mean importance, the maximum importance as well as the percentage of voxels in each region with an importance in the bins $.25 < i < .5$, $.5 < i < .75$ and $.75 < i$. For more detail see also the histograms in Figure 6.4.

	inter-class variance		% voxels in bin		
	mean	max	.25 to .5	.5 to .75	> .75
background voxels	5.71e-06	1.71e-03	0.063	5.9e-4	0
not ROI	1.34e-04	3.07e-03	1.2	0.37	0.092
inter hemispheric	2.05e-04	2.81e-03	3.2	0.98	0.24
hippocampus	3.81e-04	8.80e-04	2.7	0	0
inferior parietal	5.66e-04	2.22e-03	19	11	0
inferior temporal	4.40e-04	1.76e-03	14	0.99	0
middle temporal	3.91e-04	1.82e-03	12	0.68	0
posterior cingulate	4.82e-04	2.37e-03	18	4.3	0.15
precuneus	6.28e-04	**2.91e-03**	**20**	**8.7**	**2.2**
superior parietal	4.12e-04	1.78e-03	18	0.42	0
superior temporal	2.09e-04	1.80e-03	4.6	0.24	0

Table 6.3.: Comparison of the inter-class variance map and the map of ROIs. Given are mean and maximum class-separation distance as well as the percentage of voxels in each region with an class-separation distance in the bins $.25 < i < .5$, $.5 < i < .75$ and $.75 < i$ of the maximum class-separation distance.

scans we found posterior cingulate and precuneus. A certain amount of voxels not included in any of the defined ROIs was found to be also of high interest. In fact, the voxels of maximum RF importance were not contained in any defined ROI but in close vicinity to posterior cingulate and/or precuneus. This finding might be due to the imperfectness of registration to the MNI standard brain. We suggest therefore to consider slightly enlarging ROIs when using a model similar to ours.

The presented approach to extract importance maps directly from image data is complementary to voxel-wise analysis, either by stratified statistics (see section 5.5) or SPM, insofar as it does not perform a univariate analysis on each voxel but a multivariate analysis. The RF classifier does not treat the features, and due to the design of our approach equivalently the patches in the brain, independently. Interactions between them can contribute therefore to the resulting map, which is not possible with the before mentioned other methods. Furthermore, we obtain by the possibility to vary the patch size the opportunity to calculate the features at different scales.

7. Data reduction by importance threshold

A straight forward way to use the importance map generated in the preceding chapter 6 for data reduction is to divide the images in a regular grid of non-overlapping patches and exclude the patches whose mean importance is not exceeding a certain threshold. We have implemented this approach and will use the obtained results to compare classification error and the possible data reduction with a more sophisticated way to reduce the data in chapter 8.

7.1. Experiment protocol

The images and the importance map are divided in a regular grid of non-overlapping patches as we have done in section 2.3. First we calculate the mean of all patches in the importance map, which results in an importance map of the same size as the grid of patches, i.e. one importance value per patch. The patches with an importance below threshold are excluded from the further calculations. We then calculate a feature for each remaining patch. This feature is the mean intensity of the patch, as this feature showed high predictive power in the preceding section 8.3 and using the same feature allows comparison with those results. The feature vectors and labels are passed to a SVM for classification. Obtaining only poor results with linear and polynomial kernels, we used a radial basis function (RBF) in the displayed results. The kernel parameters used are: $\gamma = \frac{1}{\#\text{of features}}$ and $C = 1$, the default parameters of the used libsvm implementation. We performed a 100 fold bootstrap cross-validation in order to obtain a prediction of the error rates for unknown data. As there is no analytic method to determine the level of the threshold we performed a sweep for this parameter.

In this chapter we also want to compare the performance of the importance map with the performance of another map representing prior information. The inter-class variance map, shown in Figure 5.5, is such a map. It is obtained by voxel-wise statistics and does therefore not take any patterns into account. We will proceed for the inter-class variance map in the exact same way as for the importance map. Evidently, the threshold values have to be different in this case as the inter-class variance values of our dataset are in a range of $[0; 0.003]$, whereas the importance values are normalized to the interval $[0; 1]$. Therefore, we have to compare the results originating from the two maps by the number of patches exceeding the threshold level.

It is to be expected that the map which captures better the distribution of the importance for the classification provides lower classification error at lower numbers of

patches.

7.2. Results

We present plots of the error rates depending on the number of patches exceeding threshold in Figure 7.1. The corresponding numerical results are shown in Table 7.1 and Table 7.2. Three different maps are compared: The map created by fusion of the RF importance maps with patch size $2\times2\times2$ to $7\times7\times7$ (see Figure 6.2), the inter-class variance map (see Figure 5.5) and the RF importance map obtained at voxel size $2\times2\times2$ alone (see Figure 6.1a). The false positive rate (FPR) corresponds to classification error for the NC cases and the false negative rate (FNR) to the classification error for the AD cases.

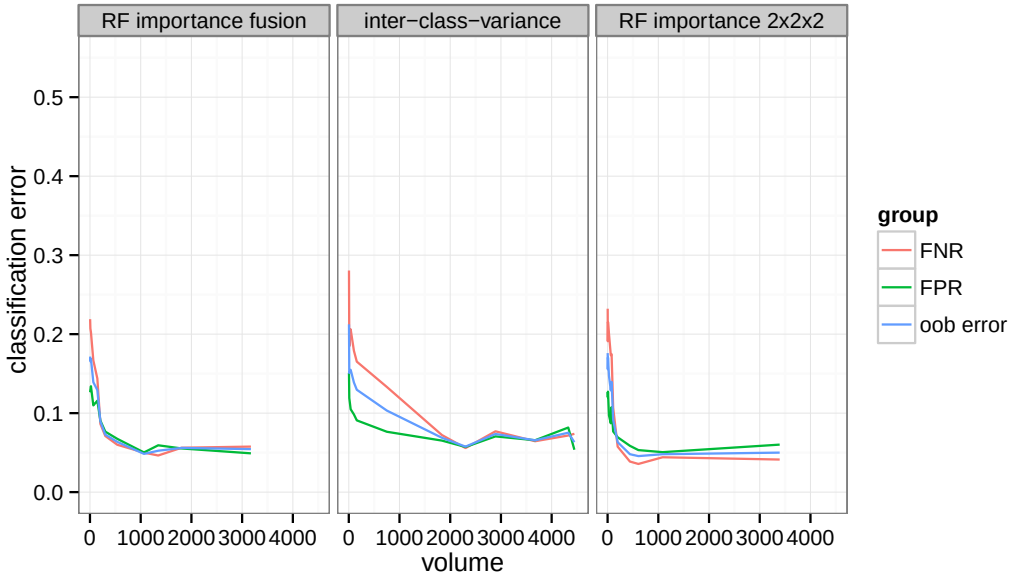
In Figure 7.1 we can see that using either of the importance maps minimal classification error is achieved using approximately 1000 patches and remaining stable for higher numbers of patches. The 1000 patches used here correspond to approximately 7.5 % of the patches used in the extraction of the importance map with same patch size in chapter 6. The minimal classification error achieved with the inter-class variance map is, not only by 2-3% higher, but also achieved with a larger number of patches. At very low numbers of patches the error rates raise regardless the map we used to select the patches. This raise is caused by the fact that we start to exclude information that would be necessary for the classification. This raise is more abrupt for the RF importance maps than for the inter-class variance, which leads to the conclusion that in the RF importance map the patches are better ranked according to their actual importance in the classification than in the inter-class variance map.

When using both variants of importance map the FNR was lower than the FPR when using more than approximately 500 patches. Using the inter-class variance map we observed the inverse, FNR was higher than FPR.

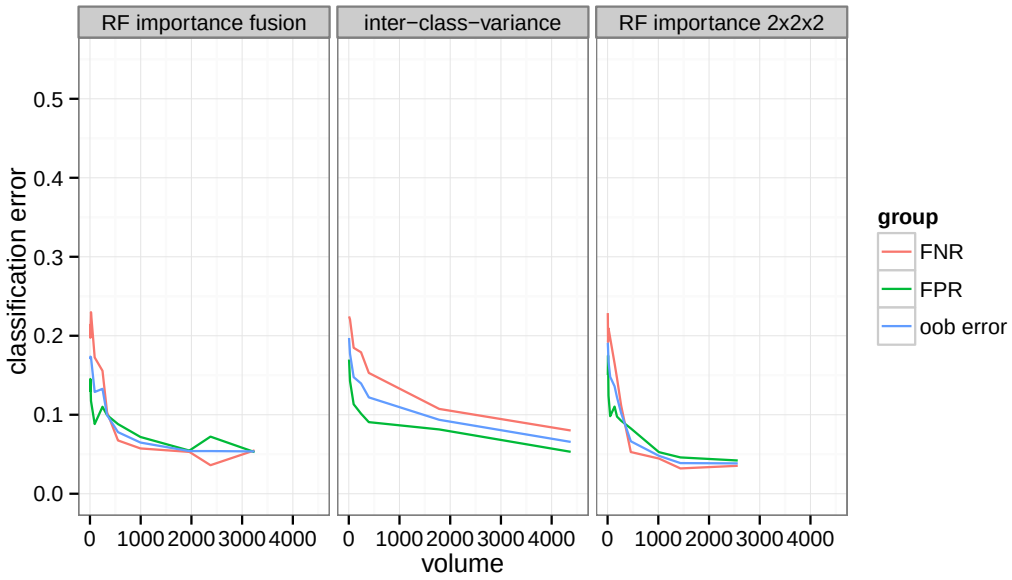
7.3. Discussion

With this experiment we were capable to confirm that the importance maps created with our proposed method capture better the location of class distinguishing features in the PET scans than the inter-class variance. Not only resulted the use of the importance maps in lower classification error, but also were best classification performances achieved with far less patches included in the classification. This means that thresholding with importance maps provides better data reduction than using the inter-class variance map. At the same time we achieved state of the art classification rates of up to 95.7 %.

The raise in classification error when reducing the amount of patches below 500 shows, that important information is excluded by our feature selection via the thresh-



(a) Classification with patch size $4 \times 4 \times 4$



(b) Classification with patch size $3 \times 3 \times 3$

Figure 7.1.: Classification error versus volume of selected patches; Classification error versus volume of selected patches; Feature, or equivalently ROI selection, by applying thresholds to fusioned RF importance map (left), inter-class variance map (middle) and $2 \times 2 \times 2$ RF importance map (right). Classification error rates are calculated by a 100 fold .632 bootstrap cross-validation.

thresh.	vol.	err. rate	FPR	FNR
0.0010	6321	0.080	0.093	0.064
0.0025	3173	0.054	0.049	0.058
0.0050	1807	0.056	0.055	0.056
0.0075	1345	0.052	0.059	0.046
0.0100	1065	**0.048**	0.050	0.050
0.0250	538	0.064	0.068	0.060
0.0500	302	0.073	0.077	0.071
0.0750	205	0.090	0.091	0.086
0.1000	145	0.129	0.116	0.144
0.1500	95	0.142	0.102	0.181
0.2000	66	0.139	0.110	0.167
0.2500	49	0.133	0.084	0.185
0.3000	38	0.146	0.115	0.177
0.4000	19	0.154	0.129	0.180
0.5000	13	0.164	0.113	0.222
0.7000	4	0.158	0.124	0.196

(a) Fusioned RF importance map

thresh.	vol.	err. rate	FPR	FNR
0.0010	8406	0.082	0.087	0.078
0.0025	3393	0.057	0.057	0.056
0.0050	1084	**0.043**	0.053	0.031
0.0075	607	0.051	0.053	0.047
0.0100	440	0.052	0.056	0.049
0.0250	199	0.066	0.072	0.059
0.0500	110	0.102	0.092	0.113
0.0750	79	0.147	0.102	0.197
0.1000	63	0.147	0.112	0.185
0.1500	38	0.142	0.099	0.189
0.2000	27	0.155	0.106	0.209
0.2500	20	0.157	0.112	0.207
0.3000	12	0.177	0.140	0.223
0.4000	8	0.170	0.132	0.203
0.5000	4	0.173	0.126	0.225
0.7000	2	0.167	0.123	0.213

(b) $2 \times 2 \times 2$ RF importance map

thresh.	vol.	err. rate	FPR	FNR
1.0×10^{-5}	4327	0.074	0.053	0.095
2.5×10^{-5}	3674	0.070	0.075	0.065
5.0×10^{-5}	2892	**0.062**	0.055	0.069
1.0×10^{-4}	1836	0.074	0.085	0.062
2.0×10^{-4}	959	0.092	0.081	0.105
4.0×10^{-4}	395	0.116	0.082	0.151
6.0×10^{-4}	219	0.126	0.089	0.164
8.0×10^{-4}	147	0.150	0.114	0.191
1.0×10^{-3}	101	0.142	0.099	0.183
1.2×10^{-3}	71	0.138	0.097	0.181
1.4×10^{-3}	49	0.149	0.115	0.186
1.6×10^{-3}	27	0.156	0.110	0.201
1.8×10^{-3}	14	0.165	0.124	0.208
2.0×10^{-3}	10	0.164	0.116	0.216
2.2×10^{-3}	5	0.204	0.181	0.231
2.4×10^{-3}	2	0.242	0.157	0.325

(c) Inter-class variance map

Table 7.1.: Classification using $4\times4\times4$ voxel patches. Results obtained using a threshold to select most discriminative features. Volume is given in units of $4 \times 4 \times 4$ voxel patches.

thresh.	vol.	err. rate	FPR	FNR
0.0025	5678	0.280	0.358	0.219
0.0050	3245	0.056	0.060	0.050
0.0075	2376	**0.052**	0.055	0.049
0.0100	1960	0.055	0.065	0.048
0.0250	995	0.057	0.068	0.045
0.0500	552	0.080	0.088	0.071
0.0750	337	0.097	0.079	0.115
0.1000	247	0.136	0.100	0.175
0.1500	157	0.141	0.111	0.173
0.2000	93	0.150	0.118	0.186
0.3000	37	0.168	0.122	0.211
0.4000	20	0.171	0.117	0.225
0.5000	14	0.176	0.131	0.222
0.6000	8	0.171	0.145	0.199
0.7000	5	0.172	0.166	0.182
0.8000	3	0.174	0.139	0.212

(a) Fusioned RF importance map

thresh.	vol.	err. rate	FPR	FNR
0.0025	7761	0.050	0.038	0.060
0.0050	2566	**0.038**	0.042	0.035
0.0075	1436	0.039	0.046	0.032
0.0100	1012	0.046	0.053	0.039
0.0250	458	0.066	0.069	0.064
0.0500	266	0.092	0.090	0.093
0.0750	186	0.122	0.094	0.154
0.1000	138	0.142	0.112	0.176
0.1500	90	0.145	0.108	0.182
0.2000	51	0.141	0.101	0.179
0.3000	28	0.160	0.123	0.196
0.4000	18	0.174	0.136	0.215
0.5000	12	0.178	0.154	0.200
0.6000	9	0.189	0.175	0.204
0.7000	5	0.167	0.136	0.195
0.8000	4	0.172	0.152	0.197

(b) $2 \times 2 \times 2$ RF importance map

thresh.	vol.	err. rate	FPR	FNR
7.5×10^{-6}	10182	0.068	0.071	0.063
1.0×10^{-5}	9843	0.064	0.063	0.067
2.5×10^{-5}	8309	0.068	0.067	0.068
5.0×10^{-5}	6562	0.066	0.056	0.077
7.5×10^{-5}	5291	0.062	0.066	0.056
1.0×10^{-4}	4375	**0.054**	0.043	0.068
2.5×10^{-4}	1783	0.104	0.080	0.127
5.0×10^{-4}	714	0.126	0.089	0.163
7.5×10^{-4}	395	0.116	0.095	0.136
1.0×10^{-3}	242	0.135	0.098	0.175
1.3×10^{-3}	152	0.138	0.109	0.169
1.5×10^{-3}	96	0.152	0.112	0.192
1.8×10^{-3}	47	0.168	0.137	0.203
2.0×10^{-3}	24	0.166	0.149	0.187
2.3×10^{-3}	13	0.189	0.148	0.225
2.5×10^{-3}	5	0.200	0.189	0.211

(c) Inter-class variance map

Table 7.2.: Classification using $3\times3\times3$ voxel patches. Results obtained using a threshold to select most discriminative features. Volume is given in units of $3 \times 3 \times 3$ voxel patches.

old. To achieve better classification results we have to either enhance the map or chose a more discriminative feature. It remains an important task to choose the right threshold in order to retain as much information as necessary to achieve best classification rates. This task can not be solved analytically and the optimal threshold can only be approximated by trial end error.

8. Sampling patches from Alzheimer's disease affected brain areas

In the previous chapter we have used the importance map extracted in chapter 6 and inter-class variance in conjunction with a threshold to reduce the amount of data that has to be analyzed. This approach comes with the drawback that we have to determine the threshold heuristically, which can be very time consuming. Furthermore, we do exclude in an absolutely strict manner the areas with an importance below the threshold. This however can be a disadvantage if the available model for the distribution of important information is not very precise. This is the case in many medical applications, especially in the early detection of AD .

In this chapter we will explore the possibilities of sampling patches from the images. That means we will draw a number of locations in the image and base the analysis on the voxels situated around those locations. In section 8.1 we will present the algorithm used for sampling according to a model of the distribution of information important for the classification which is stored in a map. We will use a map with predefined ROIs (see section 8.2) and the importance map extracted directly from our image set in chapter 6 using the importance map extraction technique presented in chapter 2 (see section 8.3).

8.1. Spatial importance sampling

The sampling method we will present can be used in conjunction with patch wise image analysis and dictionary learning. A common problem in patch wise analysis of images as well as learning of patch based dictionaries, like in Mairal et al. [2009] and Hyvärinen [2010], is the large amount of patches to be treated. Assuming we have an importance map $\xi(j, k, \ell) \in [0; 1]$ showing the location of important information in the image, with 0 the least and 1 the most important. We could then restrict our analysis to areas with an importance exceeding a certain threshold γ. But by analyzing only patches in areas $\xi(j, k, \ell) > \gamma$ we are, of course, at risk to miss information that appears in areas below the threshold. Think of the famous "search the difference" cartoons in magazines. Imagine the cartoonist hides the differences almost always in the left half of the image. Consequently we would tend to exclude the right half of the image by a thresholding approach and miss the few, but perhaps very obvious, differences in the right half.

An other possibility to reduce the amount of patches is to draw a representative random sample of patches for the analysis. This approach avoids to exclude whole areas of the image, but makes no use of the available prior information. Furthermore the size for a random drawn sample to be representative can still be very large in the case of high resolution images.

Therefore, we propose to draw patch locations according to $\xi(j, k, , \ell)$ which is a model for the distribution of the important information. To sample the patch locations according to the map ξ we use an algorithm very similar to the Metropolis-Hastings algorithm originally published by Metropolis et al. [1953]. This algorithm was developed to sample random numbers from complex distributions with heavy tails. This method is often referred to as importance sampling. Because of the analogy with the Metropolis-Hastings algorithm we call our method *spatial* importance sampling.

8.1.1. Sampling algorithm

As prerequisites for our sampling algorithm we require the image I we intend to draw the patches from to be of dimensions $J \times K \times L$ and an importance map $\xi(j, k, \ell)$ of the same dimensions.

In our sampling algorithm we define the drawn patches by their center coordinates $c_0 = (j, k, \ell)$. The patch size is $Q \times R \times S$. The neighborhood of the center pixel constitutes the patch. The corner coordinates are called $c_1, \ldots, c_8$ with the convention that $\|c_1\| < \|c_i\|, \forall i \neq 1$ and c_1 is obtained by

$$c_1 = c_0 - (Q \div 2, R \div 2, S \div 2) \tag{8.1}$$

where $\div$ is the Euclidean division and the other corner coordinates are obtained from c_1 by

$$c_i = c_1 + (Q, R, S) \bullet b, \quad i > 1 \tag{8.2}$$

where $b \in \{(0, 0, 1), (0, 1, 0), (1, 0, 0), (1, 1, 0), (0, 1, 1), (1, 0, 1), (1, 1, 1)\}$.

Example in 2 dimensions: Be 7×6 pixel the patch size and the center coordinate $(10, 17)$. Then the coordinates of the corners are $(7, 14)$, $(7, 20)$, $(14, 14)$ and $(14, 20)$. This example is visualized in figure 8.1.

The spatial importance sampling algorithm proceeds as follows: First a (j, k, ℓ)-coordinate is drawn at random in the image. Then this coordinate is accepted as the center coordinate c_0 of a patch with the probability $\xi(j, k, \ell)$ that is assigned to the same coordinate in the importance map. This method for selecting the patch is the reason why we prefer the unconventional use of the patch center as reference point over

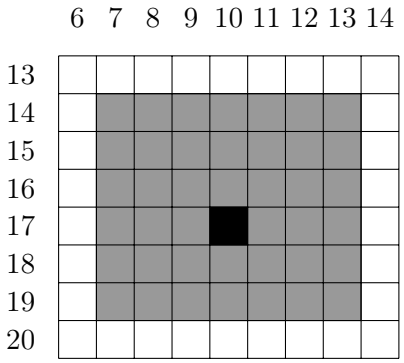


Figure 8.1.: Patch on the pixel grid. Black pixel is the patch center c_0 and the gray pixels around form the patch. Visualization of the two dimensional example.

the usual way to use the upper left or lower right corner as reference. Practically the acceptance of a patch center coordinate is realized by drawing a random number in an interval $[0, 1]$ and comparing with $\xi(j, k, \ell)$. In the case the random number is smaller the coordinate is accepted and rejected in the contrary case.

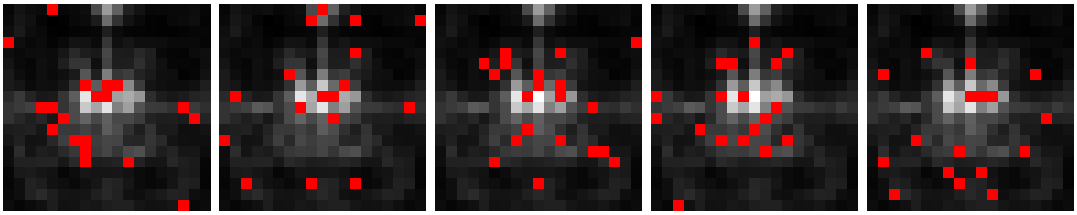
This two steps are repeated until the desired number of patches is reached. Afterwards 2D- or 3D-patches for each of the drawn center points can be analyzed by any desired method. This procedure is also summarized in algorithm 4.

Algorithm 4 Map based patch sampling.

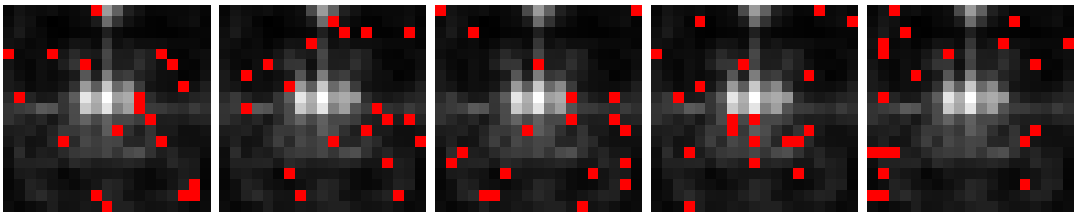
Require: $\xi(j, k, \ell)$ the importance map
1: **repeat**
2: Draw a coordinate (j, k, ℓ) at random
3: **if** $\xi(j, k, \ell) > \alpha \in U[0, 1]$ **then**
4: accept the coordinate as center c_0 of a patch that is to be analyzed
5: **else**
6: reject this coordinates
7: **end if**
8: **until** desired number of patches is reached
9: Gather the voxels around the coordinates that have been selected for analysis
Return: Selected patches

To get a visual impression of the difference between spatial importance sampling and uniform random sampling we have drawn samples of 20 points according to the example importance map shown in Figure 2.5 which we have extracted from center for biological and computational learning (CBCL) face images in section 2.4. Figure 8.2 shows the

distribution of the resulting sample of both methods over the importance map. Using importance sampling the points are drawn concentrated in important areas.



(a) 5 examples for importance sampling.



(b) 5 examples for uniform sampling.

Figure 8.2.: Uniform sampling and importance sampling. 20 points are sampled and displayed as red dots on the face importance image used for the importance sampling.

8.1.2. Prior probability calculus to fix selection ratios between ROIs

In the case we want to use spatial importance sampling to draw patches form two ROIs in a image according to a fixed ratio we have to consider the relative sizes of the ROIs in the patch drawing process. We have to apply a prior to the attributed probabilities to achieve a patch drawing according to the desired ratio.

We consider the situation shown in figure 8.3: Two ROIs (A, B) in an image. We want to draw patches in the two ROIs only and obtain intend to fix the ratio between the probability to draw a patch in A and the probability to draw a patch in B.

Let a be the event that a point $x \in A$ is randomly chosen and b the event that a point $x \in B$ is chosen. Let further s be the event that a point is accepted, then $P(a)$ is the probability to draw a point $x \in A$ and $P(b)$ the probability to draw a point $x \in B$. Consequently $P(s \mid a)$ is the conditional probability that the point is accepted when a point from A is drawn, respectively $P(s \mid b)$ the very same probability when a point from B is drawn. These probabilities are the a priori values attributed to the voxels in our map. Using the law of Bayes it is possible to calculate the probability $P(a \mid s)$

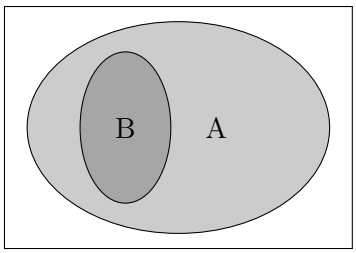


Figure 8.3.: Illustration of two regions, A and B, in the image. When drawing coordinates at random we will obviously obtain more patches in area A than in area B. $\{x \in A\} \subset \{x \in B\} = \emptyset$.

for an accepted point being a point $x \in A$ and $P(b \mid s)$ being a point $x \in B$:

$$P(a \mid s) = \frac{P(s \mid a)P(a)}{P(s)} \tag{8.3}$$

$$P(b \mid s) = \frac{P(s \mid b)P(b)}{P(s)} \tag{8.4}$$

As we are interested in the proportions that the patches are drawn in the random drawing process, described in the following section, we calculate the fraction of equation (8.3) and (8.4)

$$\frac{P(a \mid s)}{P(b \mid s)} = \frac{P(s \mid a)}{P(s \mid b)} \frac{P(a)}{P(b)}. \tag{8.5}$$

To achieve certain ratios between VOI patches and patches not in the VOI, *i.e.* $\frac{P(a|s)}{P(b|s)}$, with a the VOI and b the rest of the brain, we solve equation (8.5) for $P(s \mid b)$, fixing $P(s \mid a)$ at a value close to one and calculate $P(s \mid b)$. This equation also immediately shows that changes will get very small when augmenting the ratio beyond 1 as we are dealing with a $\frac{1}{x}$ relation.

Example: suppose we want to draw as many patches in the parietal lobe than in the rest of the brain. So we set in equation (8.5) $P(a \mid s) = P(b \mid s)$ with a the event of drawing a point in the parietal lobe and b the event of drawing a point outside the parietal lobe. Then $\frac{P(a)}{P(b)}$ is the ratio between the sizes of parietal lobe and the rest of the brain. Now we can see that $\frac{P(s|b)}{P(s|a)} = \frac{P(a)}{P(b)} \approx 0.2$. If we set $P(s \mid a) = 1$ then $P(s \mid b) \approx 0.2$.

8.1.3. Critical remarks

Spatial importance sampling allows us to reduce the number of patches that are analyzed in areas of low importance to the classification without excluding them completely from the analysis. This is an advantage over the application of a threshold in cases we

are not absolutely sure that areas of low importance, specifically those below a possible threshold, are not necessary in all the instances we want to analyze. A disadvantage of importance sampling is that there is a possibility that not all pixels in areas of high importance are included into the analysis especially if the number of patches drawn is chosen to small. In methods dividing the image in a regular grid of non-overlapping patches or overlapping patches and thresholding the patches to use in contrary the inclusion of all pixels of high importance is assured. Success of the importance sampling strategy depends on a well chosen number of patches.

8.2. Predefined regions of interest

From histopathologic examinations the regions that are affected in the late stadium of AD are known. As it is extremely improbable that those zones start being affected by AD only in the late stages of the disease we can assume that the same regions found in histopathologic examinations show also signs of AD in its early onset. Publications in the medical community state congruently the occurrence of hypo-metabolism in the temporal and parietal lobe which is confirmed by histopathological findings [Hoffman et al., 2000, Silverman, 2009]. This knowledge about the affected brain regions can be used to focus a patch based analysis to the regions that are most probable to be affected.

In patch based analysis the amount of patches is often a problem as we can define as many patches in an image as there are pixels. A common method to reduce the amount of patches is to draw patches according to a uniform distribution over the whole image. In the case of the brain scans it is evident that this approach will provide a quite poor sample of the information in the image as there are lots of patches that are located outside the brain volume. It is also clear that it is most promising to examine in the first place the regions that are most probable to be affected. But the rest of the brain should not be neglected absolutely as it might also contain relevant information or provide at least a reference intensity. For this reason a probability atlas for the random selection was created and the method for spatial importance sampling, presented in section 8.1, was used to draw the patches. The probability atlas is used to exclude the outside of the brain from the analysis and to draw the patches for the analysis in a non-uniform way in the brain matter that can be determined as desired.

The first Version of the atlas was based on the electronic Talairach brain atlas created by Lancaster et al. [2000] which is derived from the brain atlas of Talairach and Tournoux [1988]. Some more detail about this brain atlas can be found in subsection 4.3.1. It provides the connection between semantic expressions for the single brain regions and pixels in the brain image. By inversion of this tool it is possible to select brain regions by semantic expressions. This enables the user also to create a brain

map containing the probability values for the occurrence of AD. The problem with this approach is the fact that the final atlas is in Talairach space whereas the images are in MNI space. Therefore, a normalization of the atlas to the MNI space is necessary. An additional spatial normalization of the created atlas, however, might cause undesired probability values and consumes a lot of time if many different versions of the atlas are to be tested e.g. in an attempt to recursively improve the atlas. Slices of this first version of the atlas are shown in Figure 8.4a.

The second Version of the atlas is created with the WFU PickAtlas that provides access to the Talairach daemon. Using this tool it is possible to generate a brain mask based on the Talairach brain atlas by selecting the regions in a GUI. Another big advantage is that the mask is generated directly in MNI space which makes a spatial normalization superfluous. The masks for each desired brain region can be saved as a indexed image and the probability values adopted without effort. This second version overcomes all drawbacks of the first version of the map and is therefore used for the following experiments. Slices of the improved atlas are shown in Figure 8.4b.

8.2.1. Experiment protocol

To evaluate the quality of the method, 8 different probability maps were created for both parietal and temporal lobe. The values in the atlas were chosen such that the ratio between the probability of selecting a patch in the ROI and in the rest of the brain assumes the values

$$\frac{P(x \in \text{ROI}|a)}{P(x \in \text{rest of the brain}|a)} \in \{0, 0.05, 0.10.15, 0.2, 0.25, 0.5, 0.75, 1\} \tag{8.6}$$

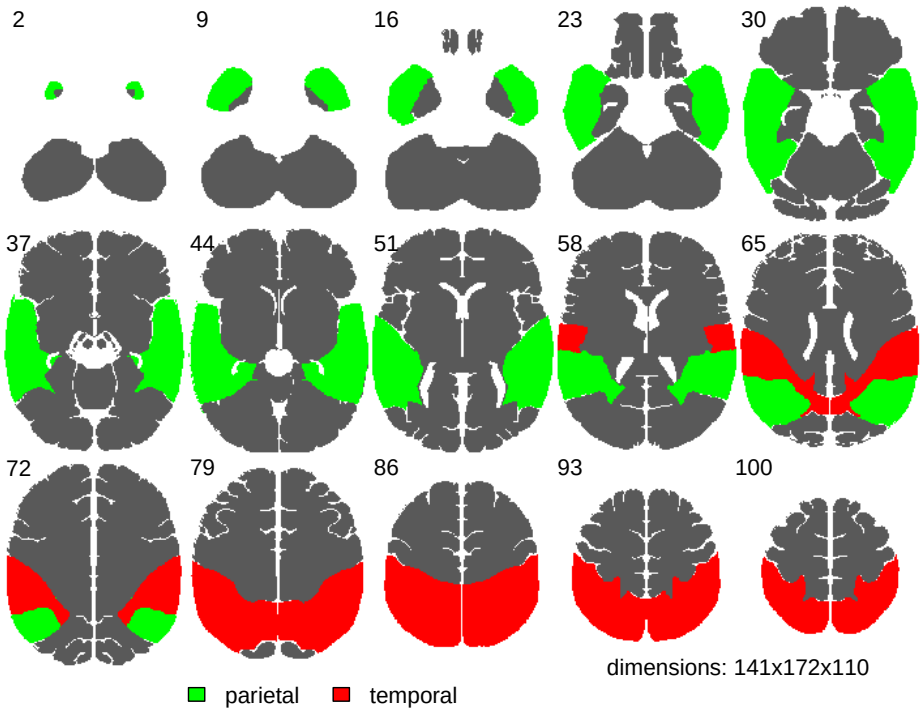
Using the method described in subsection 8.1.2 we are able to take the different sizes of the zones into account and to calculate the values that have to be attributed to the voxels in each zone of the brain.

Afterwards patch drawing according to the selected ratio of patches in the ROI and in the rest of the brain was performed. To obtain statistically sound results 100 fold bootstrapping was performed. This allows despite the small amount of round 80 examples per class good prediction of classification accuracy. Calculations were performed using different patch sizes using as well 3D as 2D patches.

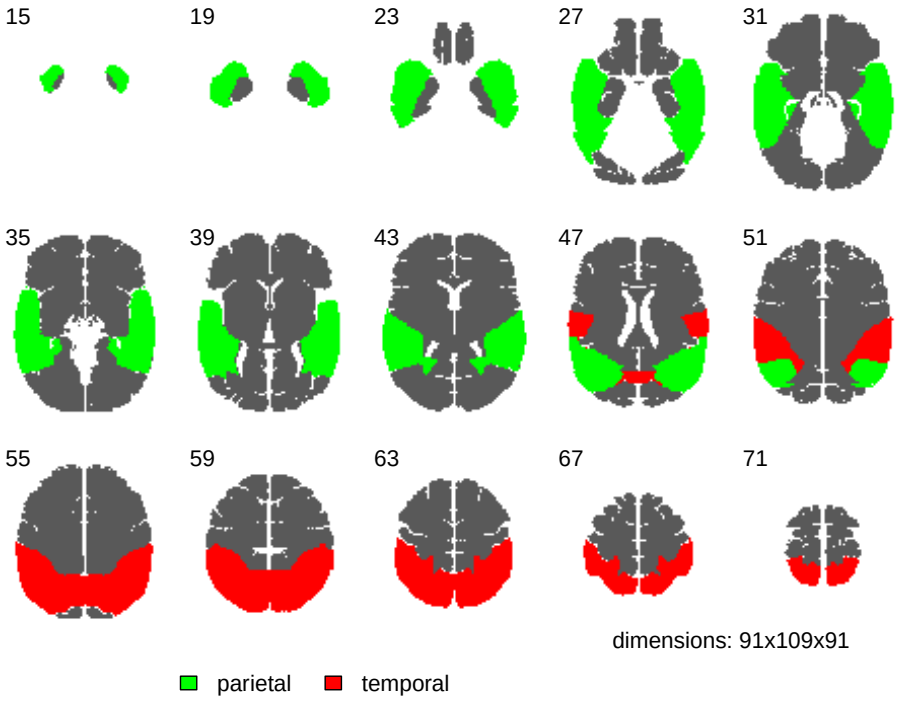
Under the assumption we have correctly chosen a ROI the design of the experiment makes us expect an augmented error rate for

$$\frac{P(x \in \text{ROI}|a)}{P(x \in \text{rest of the brain}|a)} < \frac{V(\text{ROI})}{V(\text{rest of the brain}))}, \tag{8.7}$$

where $V(\cdot)$ denotes the volume of an area in the brain. The quotient of volumes $\frac{V(\text{ROI}}{V(\text{rest of the brain})}$ corresponds to the ratio of patches drawn in the corresponding regions



(a) Version 1: created by the electronic Talairach atlas in Talairach space



(b) Version 2: created by WFU pickatlas in MNI space

Figure 8.4.: Atlas for the occurrence of Alzheimer's disease in the human brain according to literature findings. Three zones in the brain: temporal lobe, parietal lobe and the rest of the brain.

in the brain that would be obtained by drawing at random without spatial importance sampling. On the other hand we expect a reduction of the error rate for

$$\frac{P(x \in \text{ROI}|a)}{P(x \in \text{rest of the brain}|a)} > \frac{V(\text{ROI})}{V(\text{rest of the brain}))} \tag{8.8}$$

due to a reduction of the amount of unnecessary information passed to the classifier. This reduction of the error rate is assumed to be limited by the quality of the chosen feature.

If we assume in contrary a badly chosen ROI we can expect lower error rates if Equation 8.8 is true as if Equation 8.7 is true, as in the latter case significant patches would be more likely not to be drawn.

In this experiment a RF classifier was used with $T = 500$ trees and $q = \sqrt{\#\text{of features}}$ variables tried to find the best split.

8.2.2. Results

All results shown are obtained with the second version of the importance map and 400 patches. Results using 3D patches of size $4 \times 4 \times 4$ and $6 \times 6 \times 6$ are shown in Figure 8.5 and using 2D patches of size 4×4 and 6×6 in Figure 8.6. The dashed vertical line in each figure marks the ratio of patches that would be obtained by sampling the patches uniformly in the brain, i.e. without spatial importance sampling. This ratio corresponds to the relative size of the ROI and the rest of the brain. Result tables for the whole range of 3D patches from size $2 \times 2 \times 2$ to $7 \times 7 \times 7$ and 2D patches from size 3×3 to 7×7 are shown in Table B.1 and Table B.2 which are located in Appendix B.

For the *parietal* lobe we obtain a classification rate development as expected for a correctly chosen ROI (see Figures 8.5a, 8.5c, 8.6a and 8.6c). The classification error raised when the probability of drawing patches in the parietal lobe is reduced in comparison to the probability of random drawing. When, in contrary, increasing the probability of drawing patches in the parietal lobe parietal lobe the classification error decreased in comparison to random drawing. It is remarkable that, while the FNR (rate of misclassified AD cases) remains almost unchanged at reduced probability of drawing in parietal lobe, the FPR (the rate of misclassified NC cases) augments significantly.

For the ROI *temporal* lobe the obtained classification rate development corresponds more or less to a badly chosen ROI. In Figures 8.5b, 8.5d, 8.6b and 8.6d we observe only a insignificant raise in classification error when diminishing the rate of drawing patches in temporal lobe below the rate of random drawing. Furthermore, we observe an increase of the FPR when augmenting the ratio of patches drawn in temporal lobes that exceeds the ratio of drawing at random. This implies that patches outside temporal lobe that are important for the classification of AD are not drawn anymore.

In both cases, temporal and parietal lobe, results show no significant difference between the use of 3D and 2D patches. Using the mean intensity of the patches as

feature on the smooth PET images this behavior is rather expected than surprising. However a more elaborate feature than the used one could take profit of 3D patches.

8.3. Sampling with learned importance map

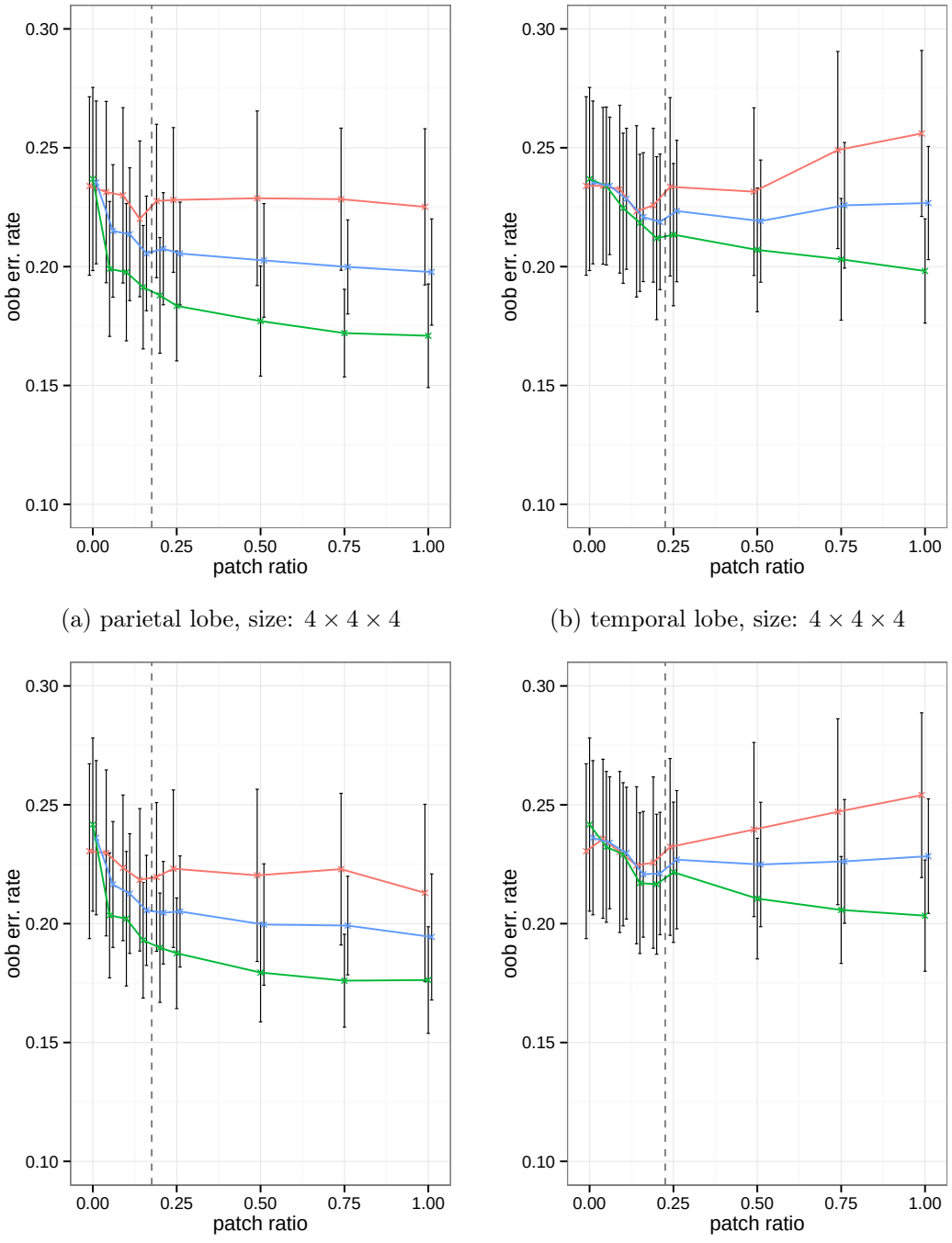
We will now combine the method for learning importance maps from data (see chapter 2) and the spatial importance sampling method (see section 8.1). We use the importance map we have already extracted from our data set in chapter 6. We will compare the results of spatial importance sample using the RF importance map with uniform sampling and with the results obtained in the previous section 8.2. This will allow us to asses the quality of the extracted maps.

8.3.1. Experimental setup

For the comparison result we draw samples of patches uniformly distributed over the brain and distributed according to the inter-class variance. Uniform sampling is achieved by using the importance sampling algorithm with a brain mask as importance map for uniform sampling. This causes coordinates inside the brain volume to be accepted with probability $p = 1$ and coordinates outside the brain to be accepted with probability $p = 0$. Sampling according to the inter-class variance is also achieved using the importance sampling algorithm but using the inter-class variance normalized to the interval $[0; 1]$ as map. For each selected patch we calculate the mean intensity as feature. The so formed feature vector for all images and the corresponding labels are fed to a RF classifier. This procedure is repeated 100 times to calculate a statistics. Remember that a cross-validation is not necessary as the RF internally calculates an estimate for the prediction error. We used again RFs with $T = 500$ trees and $q = \sqrt{\#\text{of features}}$ variables tried to find the best split.

For the sampling according to the importance map of chapter 6 we use the importance sampling algorithm as well. With the importance map used the patches will be distributed according to the map shown in Figure 6.2a. Classification is done in the exactly same way as for the comparison result.

We will repeat the classification test with 100, 200, 300, 400, 500 and 1000 patches. To be able to compare the results with results of previous chapters we use a patch size of $4 \times 4 \times 4$ voxels. The difference in classification error and standard deviation of the classification error due to the patch sampling will show the gain obtained due to the importance sampling.



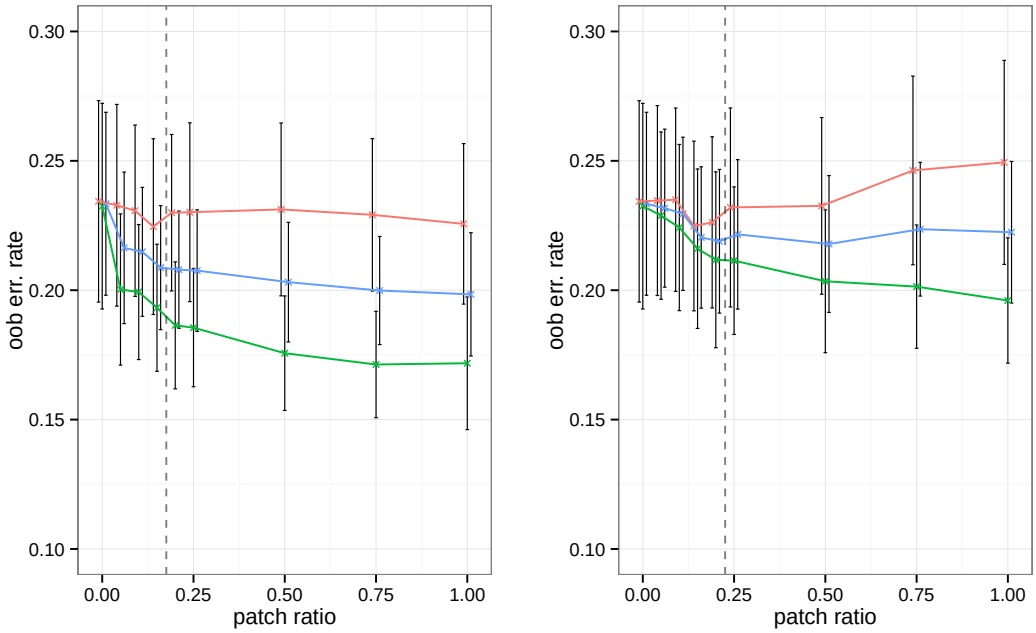
(a) parietal lobe, size: $4 \times 4 \times 4$

(b) temporal lobe, size: $4 \times 4 \times 4$

(c) parietal lobe, size: $6 \times 6 \times 6$

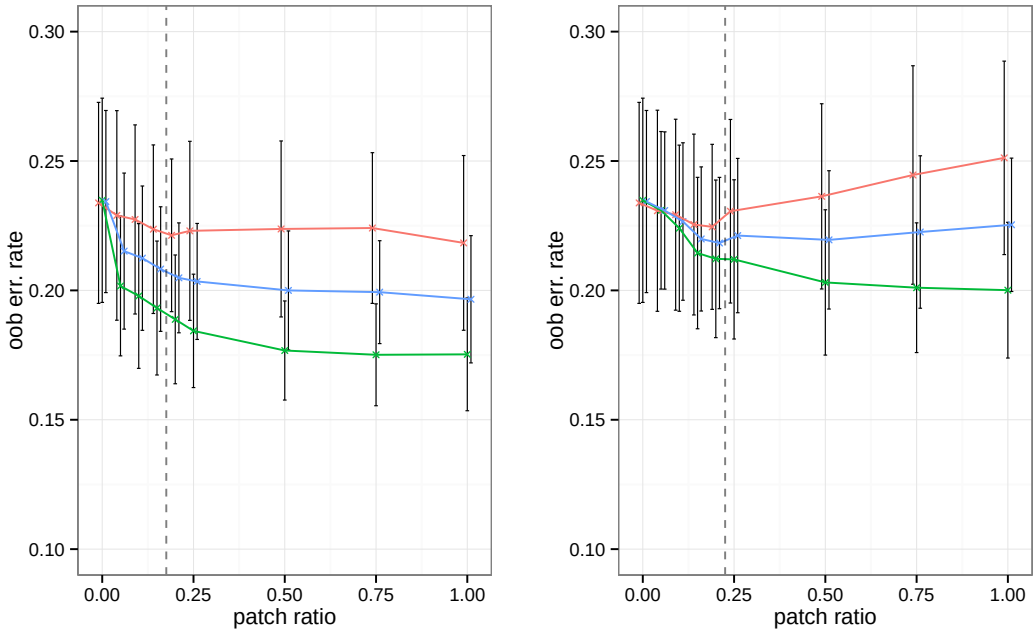
(d) temporal lobe, size: $6 \times 6 \times 6$

Figure 8.5.: Spatial importance sampling of 3D-patches; oob. error rate versus ratio between ROI patches and not ROI images. blue: ooB error rate, red: FNR, green: FPR. The vertical line marks the ratio of ROI volume to volume of the rest of the brain which corresponds to the ratio that would be obtained by random drawing.



(a) parietal lobe, size: 4×4

(b) temporal lobe, size: 4×4



(c) parietal lobe, size: 6×6

(d) temporal lobe, size: 6×6

Figure 8.6.: Spatial importance sampling of 2D-patches; oob. error rate versus ratio between ROI patches and not ROI images. blue: ooB error rate, red: FNR, green: FPR. The vertical line marks the ratio of ROI volume to volume of the rest of the brain which corresponds to the ratio that would be obtained by random drawing.

8.3.2. Results

Figure 8.7 shows the plot comparing classification using uniform sampling, importance sampling and sampling according to the inter-class variance. The uniform sampling provides as expected a poorer classification result than the importance sampling. Classification error grows significantly when reducing the number of patches when using uniform sampling (by about 5%). When using importance sampling or sampling according to the inter-class variance, in contrary, the effect of reducing the amount of patches is insignificant. Importance sampling and sampling according to the inter-class variance result in the same classification error. Table 8.1 contains the numerical results of the plot. The standard deviation of the results obtained with importance sampling and sampling according to inter-class variance are only half of the standard deviation of the results obtained with uniform distributed patch locations. The lower variance of the classification result and the lower ooB error combined show clearly that the importance map and the inter-class variance restricts the patch drawing to areas that are important for the classification.

Comparison of the results obtained with this experimental setup and the results of section 8.2 shows sampling the importance map results in lower classification error and lower variance than using the pre-defined ROIs parietal and temporal lobe.

8.4. Discussion

The comparison of uniform sampling and spatial importance sampling according to the RF importance map shows the clear superiority of the later. The classification error of uniform sampling is, as expected, decreasing with augmenting numbers of patches. When sampling according to the RF importance map, in contrary, we observed no significant increase in classification error even for small numbers of patches. The standard deviation of the classification error behaves in the same way, which shows that the information content of the small samples varies more when drawn with uniform sampling than with importance sampling.

Comparing the results in section 8.3 with the results in section 8.2 we can observe that the classification error obtained with the RF importance map, as well as the standard deviation of the error, is lower than with the maps for the predefined ROIs parietal and temporal lobe. The classification error obtained in section 8.3 is lower by ≈ 0.03 and the standard deviation only half than with predefined ROIs. This observations lead to the conclusion that the RF importance map extracted from the data set in chapter 6 are a better approximation of the distribution of classification relevant information then the predefined ROIs parietal and temporal lobe.

In chapter 6 we have also classified our data using $4 \times 4 \times 4$ patches. There a regular grid of non-overlapping patches, resulting for this size in ≈ 13000 patches, was used.

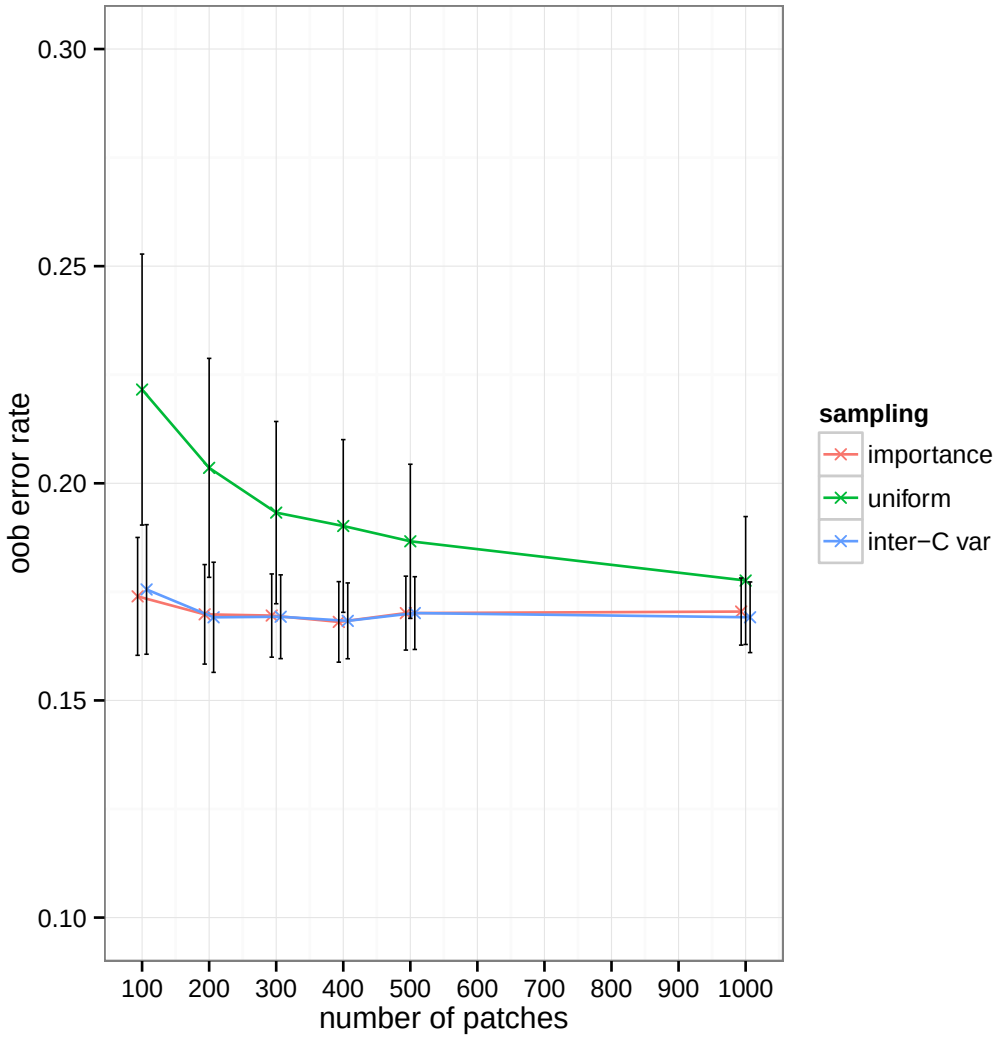


Figure 8.7.: Oob. error rate versus number of selected patches. Importance sampling with map extracted from data compared with uniform sampling and inter-class variance (inter-C var).

# of patches	oob err. rate	FPR	FNR
100	0.174 ± 0.014	0.156 ± 0.014	0.192 ± 0.022
200	0.170 ± 0.011	0.153 ± 0.014	0.187 ± 0.016
300	0.170 ± 0.010	0.154 ± 0.013	0.185 ± 0.014
400	$\mathbf{0.168 \pm 0.009}$	0.152 ± 0.012	0.185 ± 0.013
500	0.170 ± 0.009	0.156 ± 0.011	0.184 ± 0.013
1000	0.170 ± 0.008	0.157 ± 0.010	0.185 ± 0.011

(a) importance sampling

# of patches	oob err. rate	FPR	FNR
100	0.222 ± 0.031	0.204 ± 0.032	0.239 ± 0.040
200	0.204 ± 0.025	0.184 ± 0.025	0.223 ± 0.037
300	0.193 ± 0.021	0.173 ± 0.023	0.214 ± 0.030
400	0.190 ± 0.020	0.168 ± 0.021	0.213 ± 0.028
500	0.187 ± 0.018	0.166 ± 0.016	0.207 ± 0.028
1000	$\mathbf{0.178 \pm 0.015}$	0.161 ± 0.016	0.195 ± 0.022

(b) uniform distributed

# of patches	oob err. rate	FPR	FNR
100	0.176 ± 0.015	0.157 ± 0.016	0.194 ± 0.020
200	0.169 ± 0.013	0.157 ± 0.015	0.182 ± 0.017
300	0.169 ± 0.010	0.151 ± 0.013	0.188 ± 0.014
400	$\mathbf{0.168 \pm 0.009}$	0.152 ± 0.011	0.185 ± 0.013
500	0.170 ± 0.008	0.155 ± 0.010	0.186 ± 0.012
1000	0.169 ± 0.008	0.157 ± 0.010	0.182 ± 0.013

(c) distributed according to inter-class variance

Table 8.1.: Comparison of classification results obtained with uniform and importance sampling. Patch size $4 \times 4 \times 4$. Statistics over 100 samples of patches.

The error rate obtained was 0.185 ± 0.010. Using spatial importance sampling and the importance map extracted from the data set we obtained with a number of only 400 patches a error rate of only 0.168 ± 0.009. This corresponds to a reduction of the number of patches to $\approx 3\%$ of the number of patches used in the extraction of the importance map in chapter 6.

9. Classification of Alzheimer's disease via NTF decomposition

In this section we will present our findings in the application of non-negative tensor factorization (NTF) as feature extraction method in the classification of FDG PET scans of AD patients. We consider each scan in our data base to be a mixture of basis components. These basis components will be *learned* from the data by the non-negative sub-tensor set factorization (N_STS_F) algorithm. As we are using an NTF algorithm we will obtain basis components that are strictly non-negative and can therefore easily be visualized as basis images. The corresponding weights in this new basis of representation as well will be non-negative. With a large enough number of scans the obtained basis will generalize to new examples and the weights of the decomposition will appropriately represent the images. In this way we are able to reduce the dimensionality of our problem. The dimensionality in the voxel representation is the number of voxels, i.e. in our case approximately 900 000. Changing the basis using the decomposition the dimensionality reduces to the *rank* R of the decomposition. The choice of R is rather heuristic. We can, however, use the Akaike information criterion (AIC) or the spectrum of eigenvalues obtained by a PCA to determine an initial guess. Experience tells us that an R smaller than the biggest dimension of the input tensor is sufficient. Therefore we can achieve in our case a dimension reduction by a factor of ≈ 9000. In comparison to a basis obtained by matrix methods, such as PCA, independent component analysis (ICA) or NMF, we have also the advantage that the basis obtained by tensor methods is much smaller in terms of used disk space.

9.1. Experiment protocol

The 335 images of the ADNI data set were decomposed using the N_STS_F algorithm presented in section 3.4. The brain scan data is presented to the N_STS_F algorithm both as a 4^{th} order tensor of dimensions $91 \times 109 \times 91 \times 335$ and as a 3^{rd} order tensor of dimensions $109 \cdot 91 \times 91 \times 335$ formed by vectorizing the transversal slices of each brain scan. The decomposition performed by the N_STS_F algorithm consists of 4, respectively 3, factor matrices, depending on the order of the input tensor. The last factor matrix encodes the scans of the different patients in terms of the first 3, respectively 2, factor matrices and will be called the *encoding matrix* henceforth. The other factor matrices together form the decomposition basis and will be called the *basis factor matrices*.

Each basis factor matrix corresponds to one mode of the decomposed tensor. For visualization we transform the basis factor matrices to basis images, which are easier to interpret. The basis images I_r are obtained by

$$I_r = \left[\!\left[\boldsymbol{A}^{(1)}, \ldots, \boldsymbol{A}^{(N-1)}_{:r} \right]\!\right] \quad \in \mathbb{R}^{I_1,\ldots,I_{N-1}}, r = 1, \ldots, R \tag{9.1}$$

where R is at the same time the rank of the decomposition and the number of basis images. In the case of the 3rd order input tensor we also have to revert the vectorization of the transversal slices. The basis images do not have a specific order as it is the case in e.g. PCA or ICA. The basis images of the decomposition without vectorization is shown in Figure B.1 located in Appendix B.

In the case of the 4th order representation the learned image basis has a memory footprint of $291 \cdot R$ and in the case of 3rd order representation $10\,010 \cdot R$. Compared to a size of $902\,628 \cdot R$ of a basis learned by NMF we have achieved a considerable reduction.

After decomposition the encoding matrix is passed alongside with the class labels to train a classifier. In the case of the 4th order tensor this matrix is the 4th factor matrix and in the case of the 3rd order tensor it is the 3rd factor matrix.

For a single patient classification is not necessary to perform a new the decomposition. It is sufficient to optimize the encoding matrix of the patient which is in this case actually a vector. This vector is passed to the already trained classifier. The time used for this calculation is much shorter than a decomposition, which makes the feature extraction by a pre-calculated basis interesting for real life applications.

In classification tests including $\mathrm{MCI_{AD}}$ subjects the size of the other involved classes has to be reduced to the number of $\mathrm{MCI_{AD}}$ subjects available as a imbalanced training and testing set could influence the outcome of the test. We have realized this by drawing bootstrap samples of the size of the $\mathrm{MCI_{AD}}$ group and using the remaining scans for evaluation of the trained classifier. This solution has the advantage that we can profit of the variety of all available scans and therefore obtain the best possible error estimate for the classification of an unknown example.

We will examine a RF and a SVM classifier in a two-class classification set up. We will afterwards analyze the specificities of MCI and $\mathrm{MCI_{AD}}$ patients based on classifiers trained on NC and AD patients only. To obtain information about the patients we will train the SVM on a sample of NC and AD scans and subsequently retrieve the classifiers response to both, the remaining NC and AD scans, and the MCI and $\mathrm{MCI_{AD}}$ cases. The in this way trained classifier will attribute either NC or AD label to the MCI and $\mathrm{MCI_{AD}}$ cases. This procedure will be implemented as a bootstrap to obtain a representative statistics. We expect the classifier to attribute AD label to the $\mathrm{MCI_{AD}}$ cases as the patient developed AD during the follow-up period. For the MCI cases we rather expect a NC label to be attributed by the classifier.

For these experiments we used the classifiers with their default parameters, i.e. SVM with RBF and the kernel parameters $\gamma = \frac{1}{\#\text{of features}}$ and $C = 1$ and the RF with $T = 500$ trees and $q = \sqrt{\#\text{of features}}$ tried to find the best split at each node. Please note that optimization of the parameters would make it necessary to perform a nested crossvalidation in order to exclude a positive bias on the estimate of prediction error.

9.2. Results

A decomposition of this large data tensor (approx. 302 million elements) with other NTF algorithms[1] was impossible due to a lack of memory. Using $\text{N}_\text{S}\text{T}_\text{S}$F, however, a decomposition is possible and took approx. 10 hours on a 2.7GHZ dual core machine with 4G of RAM. The relative decomposition error for the 4D tensor was $C_{\text{rel.}} = 0.1473$ and for the 3D tensor $C_{rel.} = 0.1046$.

The basis images shown in Figure B.1 are mostly symmetric to the inter hemispherical plane, but some basis images are not. This reflects the high symmetry of the brain metabolism but by means of the available asymmetric basis images it is also possible to approximate slight asymmetries. As can be seen in the visualization the basis images are sparse as expected from a non-negative decomposition.

In Table 9.1 we show the classification results NC vs. AD using $\text{N}_\text{S}\text{T}_\text{S}$F decomposition as feature extraction means. We can see that the classification error is lower for the input tensor of order 3, i.e. when transversal slices have been vectorized. Furthermore, the SVM has outperformed the RF. The obtained classification errors are competitive with the classification errors of the previous experiments.

classifier	input order	oob-err	FPR	FNR
RF	3	0.187	0.177	0.196
RF	4	0.237	0.242	0.231
SVM	3	**0.074**	0.070	0.078
SVM	4	0.194	0.180	0.211

Table 9.1.: Classification of NC vs. AD using $\text{N}_\text{S}\text{T}_\text{S}$F as feature extraction means. RF and SVM have been used with the factor matrix corresponding to the patients as feature matrix.

Given this good classification results we also performed classification test for the other possible class combinations. The results of these test are shown in Table 9.2. These class combinations are more or less interesting with regard to early detection of AD.

[1] We tested the algorithms used in subsection 3.4.6.

The most interesting is the classification of MCI vs. MCI_{AD}. A successful classification of these classes allows to discriminate future AD patients from the MCI group. It is important to remember, that in this classification test only the baseline scan of each patient was used. This means that the SVM trained with the N_STsF features was capable to discriminate MCI from MCI_{AD} cases with an error of 15.3 %, where medical doctors found the MCI_{AD} cases to be different only more than 12 months later.

The classification of MCI itself is of lower interest as it only is a symptom. The underlying disease, however, is not clearly determined. The remaining class combinations are displayed for completeness.

	oob-err	FPR	FNR
MCI / MCI_{AD}	**0.153**	0.148	0.158
NC / MCI	0.162	0.174	0.148
MCI / AD	0.140	0.119	0.161
NC / MCI_{AD}	**0.082**	0.073	0.087
AD / MCI_{AD}	0.224	0.232	0.217

(a) Classification by SVM using N_STsF decomposition *with* slice vectorization, i.e. decomposition of order 3 tensor.

	oob-err	FPR	FNR
MCI / MCI_{AD}	0.212	0.203	0.209
NC / MCI	0.198	0.174	0.227
MCI / AD	0.191	0.187	0.197
NC / MCI_{AD}	0.218	0.202	0.230
AD / MCI_{AD}	0.201	0.183	0.216

(b) Classification by SVM using N_STsF decomposition *without* vectorization, i.e. decomposition of order 4 tensor.

	oob-err	FPR	FNR
MCI / MCI_{AD}	0.323	0.290	0.356
NC / MCI	0.357	0.376	0.339
MCI / AD	0.282	0.260	0.303
NC / MCI_{AD}	0.219	0.167	0.271
AD / MCI_{AD}	0.619	0.678	0.559

(c) Classification by RF using N_STsF decomposition *with* vectorization, i.e. decomposition of order 3 tensor.

	oob-err	FPR	FNR
MCI / MCI_{AD}	0.389	0.378	0.401
NC / MCI	0.379	0.415	0.342
MCI / AD	0.336	0.332	0.340
NC / MCI_{AD}	0.265	0.252	0.278
AD / MCI_{AD}	0.582	0.657	0.508

(d) Classification by RF using N_STsF decomposition *without* vectorization, i.e. decomposition of order 4 tensor.

Table 9.2.: Classification results obtained using N_STsF for feature extraction. The first column in the tables indicates the combination of classes, where the first class is considered the negative case and the second the positive case.

In all classifications the result of SVM was superior compared with the result obtained with RF. The SVM classifier outperformed the RF by at least 4%. In the classification AD vs. MCI_{AD} the RF failed completely with an error rate > 0.5 (see Table 9.2c and Table 9.2d row 5) while the SVM achieved an error rate just above 20 %. However, this difference should not be overrated, as the MCI_{AD} cases became actual AD cases within 2 years after the acquisition of the scans used in the classification.

Comparing results with vectorization of the slices and without vectorization we observe an advantage of vectorization when using SVMs (see Table 9.2a and Table 9.2b). AD vs. MCI_{AD} is the only case where the classification error is slightly bigger in the vectorized case than in the non-vectorized case. For all other combinations of classes error rate is lower when vectorizing the slices.

When using RFs we observe in contrary an advantage of not vectorizing of the slices. However, the overall error rates are much higher as in the SVM case and thus less conclusive.

To get closer to the ultimate goal, the early detection of AD we performed a more detailed examination of this classification approach using the more promising SVM classifier only. For the medical doctor it is most important to diagnose AD in its earliest stages to improve the prospect for his patient. The diagnosis MCI is rather unsatisfactory as many MCI cases remain stable, but about 15% of MCI patients are diagnosed AD within 2 years. Therefore, we want to examine the results of a classifier trained for NC vs. AD classification patient by patient when classifying all four available classes.

We trained the classifier using bootstrap samples of NC and AD cases only and classified the available MCI and MCI_{AD} scans alongside with the ooB scans. We performed 10 000 bootstraps of training and testing and recorded the frequency of AD labels attributed to each patient for all four classes. In Figure 9.1 we show the histogram for the distribution of AD labels over the patients.

This histogram shows that for the NC cases the trained classifiers attributed for almost all patients the correct label in all bootstraps. For the AD cases almost all patients were correctly classified in more then 80% of the bootstraps. This shows that our classification approach produces stable results regardless the subset of images we trained on. Also most of the MCI_{AD} patients are classified in more than 80% of the bootstraps as AD. For the MCI cases we observe that most of the patients are constantly classified either as NC or AD. Only a smaller part of patients has no stable label over the performed bootstraps. There are 3 NC, 1 AD and 1 MCI_{AD} scans that are classified wrong in 95% of the cases.

9.3. Discussion

The presented results show that the feature extraction by decomposing the images into basis images and presenting the corresponding encoding factors to a classifier results in state of the art AD classification performance.

In chapter 3 we argued that a vectorization of images induces a loss of information. The classification result obtained in this section do not supports this position as error rates diminish when vectorizing the slices of the scans. A possible explanation for

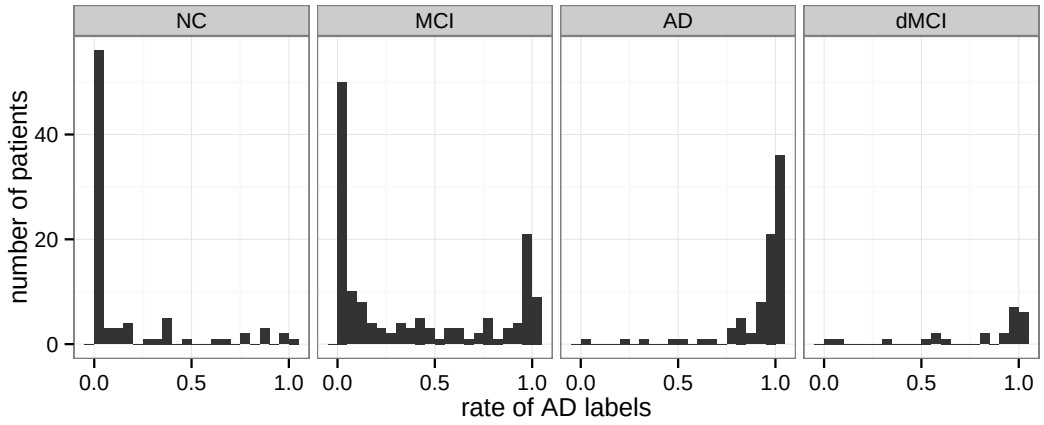


Figure 9.1.: Labels attributed to patients by the SVM classifier. The classifier was trained on samples of NC and AD scans and used to classify the remainder of scans. Training and testing was repeated 10000 times.

the reduction of the error rate is a possible better performance of the decomposition algorithm on third order tensors than on fourth order tensors.

As expected classification rates are highest for NC vs. AD and NC vs. MCI_{AD} classification (92.6% and 91.8% respectively) as the difference between those classes are the biggest. Lowest classification rates were obtained for the classification MCI_{AD} vs. AD which could also be considered one group as the latter cases developed AD during the follow-up period of 2 years. The classification of MCI vs. MCI_{AD} is, however, most interesting for the early detection of AD. The achieved classification rate of 84.7% proves our approach to be a promising step towards reliable early detection of AD.

Training SVMs on samples of NC and AD scans only we observed the classification result for each single patient which proved our approach to be very stable with respect to the images the classifier is trained on. Most of the MCI_{AD} scan were classified by almost all trained SVMs as AD and only 12.5% of MCI_{AD} scans were classified by the majority of SVMs as NC. For the MCI cases we observed much more patients that are differently classified by different SVMs. There are also about 20% of MCI cases that are classified in more than 80% of trained SVMs as AD. This leads to the conclusion, that the MCI group really is hard to classify and likely contains patients developing AD after the 2 years follow-up period.

9.4. Summary

In this section we presented the use of NTF as a feature extraction means. We decomposed large 3^{rd} and 4^{th} order tensors using the new N_STS_F algorithm in order to obtain a new, lower dimensional basis for the representation of the data. This approach produced discriminative features that allowed to train robust SVM and RF classifiers attaining state of the art classification rates for AD. The classification rate for NC vs. AD was as high as 92.6%. The classification rate of 84.7% for a classification of MCI vs. MCI_{AD} is a promising step towards AD early-detection.

10. Conclusion

In this thesis we have developed new methods for the analysis of volumetric images and tested their application on real life data in the classification of Alzheimer's disease (AD). We have presented the developed methods in Part I as application independent tools. This allows their application to all kind of volumetric images like computed tomography (CT), magnetic resonance imaging (MRI), single photon emission computed tomography (SPECT) and positron emission tomography (PET). The presented methods cover three key problems of image classification: *1*) The determination of the location of discriminative information; *2*) the reduction of the dimensionality of the classification problem; and *3*) the extraction of discriminant image features. In Part II we demonstrated possibilities of application of the presented methods in the early detection of AD.

The early detection of AD using fluorodeoxyglucose (FDG) PET scans poses a series of problems: *1)* The scans have a poor signal to noise ratio (SNR). *2)* The knowledge about the location of affected brain areas lacks precision, because of variation between the patients. *3)* The diagnosis largely relies on the experience of the clinician, not on quantifiable measures or a specified protocol for the evaluation of the scan. Nevertheless, FDG PET is considered to be a promising means for the early detection of AD. For the computerized analysis additional problems emerge: *1)* Anatomic differences between patients; *2)* label uncertainty; *3)* 3 dimensionality of the images; *4)* low number of scans available; and *5)* large data volume.

Our first step taken to overcome the expected problems was a thorough preparation of an image data set of FDG PET scans obtained from the Alzheimer's Disease Neuroimaging Initiative (ADNI) (see section 5.4). Two years of follow-up examinations were analyzed in order to obtain highest confidence in the class labels possible. This analysis also allowed us to split the group of mild cognitive impairment (MCI) patients into patients that are stable over the follow-up period and patients which are diagnosed AD within the follow-up period. The latter patients form the new mild cognitive impairement converting to Alzheimer's disease ($\mathrm{MCI_{AD}}$) group including 24 patients, as expected according to AD literature about 15 % of the whole MCI population. This newly formed group of patients allows to attempt a detection of AD using the available brain image data well *before* a clinical diagnosis was possible. Though the MCI group is split into two groups[1] in ADNI2, the continuance of ADNI, this split is based on

[1] Early and late MCI are distinguished by a score of the Wechsler memory scale.

neurophysiological examinations at enrollment. We consider the use of the followup examinations, especially for the MCI cases, very important, as the baseline examination might already show signs of the medical condition at the end of the follow-up period. Biomarkers of an AD patient acquired before the AD was diagnosable is of enormous value for the development of early detection methods. For the preprocessing steps necessary for a computerized analysis, spatial normalization and intensity normalization, we used readily existing and established methods.

Our second step was to present an approach to retrieve information about the location of classification relevant information directly from image data (method in chapter 2, application in chapter 6). For this kind of analysis usually voxel-wise statistics are used. The main disadvantage of this approach is that the voxels are analyzed independently from each other. In consequence class differences that consists in patterns are not detected. Therefore, we proposed and examined an approach that is based on patch based classification with random forests (RFs). This information extraction algorithm returns a map specific to the performed classification with respect to both the patch size and the feature calculated on each patch. But most important, it also includes the patterns formed by the features in the resulting *importance map*.

We applied the developed method to our data set using the mean intensity of each patch as feature. We performed the map extraction for different sizes of patches. The RFs with highest classification rate of 83.0 % was obtained with a patch size of $2 \times 2 \times 2$ voxels, which corresponds to a cube of $4 \times 4 \times 4$ mm. Comparison of the obtained importance map with a map of inter-class variance showed high similarity. All areas of high importance in our map are also areas of high inter-class variance, but they cover a smaller area. These areas of highest importance are located in posterior cingulate cortex and precuneus, which is also in accordance with medical literature. There are also some voxels that do not belong to literature defined regions of interest (ROIs), e.g. the inter-hemispheric fissure. Surprisingly the hippocampus, which is considered to be affected early in the course of AD, is not of special importance in our map. This kind of analysis, very specific to the choice of the patch size and the feature chosen for the classification, can complement voxel-wise analysis in the definition of precise ROIs. Its main advantages are to be specific to a chosen feature, a chosen feature size and to reflect the actual influence in the classification, not statistical class differences.

Next we used the extracted importance map for data reduction excluding areas below a certain importance threshold. We compared the classification results obtained with different importance maps to results obtained with an inter-class variance map. Classifying normal control group (NC) vs. AD we obtained classification rates of up to 95.7 % with the importance map and up to 93.2 % with the inter-class variance map using an support vector machine (SVM) classifier and a bootstrap cross-validation. The image feature used for the classification was the mean intensity of each patch exceeding the

importance or inter-class variance threshold. That means that the higher the threshold the fewer patches are included in the calculation and consequently fewer features are passed to the classifier. While best classification results using the inter-class variance map () were obtained with about 3000 patches, using the importance map best results were achieved using only 1000 patches. This means that our importance map allows in this setup higher data reduction while achieving also better classification results than the inter-class variance map. To establish a general advantage of our method over voxel-wise statistics it is necessary to perform tests on other data sets and to perform these tests also with other features.

So far we have been reducing the data by applying a hard exclusion of patches by a threshold. In a next step we were exploring the possibility to just strongly reduce the number of patches drawn in areas of low importance. To do so we developed an algorithm to sample points according to a distribution stored in a map: spatial importance sampling (see section 8.1). We applied this algorithm in two different ways: *1)* Sampling according to predefined ROIs according to medical AD literature; and *2)* sampling according to the importance map extracted in chapter 6. Besides the two presented applications it is also possible to use the algorithm to reduce the amount of patches necessary to train an image dictionary.

In medical AD literature both parietal and temporal lobe are identified as ROIs in the detection of AD. We applied the spatial importance sampling algorithm to use this information in order to perform a data reduction. We tested whether augmenting the frequency of drawing patches in one of those ROIs affects positively the classification rate at a fixed number of drawn patches. We found that an augmented frequency of drawing patches in parietal lobe led to an augmentation of the classification rate in comparison to random drawing by approximately 2 %. An augmentation of the frequency for drawing patches in temporal lobe in contrary led to a small reduction in classification rate especially for the AD cases. The same behavior was observed over a range of different patch sized and for both 2D and 3D patches. The low influence to the classification result led to the conclusion, that the ROIs are chosen to large and can therefore not effectively select the best regions.

Consequently we tested sampling according to the importance map which contains much smaller regions of high importance. We compared sampling uniform distributed over the brain and importance sampling in a range of 100 to 1000 patches. While the classification rate decreases continuously when sampling uniformly over the brain, we observed no significant decrease of the classification rate with importance sampling. The variance of the results however increased for both importance and uniform sampling. Compared with the predefined ROIs we obtained a classification rate that is higher by 3 % at the same number of patches. This shows once again that the importance map extraction produces an accurate model for the situation of AD related changes in FDG PET scans.

In the last chapter we used a completely different approach. We interpreted each PET scan as a data tensor and assumed that each scan is composed by different components, some of them related to metabolic changes caused by AD disease. This components we want to discover by non-negative tensor factorization (NTF). As the data to be decomposed has a dimensions $91 \times 109 \times 91 \times 335$ the decomposition with standard algorithms is not possible. Therefore, we developed a novel NTF algorithm for the decomposition of large tensors (see section 3.4). This algorithm, non-negative sub-tensor set factorization ($\mathrm{N_S T_S F}$), is capable to decompose large tensors as it makes effective use of the available memory. It was developed to decompose order $N \geq 3$ tensors and was tested for order 3 and 4 tensors. In the presented algorithm it is only necessary to hold a part of the data tensors, a sub-tensor in the memory, which reduces the amount of memory needed. Furthermore, each sub-tensor has to be accessed only once during the decomposition, which leads to a drastic reduction in memory overhead, allowing fast decomposition. The design of the algorithm also allows to add new sub-tensors to the data tensor without the need to restart decomposition. However, it was necessary to perform an approximation in the optimization problem. We were not (yet) able to determine an upper bound for the error committed by this approximation. Tests of our algorithm however showed good convergence properties and a performance competitive with state of the art NTF algorithms with respect to both decomposition time and error (see subsection 3.4.6). The comparison tests were performed using face images, which allows an easy interpretation of the decomposition results. We expect our algorithm to perform even better in comparison to standard algorithms with tensors bigger than the tested ones, as the computational cost of our algorithm is growing slower.

In chapter 9 we present the application of this algorithm as a feature extraction method for the early detection of AD. We performed a decomposition of the image data and used this decomposition to represent the data in lower dimensional space. This reduces the dimensionality of the classification problem from the number of voxels, approximately 900 000, to the rank R of the decomposition (we obtained good results with $R = 32$). Feeding the encoding factors of the images in this new basis to a SVM for classification we were capable to separate NC from AD with a classification rate of 92.6 %, which exceeds most state of the art classification rates for early AD. Furthermore, allowed this approach a classification of MCI vs. $\mathrm{MCI_{AD}}$ with a classification rate of 84.7 %, a distinction medical doctors were not capable of while using not only the FDG PET scans but also neurophysiological testing, MRI and other biomarkers. This results show that the detection of AD in a FDG PET scan, which was acquired approximately 2 years *before* a clinical diagnosis was possible, is feasible.

One argument that suggests that image should not be vectorized before decomposition is that a vectorization could induce an information loss, as the pixels in an image surely have a trivial dependence on each other. We have performed the decomposition

of our data in its original shape, i.e. as a 4^{th} order tensor, but also as a 3^{rd} order tensor, obtained by vectorization of the transversal slices in each image. The error of the obtained decomposition was lower for the 3^{rd} order tensor, which can have two reasons: *1)* The decomposition algorithm performs better on 3^{rd} order tensors; or *2)* the vectorization induces only a minor loss of information, as the transversal slices are full rank. However, this experiment can not answer this question, additional theoretic investigation would be necessary.

To get a more detailed insight in this classification we trained SVMs with samples of NC and AD cases and recorded the classification outcome of each scan that has not been used in the training individually. Repeating this procedure 10 000 times we obtained a statistic of the NC and AD labels attributed to the scans of all classes, including MCI and MCI_{AD}. This experiment showed that our approach has a very low dependence on the data used for the training of the classifier, shown by the high percentage of NC and AD cases correctly classified in more than 90 % of the trainings. The response of the trained classifiers to MCI_{AD} cases proved the promising results of the binary classification experiments involving the MCI_{AD} class. A large number of scans was classified AD in more than 80 % of the trainings. Classifying the MCI cases with the trained SVMs we observed a separation in scans classified mostly as NC and others mostly classified as AD. The number of patients classified as AD is much smaller. This result was expected as the MCI class contains patients with a very mildly impaired memory which can be part of normal aging and we have already separated MCI_{AD} cases that have actually been diagnosed AD in the follow-up period. Here it would be very interesting whether the MCI patients our classifiers have classified AD have been diagnosed AD after the 2 years of follow-up examinations.

In this experiment we also discovered a small number of scans that is false classified in more than 95 % of the cases. As there is no absolute certainty for the correctness of all labels a re-examination of those cases by medical doctors might be clarifying. This classification task shows the urgent need of information about the influence of label noise and assessment of the robustness of classification setups to label noise. Unfortunately the vaste majority of publications restricts its analysis to noise on the descriptive variables. Noisy target variables or labels are rarely investigated.

As statistical parametric mapping (SPM) analysis, inter-class variance and our machine learning approach to generate a map of AD detection relevant areas show concurrently that metabolic changes are localized and not wide spread over the whole brain it can be assumed that local features should be preferred in the analysis. The representations learned by NTF are purely additive and therefore parts based. With an optional sparseness constraint we are able to enforce a higher level of localization. This localization however depends only on the data and the learning algorithm. With the capability of the presented NTF algorithm we are also able to create a classification scheme based on patches. The presented $\text{N}_{\text{S}}\text{T}_{\text{S}}\text{F}$ algorithm allows the necessary

decomposition of a tensor composed by a large number of three dimensional patches. Test decompositions of 3.3 million $5 \times 5 \times 5$ patches were performed but a feature extraction based on the obtained representation basis is not yet implemented.

Our experiments have shown that a computerized early detection of AD using FDG PET scans is possible at a stage where medical doctors are not yet able to distinguish between MCI and AD. In order to incorporate our investigational efforts into a computer-aided diagnosis (CAD) system it would be necessary to either visually highlight the zones that have most influenced the classification of an individual scan, produce a AD score instead of a hard classification or use the classification result like a second opinion. Given the good classification results on clinical data such a CAD system should be able to enhance early detection of AD in the clinical situation. A next step in the development of this application should also include other dementias that are, especially in the MCI phase, hard to distinguish from AD.

The $N_S T_S F$ algorithm also leaves open questions and possibilities for enhancement. An upper bound for the error committed by the involved approximation needs to be determined. A prove of convergence or at least a prove that the update is not increasing the cost function is needed. This is complicated by the separation of the algorithm in two phases. Furthermore, the algorithm could be further enhanced by replacing the multiplicative update of the Lee and Seung scheme by an other, more effective method. For the update in the coding step this is straight forward, but problems are possible in the coding step. For a better understanding of the comportement of the $N_S T_S F$ algorithm further experiments analyzing the decomposition of sparse tensors and tensors of higher order is necessary.

A field of application for the $N_S T_S F$ algorithm that should be explored is the recognition. In this field decomposition methods that are related to NTF, like principal component analysis (PCA), independent component analysis (ICA) and non-negative matrix factorization (NMF), are widely used and achieve good results. NTF offers the possibility to leave the structure of the images unchanged, which includes the spatial structure of the pixels as well as the color channels. The problem of low decomposition speed and size restrictions of NTF are alleviated by our $N_S T_S F$ algorithm.

Glossary

Anterior Commissure-Posterior Commissure (AC-PC) line

This line passes through the superior edge of the anterior commissure and the inferior edge of the posterior commissure. It follows a path essentially parallel to the hypothalamic sulcus, dividing the thalamic from the subthalamic region. This line defines the horizontal plane (Talairach and Tournoux [1988]).

Clinical Dementia Rating (CDR)

Clinical dementia rating is based on the examination of the six domains: memory, orientation, judgement and problem solving, community affairs, home and hobbies, and personal care, where memory is the main domain. CDR assigns 0 for no dementia, 0.5 for very mild, 1 for mild, 2 or moderate and 3 for severe dementia.

Fluorodeoxyglucose (FDG)

^{18}F fluorodeoxyglucose is the most common tracer used in PET imaging. It is a glucose analog with one hydroxyl group replaced by ^{18}F, which is a positron emitter with half-live time 109.8 minutes. As the ^{18}F perhibits glycolysis FDG is metabolicaly trapped in the cell until decay. Therefore FDG concentration is a good representation for the distribution of glucose uptake. Because of its short half-live time and the need for a zyclotron for synthesis handling of FDG is complicated and costly. ${}^{18}_{9}\mathrm{F} \rightarrow {}^{18}_{8}\mathrm{O} + e^{+} + \nu_e$, $E_{e^+} = 0.7\mathrm{keV}$.

Hoffman 3-D Brain Phantom

The Hoffman 3-D Brain Phantom provides the anatomical accurate three dimensional simulation of the radioisotope distribution in the normal human brain. The phantom is compromised of sturdy plastic and a single fillable chamber that eliminates the necessity of preparing different concentrations of radioisotope.

Mild Cognitive Impairement (MCI)

Cognitive impairement that is considered to be higher than to be expected at the persons age but not as signifficant to infer dementia. Patients diagnosed with mild cognitive impairement have been found to be more prone to devellop AD than the average aged person. This raises high interes in mild cognitive impairement for the devellopment of effective early diagnosis methods for AD.

Mini-Mental State Exam (MMSE)

The mini-mental state exam consists of 30 questions covering orientation, registration, attention and calculation, recall, language and praxis. The answers are used to calculate a dementia score. 24-30 points correspond no cognitive impairment, 18-23 mild cognitive impairment and 0-17 severe cognitive impairment. Because of its easy in acquisition and evaluation this is probably the most common of all dementia tests.

Montreal Neurological Institute (MNI)

The Montreal Neurological Institute (MNI) space is based and image that represents the average over a large series of CT, MRI, PET or SPECT scans of different patients.

Pittsburgh compound B (^{11}C-PIB)

Radioligand that binds to amyloid β (Aβ) plaques and thus allows imaging of brain changes that are identified as a primary cause of AD by several studies. The half-live time of ^{11}C of 20.39 minutes allows now flaws in supply and application chain. ${}^{11}_{6}\mathrm{C} \rightarrow {}^{11}_{5}\mathrm{B} + e^{+} + \nu_e$, $E_{e^+} = 1.0\mathrm{MeV}$.

Vertico-frontal line (VCA line)

The term vertico-frontal line refers to the vertical axis of the human brain in Talairach space. It passes through the caudal-most point of the anterior commissure in the midline as viewed by ventriculography. It lies in the mid-sagittal plane and is perpendicular to the bicommissural line, which in Talairach space passes through the most dorsal point of the anterior commissure and most ventral point of the posterior commissure in the mid-sagittal plane (Talairach and Tournoux [1988]).

A. Matrix and vector operations

Kronecker product

The Kronecker product of two matrices $\boldsymbol{A} \in \mathbb{R}^{I\times J}$ and $\boldsymbol{B} \in \mathbb{R}^{K\times L}$ is denoted by $\boldsymbol{A} \otimes \boldsymbol{B}$ and the result is defined by

$$\boldsymbol{A} \otimes \boldsymbol{B} = \begin{pmatrix} a_{11}\boldsymbol{B} & a_{12}\boldsymbol{B} & \dots & a_{1J}\boldsymbol{B} \\ a_{21}\boldsymbol{B} & a_{22}\boldsymbol{B} & \dots & a_{2J}\boldsymbol{B} \\ \vdots & \vdots & \ddots & \vdots \\ a_{I1}\boldsymbol{B} & a_{I2}\boldsymbol{B} & \dots & a_{IJ}\boldsymbol{B} \end{pmatrix} \in \mathbb{R}^{(IK)\times(JL)} \tag{A.1}$$

Outer product

The outer vector product of two vectors $\boldsymbol{a} \in \mathbb{R}^n$ and $\boldsymbol{b} \in \mathbb{R}^m$ is denoted as

$$\boldsymbol{a} \circ \boldsymbol{b} = \begin{pmatrix} a_1 b_1 & a_1 b_2 & \dots & a_1 b_M \\ a_2 b_1 & a_2 b_2 & \dots & a_2 b_M \\ \vdots & \vdots & \ddots & \vdots \\ a_N b_1 & a_N b_2 & \dots & a_N b_M \end{pmatrix} \in \mathbb{R}^{N\times M} \tag{A.2}$$

with $1 \leq n \leq N$ and $1 \leq m \leq M$.

Hadamard product

The Hadamard, or element wise product, of two matrices of same dimensions $\boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{I\times J}$ is defined as

$$\boldsymbol{X} = \boldsymbol{A} \bullet \boldsymbol{B} = \begin{pmatrix} a_{11}b_{11} & a_{12}b_{12} & \dots & a_{1J}b_{1J} \\ a_{21}b_{21} & a_{22}b_{22} & \dots & a_{2J}b_{2J} \\ \vdots & \vdots & \ddots & \vdots \\ a_{I1}b_{I1} & a_{I2}b_{I2} & \dots & a_{IJ}b_{IJ} \end{pmatrix} \in \mathbb{R}^{I\times J} \tag{A.3}$$

The corresponding element wise devision be denoted as

$$\boldsymbol{X} = \frac{\boldsymbol{A}}{\boldsymbol{B}} \in \mathbb{R}^{I\times J} \tag{A.4}$$

B. Supplementary material

- Figure B.1 shows the basis images obtained by a $N_S T_S F$ decomposition with $R = 32$ in chapter 9. There is no order in the basis images. Due to the non-negativity constraint the decomposition is parts based.
- Table B.1 contains the numerical results of the plot in Figure 8.5 on page 117.
- Table B.2 contains the numerical results of the plot in Figure 8.6 on page 118.

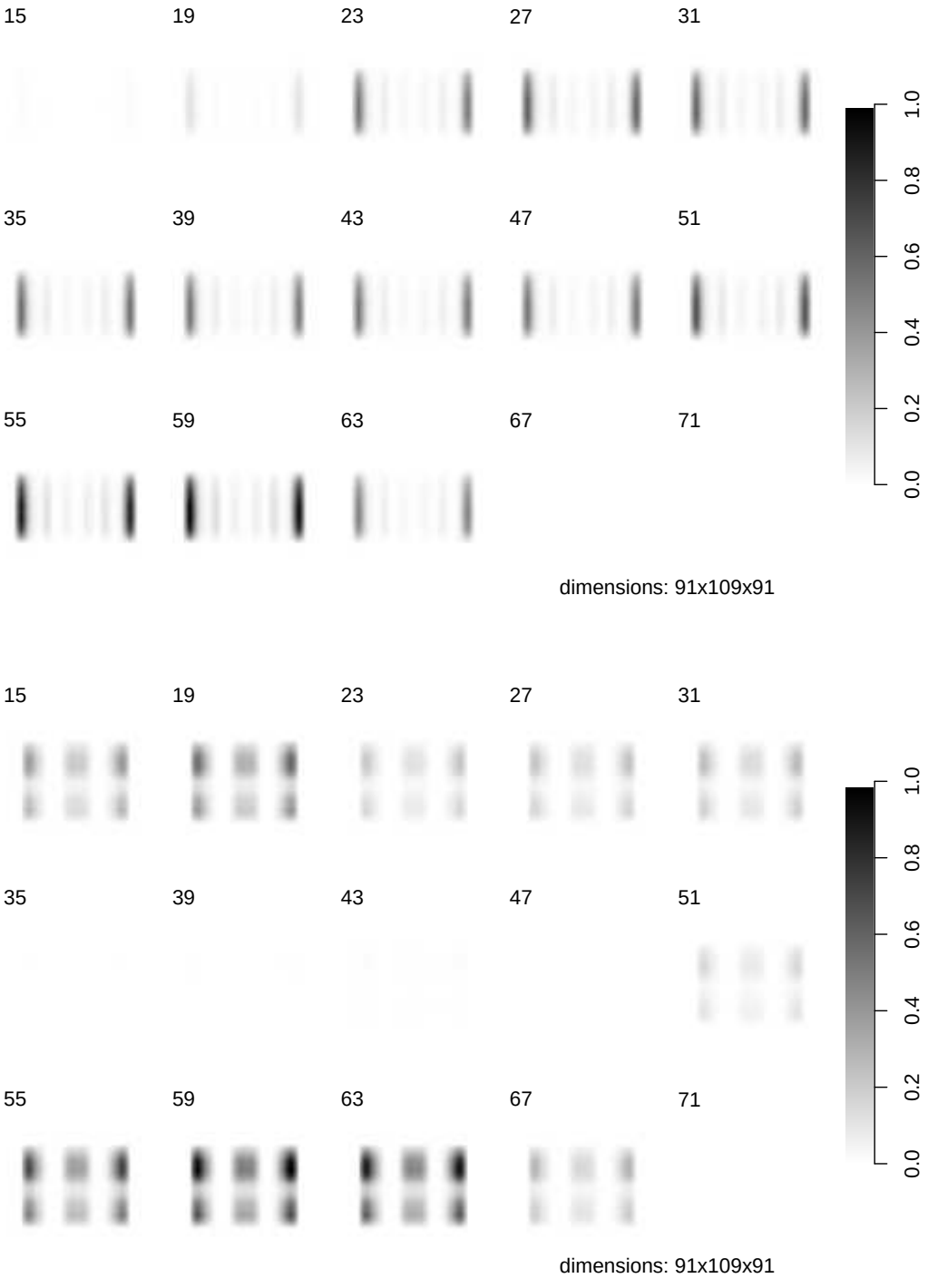


Figure B.1.: Basis images obtained by $N_S T_S F$ decomposition.

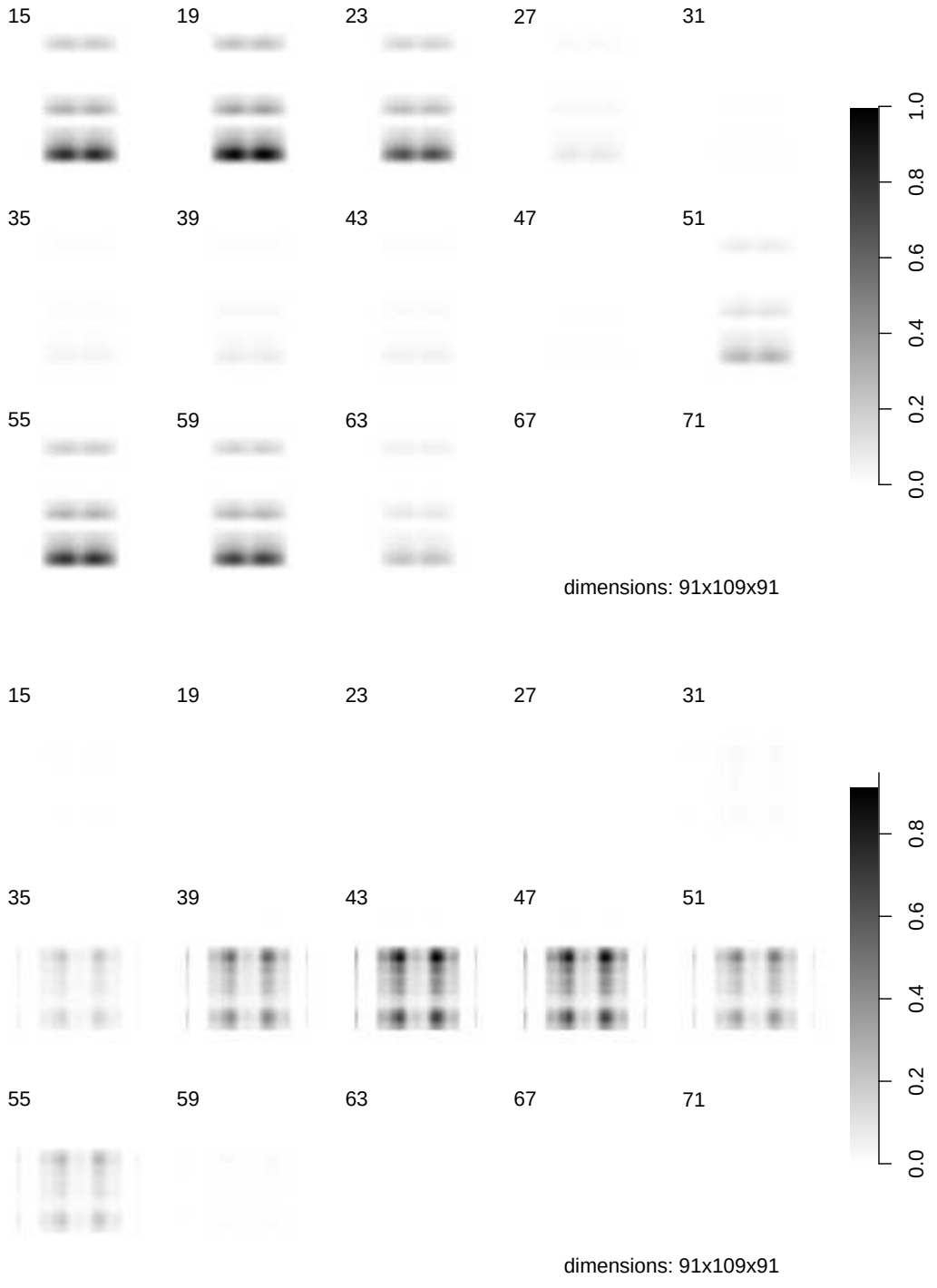


Figure B.1.: Basis images obtained by $N_S T_S F$ decomposition.

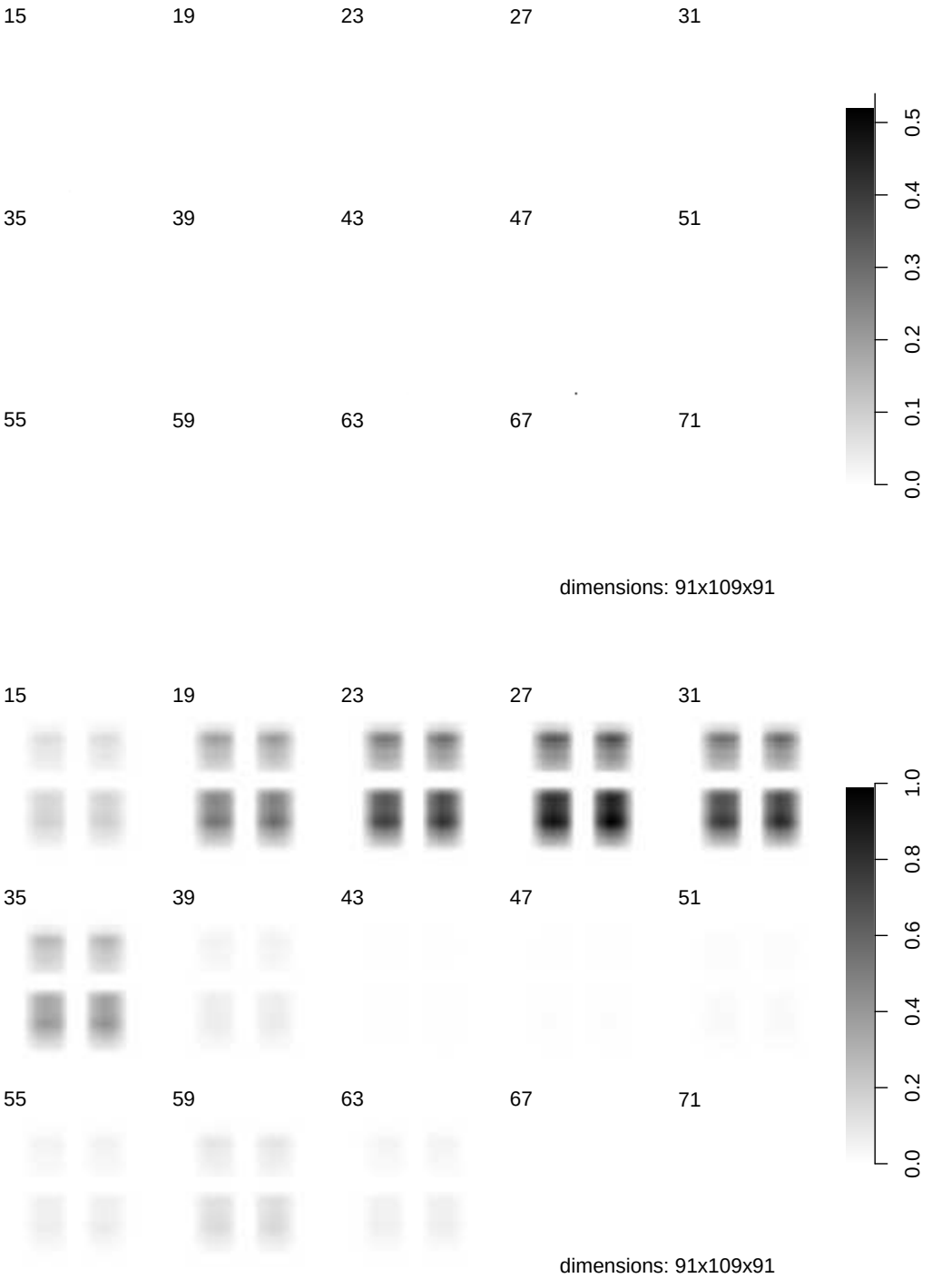


Figure B.1.: Basis images obtained by $N_S T_S F$ decomposition.

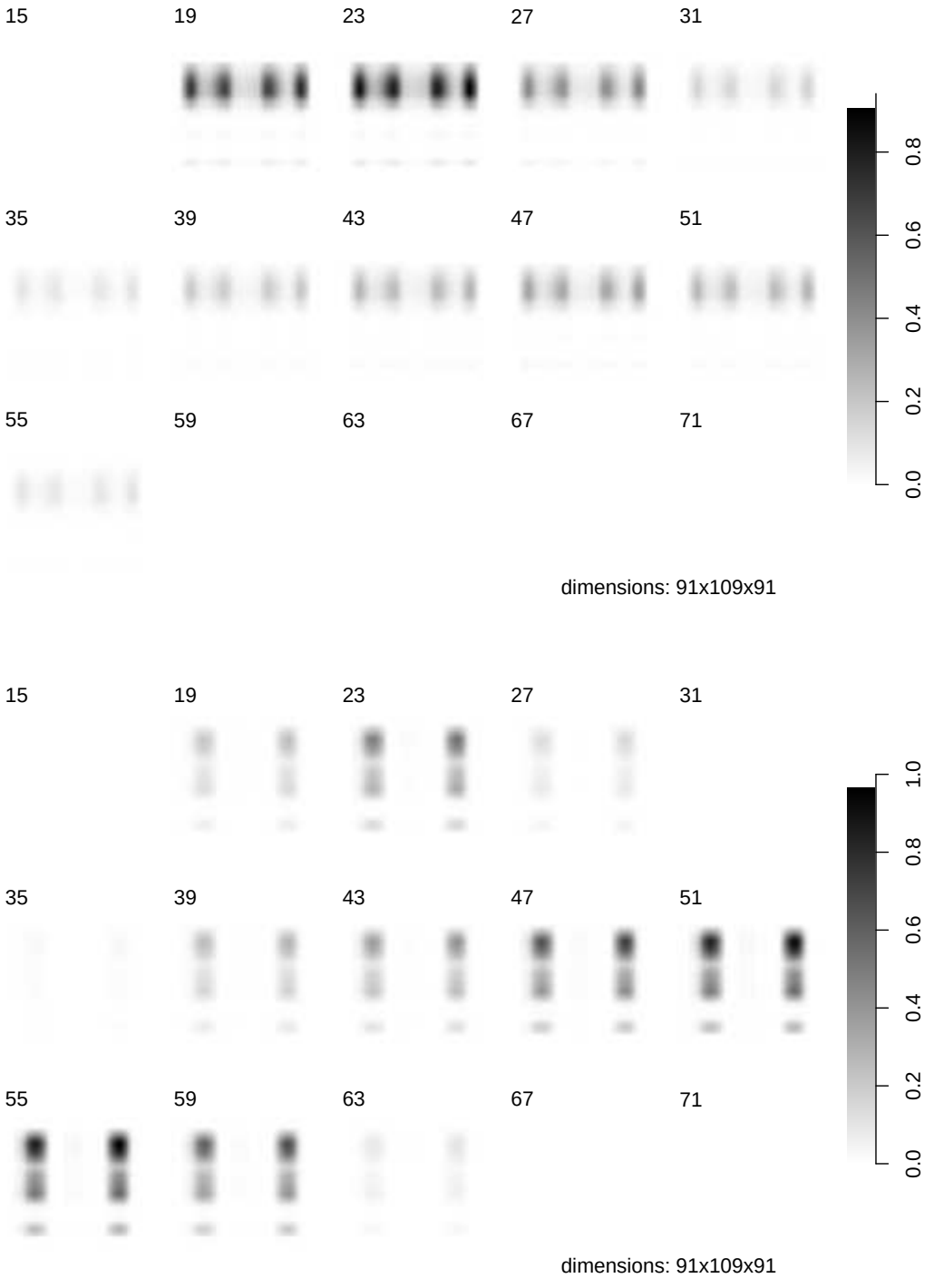


Figure B.1.: Basis images obtained by $N_S T_S F$ decomposition.

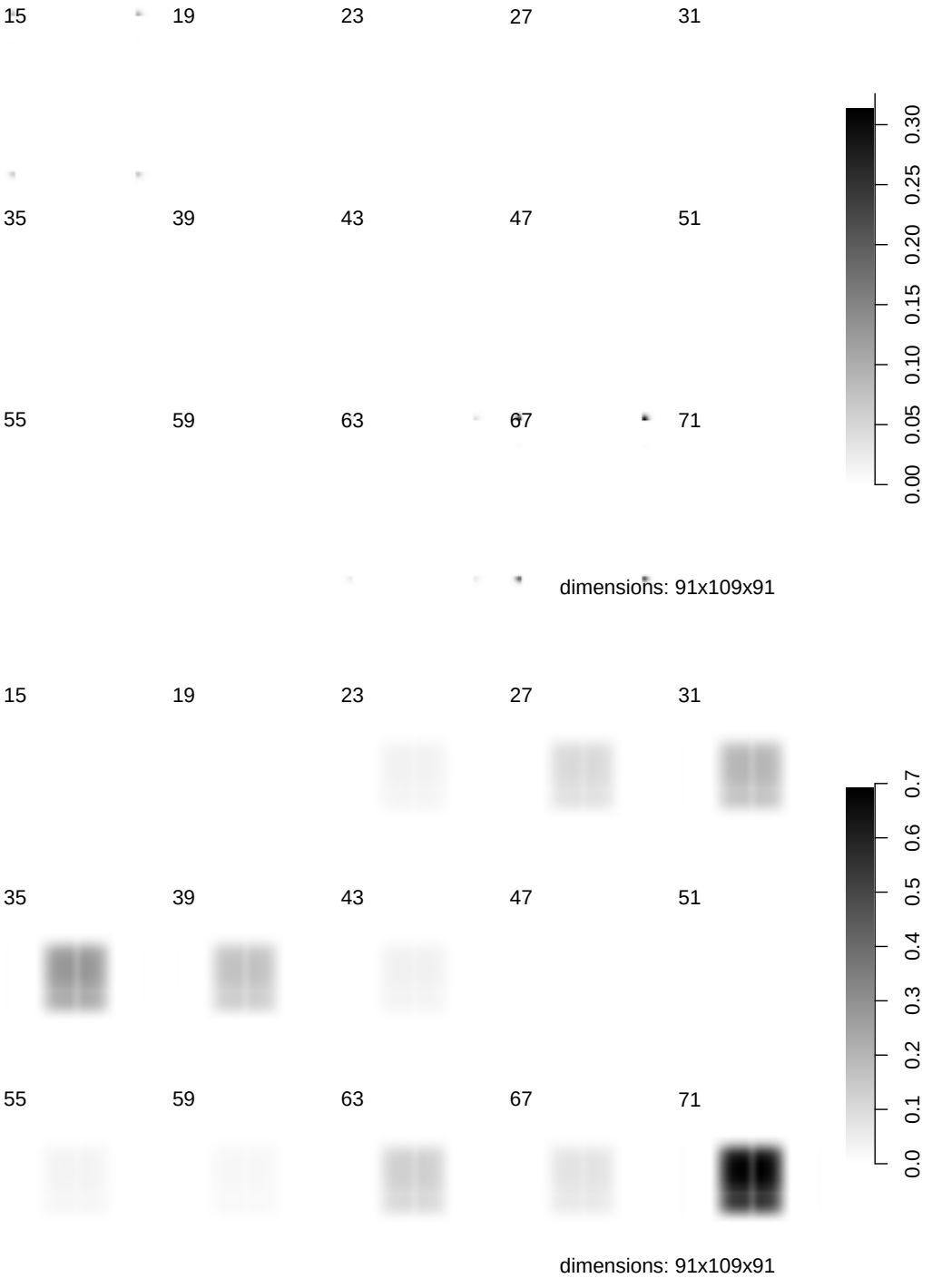


Figure B.1.: Basis images obtained by $N_S T_S F$ decomposition.

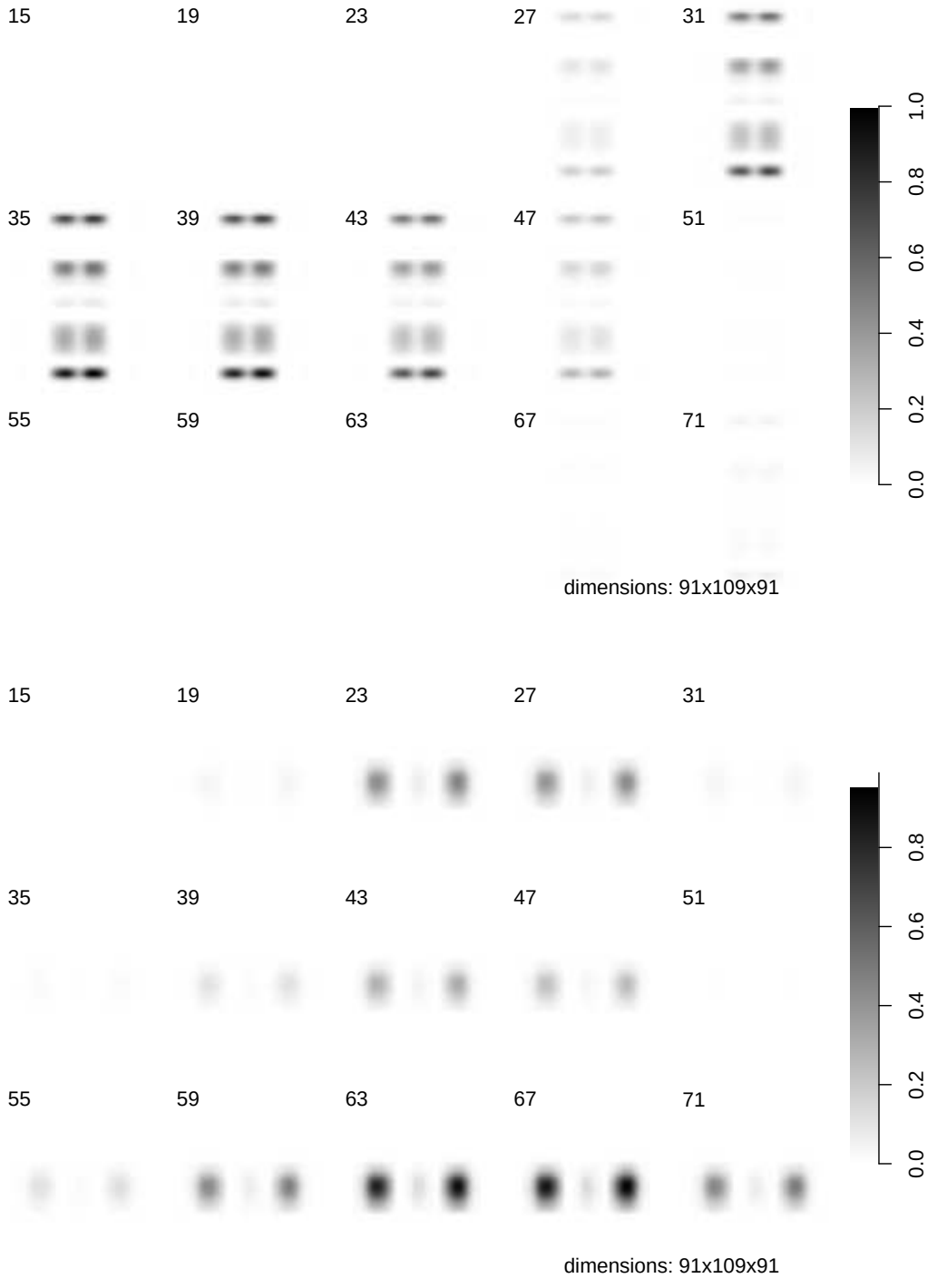


Figure B.1.: Basis images obtained by $N_S T_S F$ decomposition.

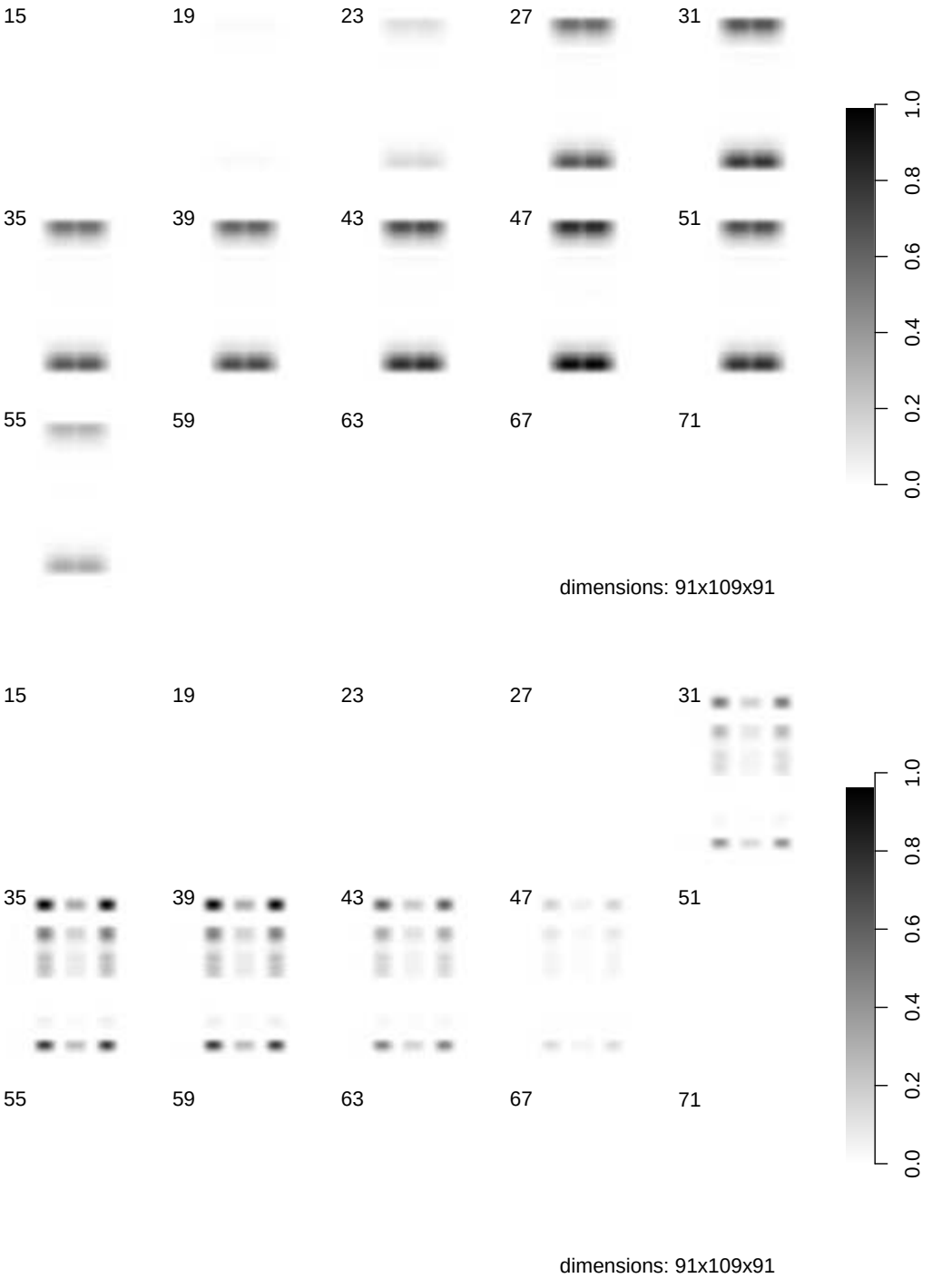


Figure B.1.: Basis images obtained by $N_S T_S F$ decomposition.

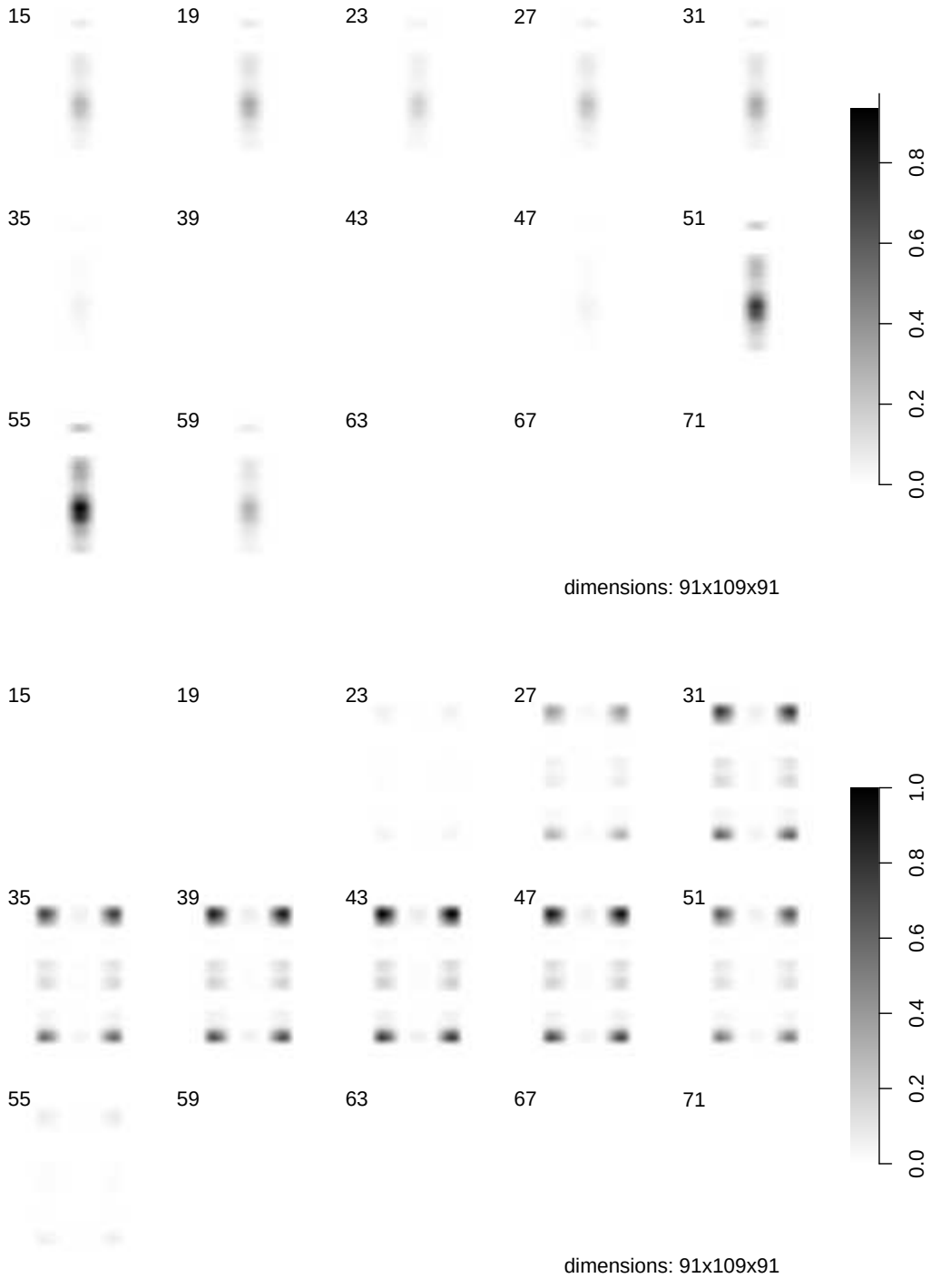


Figure B.1.: Basis images obtained by $N_S T_S F$ decomposition.

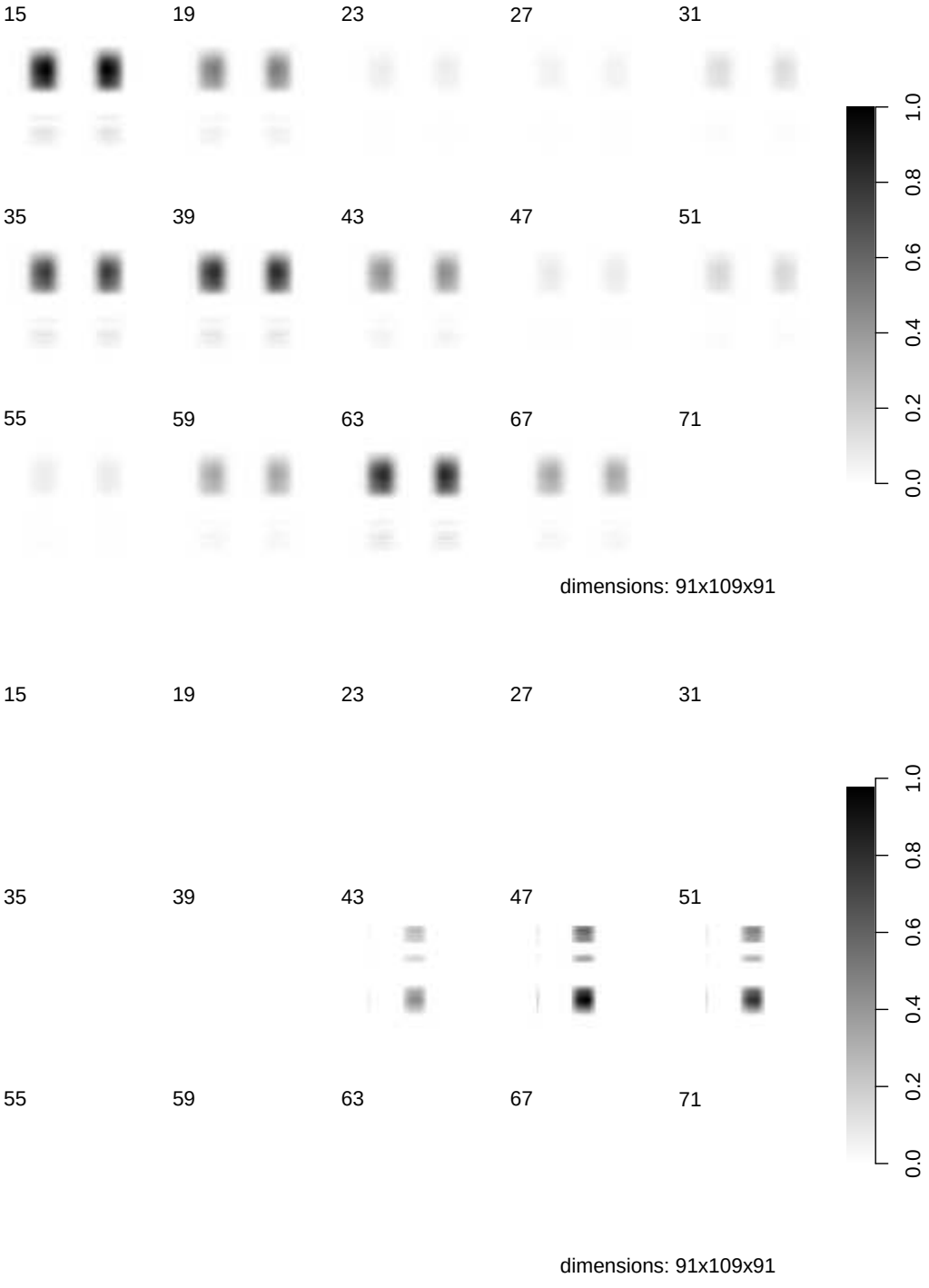


Figure B.1.: Basis images obtained by $N_S T_S F$ decomposition.

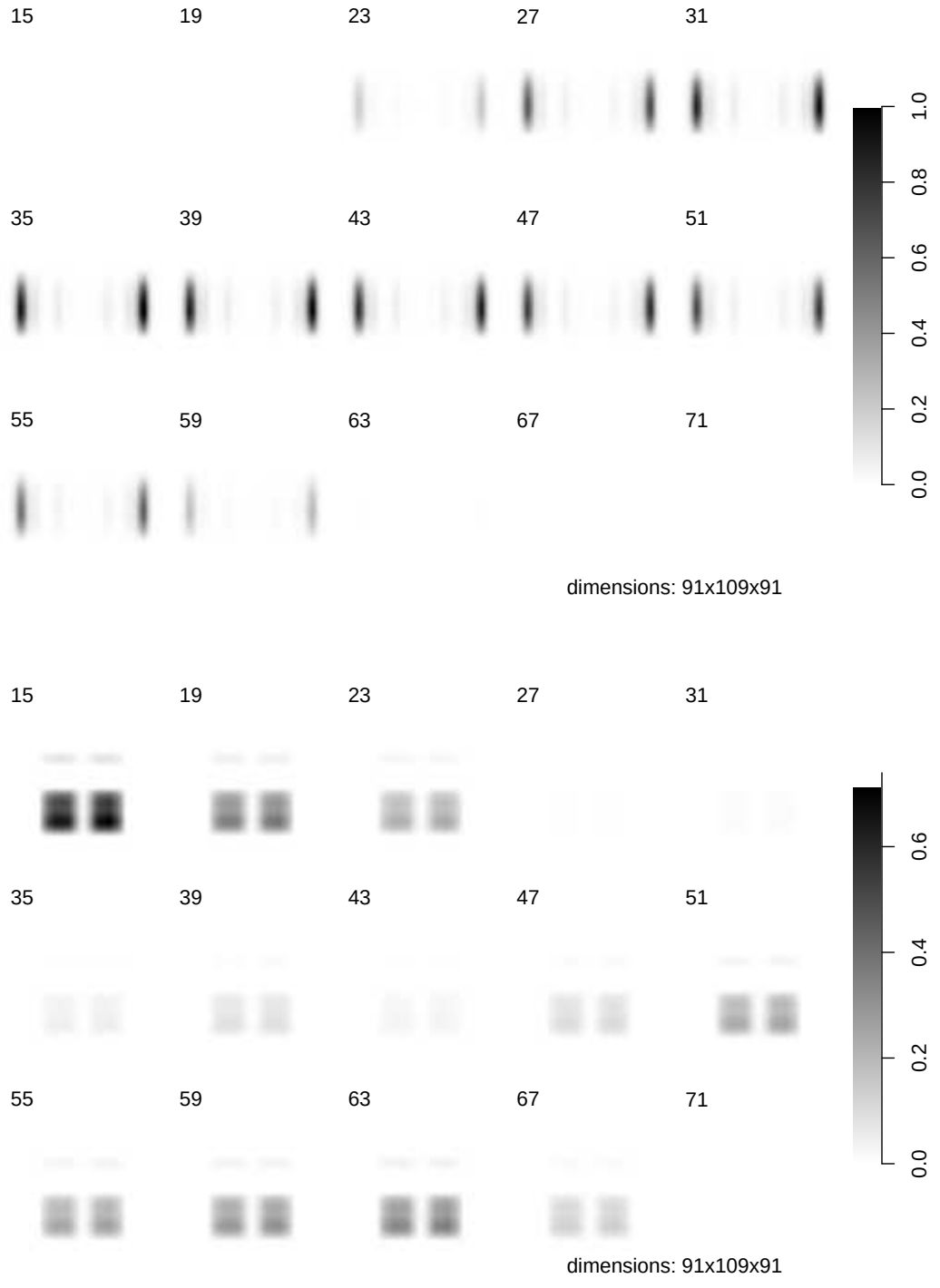


Figure B.1.: Basis images obtained by $N_S T_S F$ decomposition.

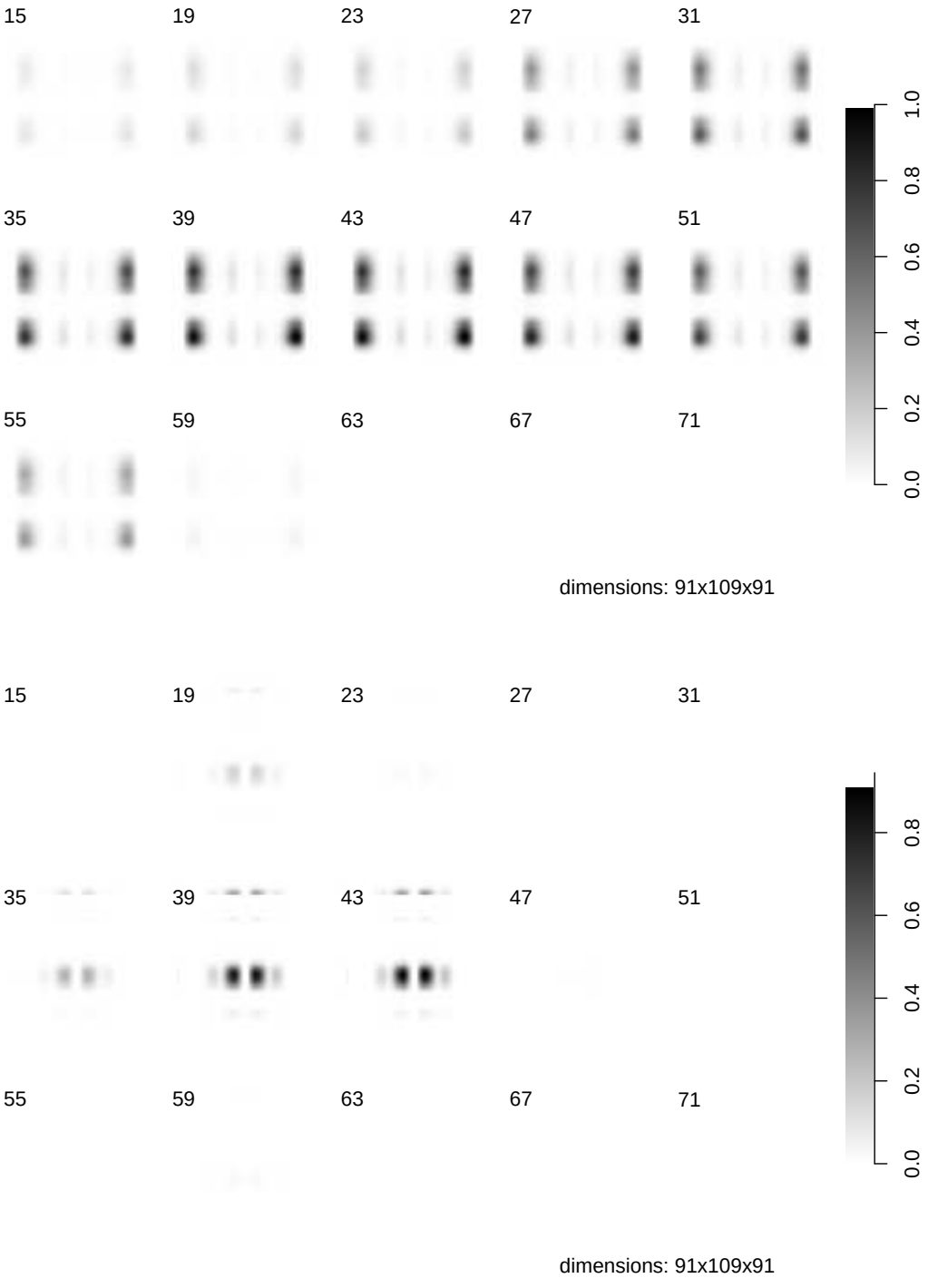


Figure B.1.: Basis images obtained by $N_S T_S F$ decomposition.

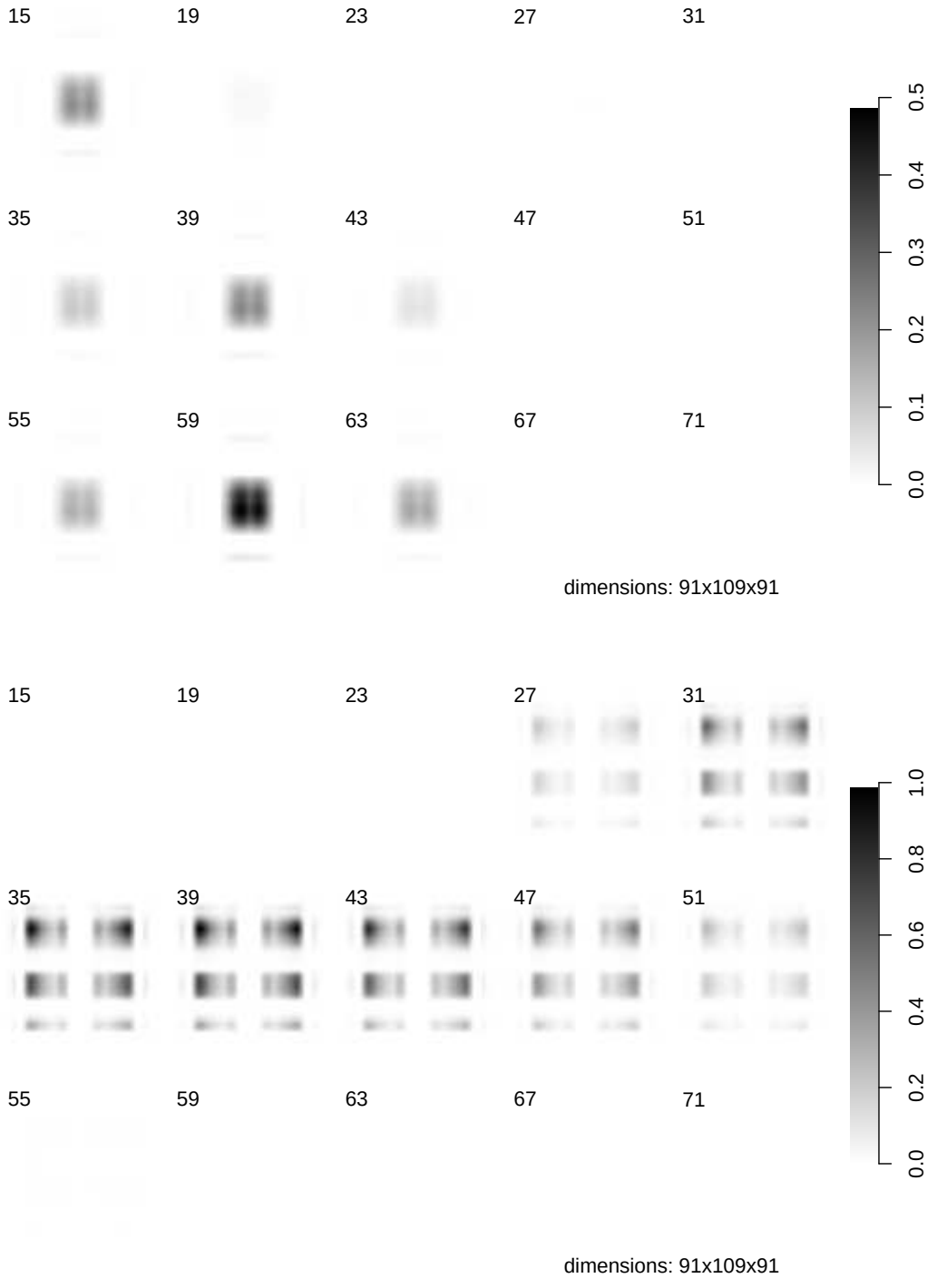


Figure B.1.: Basis images obtained by $N_S T_S F$ decomposition.

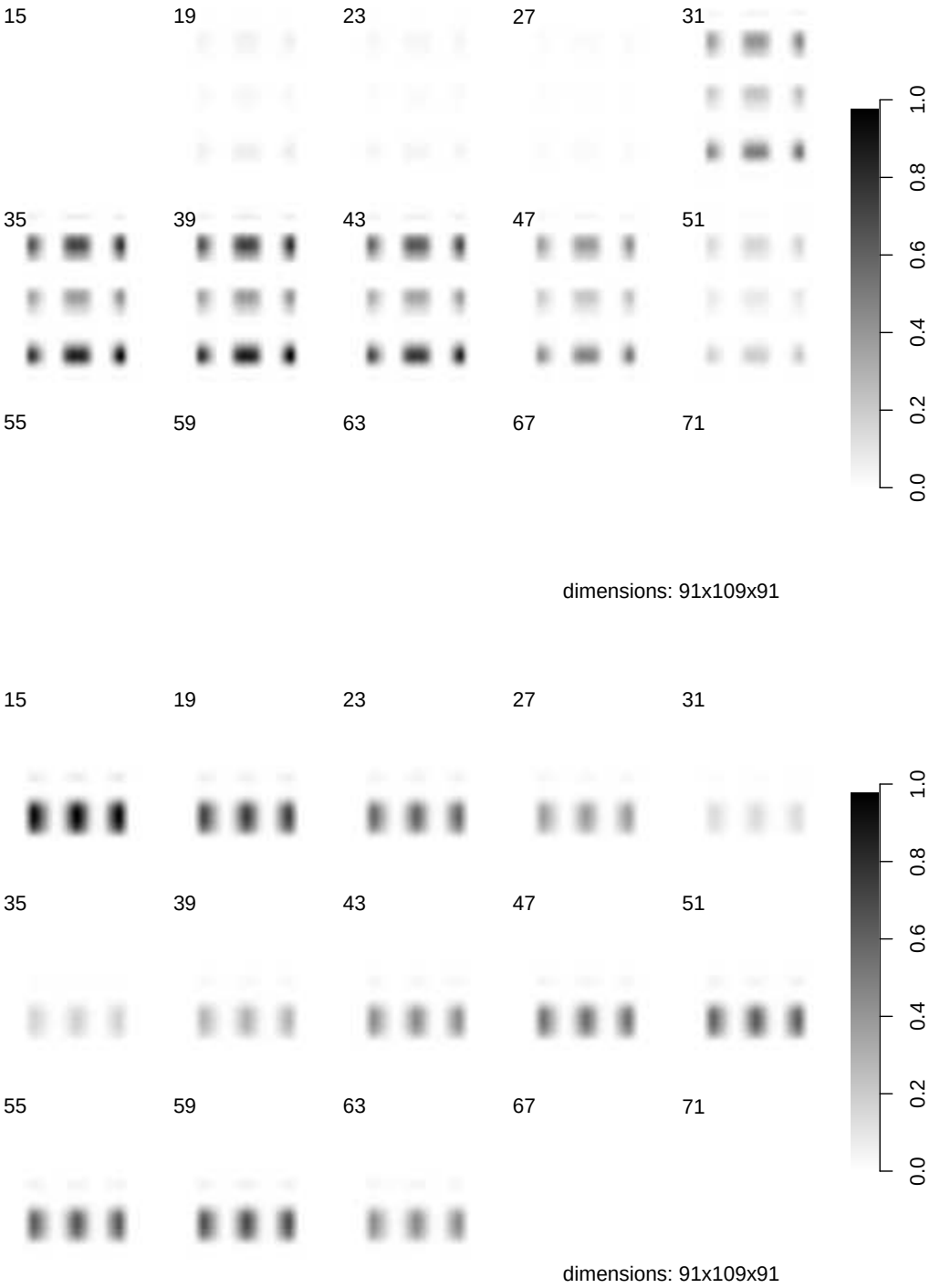


Figure B.1.: Basis images obtained by $N_S T_S F$ decomposition.

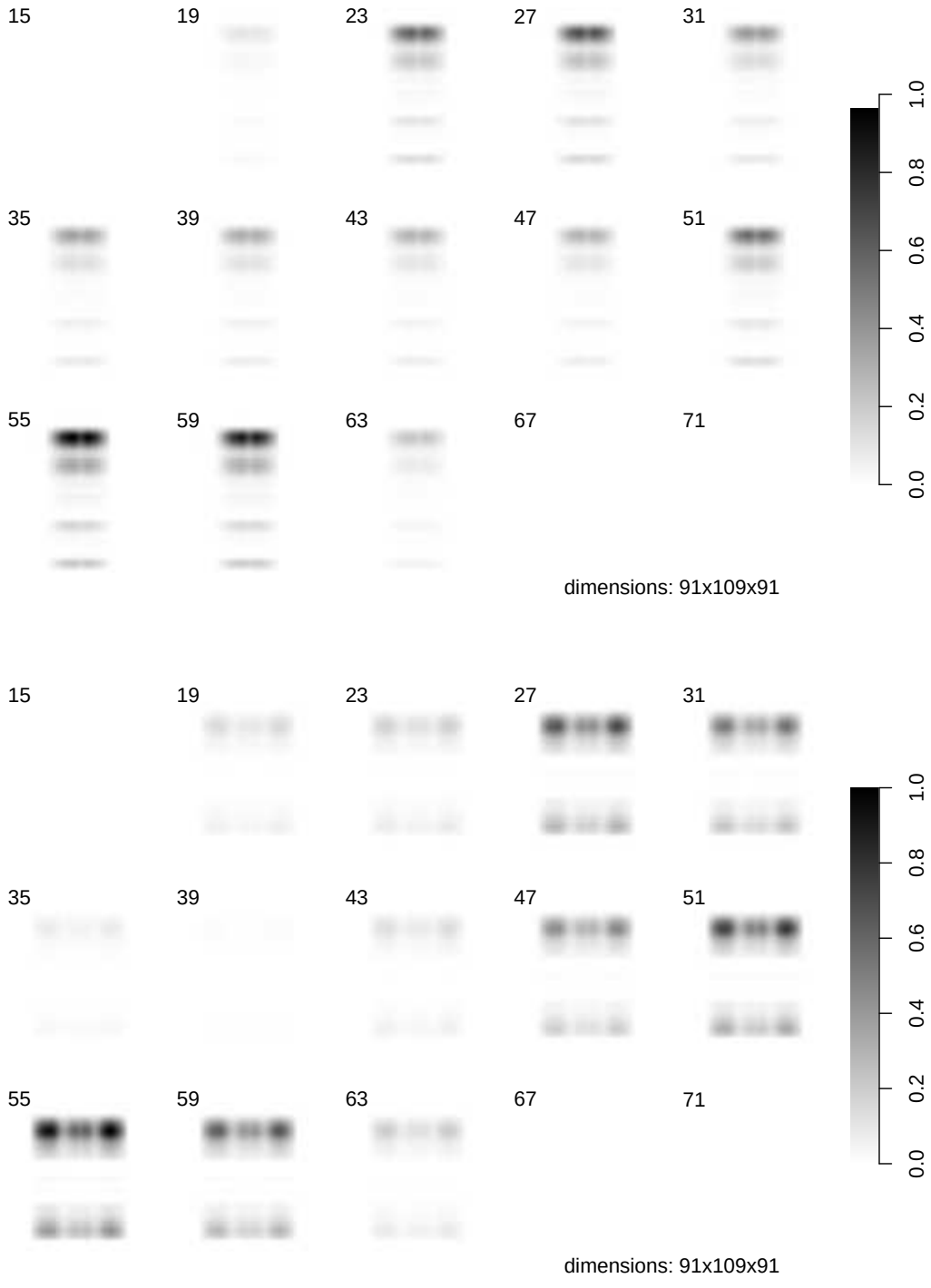


Figure B.1.: Basis images obtained by $N_S T_S F$ decomposition.

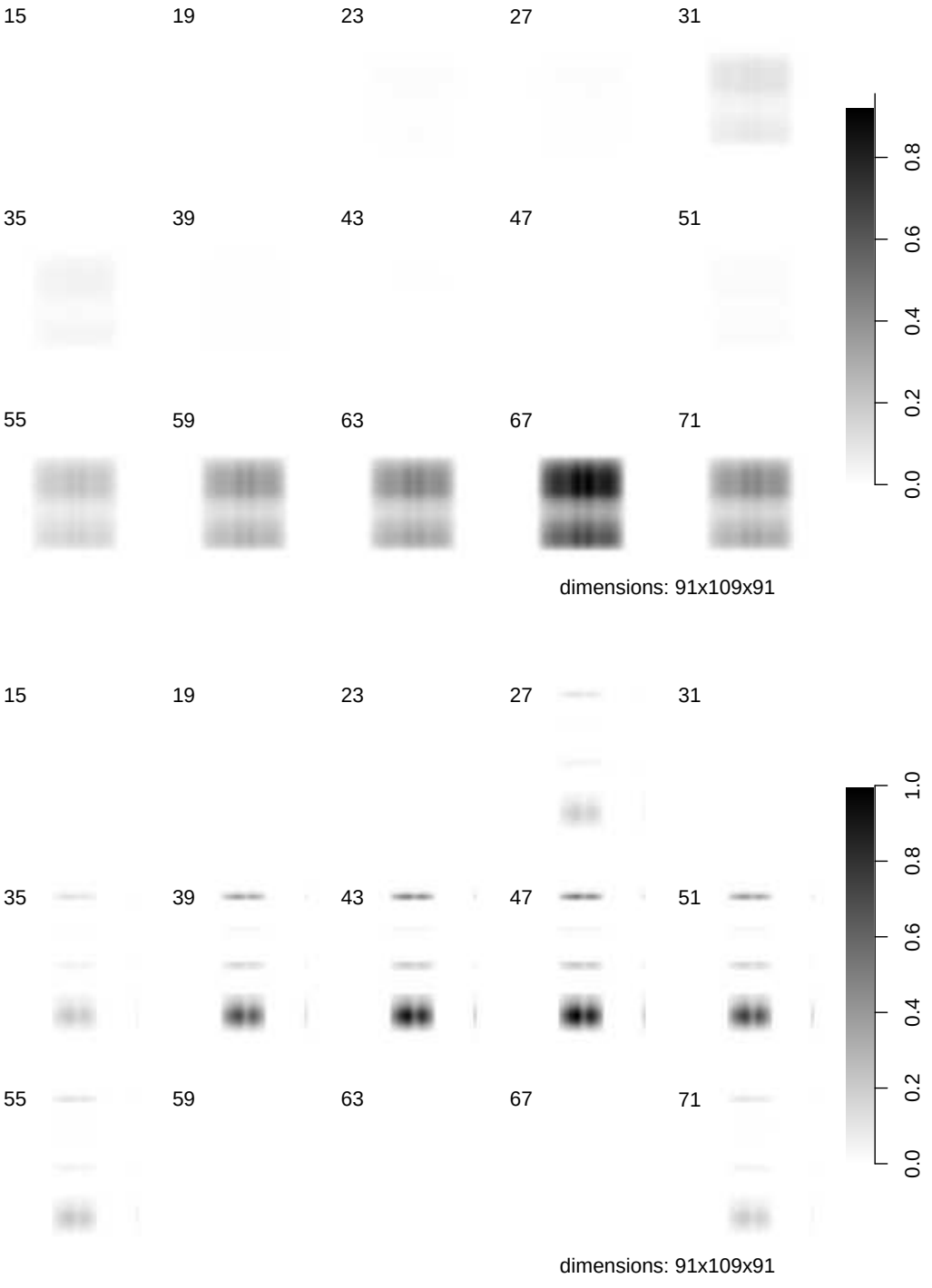


Figure B.1.: Basis images obtained by $N_S T_S F$ decomposition.

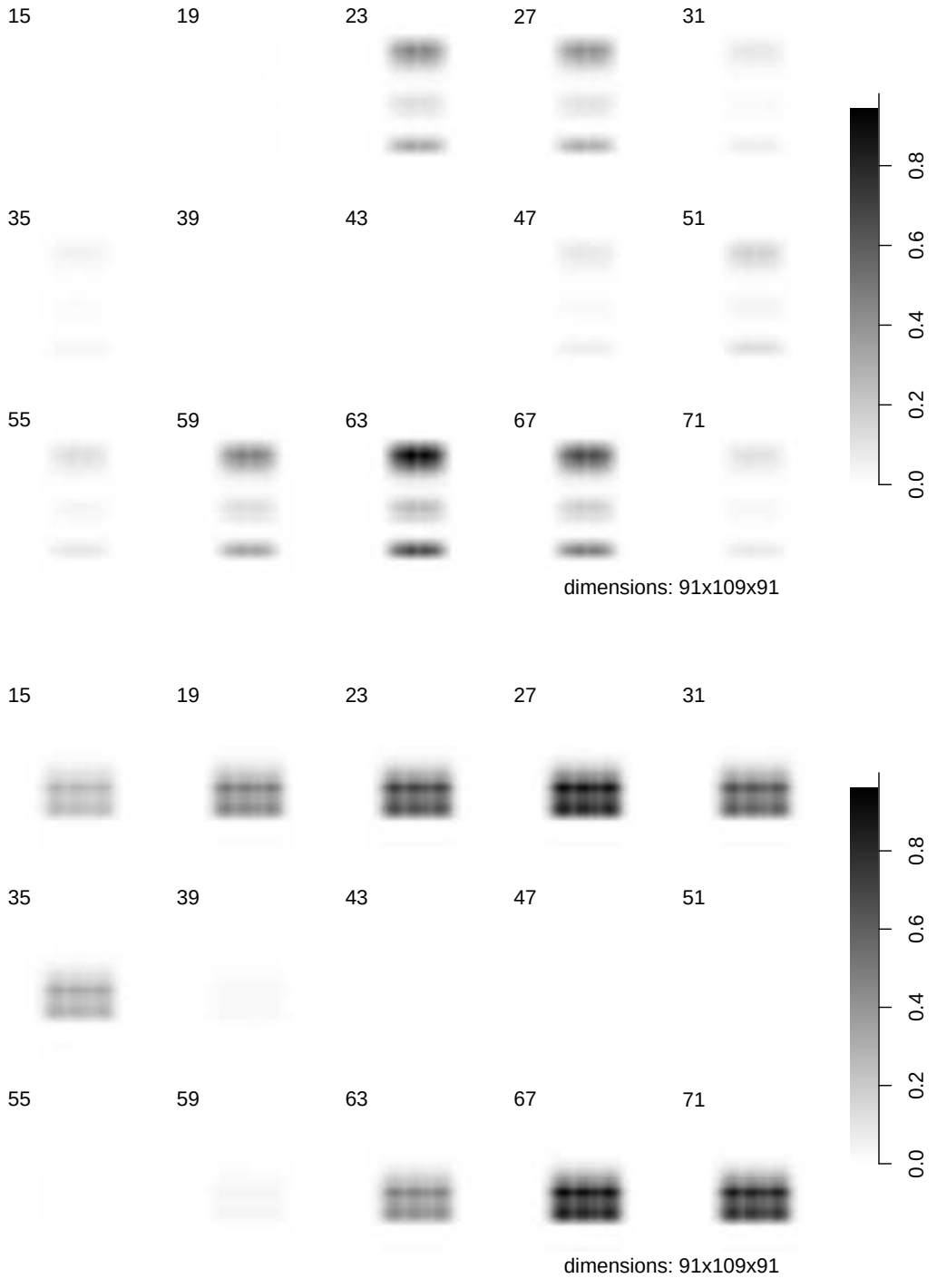


Figure B.1.: Basis images obtained by $N_S T_S F$ decomposition.

$\frac{P(\text{VOI}\|s)}{P(\text{not VOI}\|s)}$	$3\times3\times3$	$4\times4\times4$	$5\times5\times5$	$6\times6\times6$	$7\times7\times7$
0	0.76 ± 0.04	0.76 ± 0.04	0.76 ± 0.04	0.76 ± 0.04	0.76 ± 0.04
0.05	0.80 ± 0.03	0.80 ± 0.03	0.80 ± 0.03	0.80 ± 0.03	0.79 ± 0.02
0.1	0.80 ± 0.02	0.80 ± 0.03	0.80 ± 0.03	0.80 ± 0.03	0.80 ± 0.03
0.15	0.81 ± 0.02	0.81 ± 0.03	0.81 ± 0.02	0.81 ± 0.02	0.80 ± 0.03
0.2	0.81 ± 0.02	0.81 ± 0.02	0.81 ± 0.02	0.81 ± 0.02	0.81 ± 0.02
0.25	0.81 ± 0.02	0.82 ± 0.02	0.82 ± 0.02	0.81 ± 0.02	0.81 ± 0.02
0.5	0.82 ± 0.02	0.82 ± 0.02	0.82 ± 0.02	0.82 ± 0.02	0.81 ± 0.02
0.75	0.82 ± 0.02	0.83 ± 0.02	0.82 ± 0.02	0.82 ± 0.02	0.82 ± 0.02
1	0.83 ± 0.02	0.83 ± 0.02	0.82 ± 0.02	0.82 ± 0.02	0.82 ± 0.02

(a) NC classification rates with focusing on parietal lobe.

$\frac{P(\text{VOI}\|s)}{P(\text{not VOI}\|s)}$	$3\times3\times3$	$4\times4\times4$	$5\times5\times5$	$6\times6\times6$	$7\times7\times7$
0	0.76 ± 0.04	0.77 ± 0.04	0.77 ± 0.04	0.77 ± 0.04	0.77 ± 0.03
0.05	0.77 ± 0.03	0.77 ± 0.04	0.77 ± 0.04	0.77 ± 0.03	0.77 ± 0.03
0.1	0.77 ± 0.04	0.77 ± 0.04	0.77 ± 0.04	0.78 ± 0.03	0.77 ± 0.04
0.15	0.77 ± 0.03	0.78 ± 0.03	0.78 ± 0.03	0.78 ± 0.03	0.78 ± 0.03
0.2	0.77 ± 0.03	0.77 ± 0.03	0.78 ± 0.03	0.78 ± 0.03	0.78 ± 0.03
0.25	0.77 ± 0.03	0.77 ± 0.03	0.78 ± 0.04	0.78 ± 0.03	0.78 ± 0.03
0.5	0.77 ± 0.04	0.77 ± 0.04	0.78 ± 0.04	0.78 ± 0.04	0.78 ± 0.04
0.75	0.77 ± 0.03	0.77 ± 0.03	0.77 ± 0.04	0.78 ± 0.03	0.78 ± 0.03
1	0.77 ± 0.03	0.77 ± 0.03	0.78 ± 0.03	0.79 ± 0.04	0.78 ± 0.03

(b) AD classification rates with focusing on parietal lobe.

$\frac{P(\text{VOI}\|s)}{P(\text{not VOI}\|s)}$	$3\times3\times3$	$4\times4\times4$	$5\times5\times5$	$6\times6\times6$	$7\times7\times7$
0	0.76 ± 0.04	0.76 ± 0.04	0.76 ± 0.04	0.76 ± 0.04	0.76 ± 0.04
0.05	0.77 ± 0.04	0.77 ± 0.03	0.78 ± 0.03	0.77 ± 0.03	0.76 ± 0.03
0.1	0.78 ± 0.04	0.78 ± 0.03	0.78 ± 0.03	0.77 ± 0.03	0.77 ± 0.03
0.15	0.78 ± 0.03	0.78 ± 0.03	0.78 ± 0.03	0.78 ± 0.03	0.78 ± 0.03
0.2	0.79 ± 0.03	0.79 ± 0.03	0.79 ± 0.02	0.78 ± 0.03	0.78 ± 0.03
0.25	0.78 ± 0.03	0.79 ± 0.03	0.79 ± 0.03	0.78 ± 0.03	0.78 ± 0.03
0.5	0.79 ± 0.02	0.79 ± 0.03	0.79 ± 0.02	0.79 ± 0.03	0.79 ± 0.03
0.75	0.80 ± 0.02	0.80 ± 0.03	0.80 ± 0.02	0.79 ± 0.02	0.79 ± 0.03
1	0.80 ± 0.03	0.80 ± 0.02	0.80 ± 0.02	0.80 ± 0.02	0.79 ± 0.02

(c) NC classification rates with focusing on temporal lobe.

$\frac{P(\text{VOI}\|s)}{P(\text{not VOI}\|s)}$	$3\times3\times3$	$4\times4\times4$	$5\times5\times5$	$6\times6\times6$	$7\times7\times7$
0	0.76 ± 0.04	0.77 ± 0.04	0.77 ± 0.04	0.77 ± 0.04	0.77 ± 0.03
0.05	0.76 ± 0.04	0.77 ± 0.03	0.77 ± 0.03	0.76 ± 0.03	0.77 ± 0.04
0.1	0.76 ± 0.04	0.77 ± 0.04	0.77 ± 0.04	0.77 ± 0.03	0.77 ± 0.04
0.15	0.77 ± 0.04	0.78 ± 0.04	0.77 ± 0.03	0.78 ± 0.03	0.78 ± 0.04
0.2	0.77 ± 0.03	0.77 ± 0.03	0.77 ± 0.03	0.77 ± 0.04	0.77 ± 0.03
0.25	0.76 ± 0.04	0.77 ± 0.04	0.77 ± 0.04	0.77 ± 0.04	0.76 ± 0.04
0.5	0.76 ± 0.03	0.77 ± 0.04	0.76 ± 0.04	0.76 ± 0.04	0.76 ± 0.04
0.75	0.75 ± 0.04	0.75 ± 0.04	0.76 ± 0.04	0.75 ± 0.04	0.75 ± 0.04
1	0.75 ± 0.04	0.74 ± 0.03	0.75 ± 0.04	0.75 ± 0.03	0.75 ± 0.04

(d) AD classification rates with focusing on temporal lobe.

Table B.1.: Classification rates (given as mean±standard deviation) of map guided patch selection focusing on parietal lobe (a)-(b) and temporal lobe (c)-(d) using 3D patches.

$\frac{P(\text{VOI}\|s)}{P(\text{not VOI}\|s)}$	$3\times3\times3$	$4\times4\times4$	$5\times5\times5$	$6\times6\times6$	$7\times7\times7$
0	0.76 ± 0.04	0.77 ± 0.04	0.77 ± 0.03	0.77 ± 0.04	0.76 ± 0.04
0.05	0.80 ± 0.03	0.80 ± 0.03	0.80 ± 0.03	0.80 ± 0.03	0.80 ± 0.03
0.1	0.80 ± 0.02	0.80 ± 0.03	0.80 ± 0.03	0.80 ± 0.03	0.80 ± 0.03
0.15	0.81 ± 0.03	0.81 ± 0.02	0.81 ± 0.03	0.81 ± 0.03	0.81 ± 0.03
0.2	0.81 ± 0.03	0.81 ± 0.02	0.81 ± 0.02	0.81 ± 0.02	0.81 ± 0.03
0.25	0.82 ± 0.02	0.81 ± 0.02	0.82 ± 0.02	0.82 ± 0.02	0.82 ± 0.02
0.5	0.82 ± 0.02	0.82 ± 0.02	0.82 ± 0.02	0.82 ± 0.02	0.82 ± 0.02
0.75	0.82 ± 0.02	0.83 ± 0.02	0.83 ± 0.02	0.82 ± 0.02	0.82 ± 0.02
1	0.83 ± 0.02	0.83 ± 0.03	0.83 ± 0.02	0.82 ± 0.02	0.82 ± 0.02

(a) NC classification rates with focusing on parietal lobe.

$\frac{P(\text{VOI}\|s)}{P(\text{not VOI}\|s)}$	$3\times3\times3$	$4\times4\times4$	$5\times5\times5$	$6\times6\times6$	$7\times7\times7$
0	0.76 ± 0.04	0.77 ± 0.04	0.77 ± 0.04	0.77 ± 0.04	0.77 ± 0.04
0.05	0.76 ± 0.04	0.77 ± 0.04	0.77 ± 0.04	0.77 ± 0.04	0.77 ± 0.04
0.1	0.77 ± 0.04	0.77 ± 0.03	0.77 ± 0.04	0.77 ± 0.04	0.77 ± 0.03
0.15	0.77 ± 0.03	0.78 ± 0.03	0.78 ± 0.03	0.78 ± 0.03	0.78 ± 0.03
0.2	0.77 ± 0.03	0.77 ± 0.03	0.77 ± 0.04	0.78 ± 0.03	0.78 ± 0.03
0.25	0.77 ± 0.03	0.77 ± 0.03	0.78 ± 0.03	0.78 ± 0.03	0.78 ± 0.03
0.5	0.77 ± 0.04	0.77 ± 0.03	0.77 ± 0.03	0.78 ± 0.03	0.78 ± 0.03
0.75	0.77 ± 0.03	0.77 ± 0.03	0.78 ± 0.03	0.78 ± 0.03	0.78 ± 0.03
1	0.77 ± 0.03	0.77 ± 0.03	0.78 ± 0.03	0.78 ± 0.03	0.78 ± 0.03

(b) AD classification rates with focusing on parietal lobe.

$\frac{P(\text{VOI}\|s)}{P(\text{not VOI}\|s)}$	$3\times3\times3$	$4\times4\times4$	$5\times5\times5$	$6\times6\times6$	$7\times7\times7$
0	0.76 ± 0.04	0.77 ± 0.04	0.77 ± 0.03	0.77 ± 0.04	0.76 ± 0.04
0.05	0.77 ± 0.03	0.77 ± 0.03	0.77 ± 0.04	0.77 ± 0.03	0.77 ± 0.03
0.1	0.78 ± 0.03	0.78 ± 0.03	0.78 ± 0.03	0.78 ± 0.03	0.77 ± 0.03
0.15	0.78 ± 0.03	0.78 ± 0.03	0.78 ± 0.03	0.79 ± 0.03	0.78 ± 0.03
0.2	0.79 ± 0.03	0.79 ± 0.03	0.79 ± 0.03	0.79 ± 0.03	0.79 ± 0.03
0.25	0.78 ± 0.03	0.79 ± 0.03	0.79 ± 0.03	0.79 ± 0.03	0.78 ± 0.03
0.5	0.80 ± 0.03	0.80 ± 0.03	0.80 ± 0.03	0.80 ± 0.03	0.79 ± 0.02
0.75	0.80 ± 0.02	0.80 ± 0.02	0.80 ± 0.02	0.80 ± 0.03	0.79 ± 0.02
1	0.80 ± 0.03	0.80 ± 0.02	0.80 ± 0.03	0.80 ± 0.03	0.80 ± 0.02

(c) NC classification rates with focusing on temporal lobe.

$\frac{P(\text{VOI}\|s)}{P(\text{not VOI}\|s)}$	$3\times3\times3$	$4\times4\times4$	$5\times5\times5$	$6\times6\times6$	$7\times7\times7$
0	0.76 ± 0.04	0.77 ± 0.04	0.77 ± 0.04	0.77 ± 0.04	0.77 ± 0.04
0.05	0.76 ± 0.04	0.77 ± 0.04	0.76 ± 0.04	0.77 ± 0.04	0.77 ± 0.04
0.1	0.76 ± 0.04	0.77 ± 0.04	0.77 ± 0.04	0.77 ± 0.04	0.76 ± 0.04
0.15	0.77 ± 0.04	0.78 ± 0.03	0.77 ± 0.04	0.77 ± 0.03	0.77 ± 0.03
0.2	0.77 ± 0.03	0.77 ± 0.03	0.78 ± 0.03	0.78 ± 0.03	0.77 ± 0.03
0.25	0.76 ± 0.04	0.77 ± 0.04	0.77 ± 0.04	0.77 ± 0.04	0.77 ± 0.04
0.5	0.76 ± 0.03	0.77 ± 0.03	0.77 ± 0.04	0.76 ± 0.04	0.77 ± 0.03
0.75	0.75 ± 0.04	0.75 ± 0.04	0.75 ± 0.04	0.76 ± 0.04	0.75 ± 0.04
1	0.74 ± 0.04	0.75 ± 0.04	0.75 ± 0.04	0.75 ± 0.04	0.74 ± 0.04

(d) AD classification rates with focusing on temporal lobe.

Table B.2.: Classification rates (given as mean±standard deviation) of map guided patch selection focusing on parietal lobe (a)-(b) and temporal lobe (c)-(d) using 2D patches.

Bibliography

M. S. Albert, S. T. DeKosky, D. Dickson, B. Dubois, H. H. Feldman, N. C. Fox, A. Gamst, D. M. Holtzman, W. J. Jagust, R. C. Petersen, P. J. Snyder, M. C. Carrillo, B. Thies, and C. H. Phelps. The diagnosis of mild cognitive impairment due to Alzheimer's disease: Recommendations from the National Institute on Aging-Alzheimer's Association workgroups on diagnostic guidelines for Alzheimer's disease. *Alzheimer's & dementia : The Journal of the Alzheimer's Association*, 7(3):270–279, May 2011.

B. Alexeev, M. A. Forbes, and J. Tsimerman. Tensor rank: Some lower and upper bounds. *2011 IEEE 26th Annual Conference on Computational Complexity*, 10(10): 283–291, 2011. doi: 10.1109/CCC.2011.28.

A. Alzheimer. Über eigenartige Krankheitsfälle des späteren Alters. *Zeitschrift für die gesamte Neurologie und Psychiatrie*, 4:356–385, 1911. doi: 10.1007/BF02866241.

M. Bertalmío, G. Sapiro, V. Caselles, and C. Ballester. Image inpainting. In *SIGGRAPH*, pages 417–424, 2000.

B. E. Boser, I. M. Guyon, and V. N. Vapnik. A training algorithm for optimal margin classifiers. *Proceedings of the fifth annual workshop on Computational learning theory COLT 92*, 6(8):144–152, 1992. doi: 10.1145/130385.130401.

A.-L. Boulesteix, A. Bender, J. L. Bermejo, and C. Strobl. Random forest Gini importance favors SNPs with large minor allele frequency, 2011. URL `http://epub.ub.uni-muenchen.de/12224/`.

H. Braak and E. Braak. Neuropathological stageing of Alzheimer-related changes. *Acta Neuropathol*, 82(4):239–259, 1991.

L. Breiman. *Setting Up, Using, And Understanding Random Forests V4.0*, February 2003. URL `ftp://ftp.stat.berkeley.edu/pub/users/breiman/Using_random_forests_v4.0.pdf`.

L. Breiman, J. Friedman, R. Olshen, and C. Stone. *Classification and Regression Trees*. Wadsworth and Brooks, Monterey, CA, 1984.

B. Cao, D. Shen, J.-T. Sun, X. Wang, Q. Yang, and Z. Chen. Detect and track latent factors with online nonnegative matrix factorization. In *Proceedings of the*

20th international joint conference on Artifical intelligence, pages 2689–2694, San Francisco, CA, USA, 2007. Morgan Kaufmann Publishers Inc.

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(3):283–319, 1970. doi: 10.1007/BF02310791.

T. S. Cho, S. Avidan, and W. T. Freeman. The patch transform. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(8):1489 – 1501, August 2010.

A. Cichocki and A.-h. Phan. Fast local algorithms for large scale nonnegative matrix and tensor factorizations. *IEICE Transactions on Fundamentals of Electronics Communications and Computer Sciences*, E92-A(3):1–14, 2009. doi: 10.1587/transfun.E92.A.708.

A. Cichocki, R. Zdunek, S. Choi, R. Plemmons, and S.-I. Amari. Novel multi-layer nonnegative tensor factorization with sparsity constraints. *Lecture Notes in Computer Science*, 1:1–10, 2007.

A. Cichocki, R. Zdunek, A. H. Phan, and S.-I. Amari. *Nonnegative Matrix and Tensor Factorizations: applications to exploratory multiway data analysis and blind source separation*. Wiley, September 2009.

P. Comon and C. Jutten. *Handbook of Blind Source Separation*. Academic Press, 1 edition, 2010.

P. Coupé, S. F. Eskildsen, J. V. Manjón, V. Fonov, and D. L. Collins. Simultaneous segmentation and grading of anatomical structures for patient's classification: Application to Alzheimer's disease. *NeuroImage*, 59(4):3736–47, Nov. 2011. doi: 10.1016/j.neuroimage.2011.10.080.

V. de Silva and L.-H. Lim. Tensor rank and the ill-posedness of the best low-rank approximation problem. arXiv, July 2006. URL `http://arxiv.org/abs/math/0607647v2`.

A. Delacourte, J. David, N. Sergeant, et al. The biochemical pathway of neurofibrillary degeneration in aging and Alzheimer's disease. *Neurology*, 52:1158–1165, April 1999.

T. Deselaers, D. Keysers, and H. Ney. Discriminative training for object recognition using image patches. In *In CVPR 05*, pages 157–162, 2005.

K. Doi. Computer-aided diagnosis in medical imaging: historical review, current status and future potential. *Comput Med Imaging Graph*, 31(4-5):198–211, Jun-Jul 2007. doi: 10.1016/j.compmedimag.2007.02.002.

J. Driesen, L. Bosch, and H. Van. Adaptive non-negative matrix factorization in a computational model of language acquisition. In *INTERSPEECH 2009*, pages 1731–1734, 2009.

A. Drzezga, N. Lautenschlager, H. Siebner, M. Riemenschneider, F. Willoch, S. Minoshima, M. Schwaiger, and A. Kurz. Cerebral metabolic changes accompanying conversion of mild cognitive impairment into Alzheimer's disease: a PET follow-up study. *European Journal of Nuclear Medicine and Molecular Imaging*, 30:1104–1113, 2003. doi: 10.1007/s00259-003-1194-1.

L. M. Ercoli and G. W. Small. *PET in the Evaluation of Alzheimer's Disease and Related Disorders*, chapter 1, pages 3–31. Springer-Verlag, 2009.

A. C. Evans, D. L. Collins, S. R. Mills, E. D. Brown, R. L. Kelly, and T. M. Peters. 3D statistical neuroanatomical models from 305 MRI volumes. *Proc IEEE Nucl Sci Symp Med Imaging*, 3(1-3):1813–1817, 1993.

Y. Fan, S. M. Resnick, X. Wu, and C. Davatzikos. Structural and functional biomarkers of prodromal Alzheimer's disease: A high-dimensional pattern classification study. *NeuroImage*, 41(2):277–285, June 2008. doi: 10.1016/j.neuroimage.2008.02.043.

M. Friedlander and K. Hatz. Computing non-negative tensor factorizations. *Optimization Methods and Software*, 23(4):631–647, 2008. doi: 10.1080/10556780801996244.

S. Gauthier. *Clinical diagnosis and management of Alzheimer's disease*. Informa Healthcare, 3 edition, 2006. ISBN 9780415372992.

H. Guan, T. Kubota, X. Huang, X. S. Zhou, and M. Turk. Automatic hot spot detection and segmentation in whole body FDG-PET images. *2006 International Conference on Image Processing*, pages 85–88, 2006.

L. Hamel. *Knowledge discovery with support vector machines*. Wiley Series on Methods and Applications in Data Mining Series. Wiley, 2009.

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

T. Hazan, S. Polak, and A. Shashua. Sparse image coding using a 3D non-negative tensor factorization. In *International Conference of Computer Vision (ICCV)*, pages 50–57, 2005.

K. Herholz, E. Salmon, D. Perani, J. C. Baron, V. Holthoff, L. Frölich, P. Schönknecht, K. Ito, R. Mielke, E. Kalbe, G. Zündorf, X. Delbeuck, O. Pelati, D. Anchisi,

F. Fazio, N. Kerrouche, B. Desgranges, F. Eustache, B. Beuthien-Baumann, C. Menzel, J. Schröder, T. Kato, Y. Arahata, M. Henze, and W. D. Heiss. Discrimination between Alzheimer dementia and controls by automated analysis of multicenter FDG PET. *NeuroImage*, 17(1):302–316, 2002.

K. Herholz, S. Westwood, C. Haense, and G. Dunn. Evaluation of a calibrated (18)F-FDG PET score as a biomarker for progression in Alzheimer disease and mild cognitive impairment. *Journal of nuclear medicine official publication Society of Nuclear Medicine*, 52(8):1218–1226, 2011.

F. L. Hitchcock. The expression of a tensor or a polyadic as a sum of products. *J Math and Phys*, 6(1):164–189, 1927a.

F. L. Hitchcock. Multiple invariants and generalized rank of a p-way matrix or tensor. *J Math and Phys*, 7(1):39–79, 1927b.

J. M. Hoffman, K. A. Welsh-Bohmer, M. Hanson, B. Crain, C. Hulette, N. Earl, and R. E. Coleman. FDG PET imaging in patients with pathologically verified dementia. *J Nucl Med*, 41(11):1920–1928, 2000.

P. O. Hoyer and P. Dayan. Non-negative matrix factorization with sparseness constraints. *Journal of Machine Learning Research*, 5:1457–1469, 2004.

A. Hyvärinen. Statistical models of natural images and cortical visual representation. *Topics in Cognitive Science*, 2(2):251–264, 2010. doi: 10.1111/j.1756-8765.2009.01057.x.

I. A. Illán. *Análisis en Componentes de Imágenes Funcionales para la Ayuda al Diagnástico de la Enfermedad de Alzheimer*. PhD thesis, Departamento de Arquitectura y Tecnología de Computadores, June 2009.

I. A. Illán, J. M. Górriz, J. Ramírez, D. Salas-Gonzalez, M. López, F. Segovia, C. G. Puntonet, and M. Gómez-Rio. ^{18}F-FDG PET imaging for computer aided alzheimer's diagnosis. *Information Sciences*, 181(4):903–916, 2011.

K. Ishii. Clinical application of positron emission tomography for diagnosis of dementia. *Ann Nucl Med*, 16(8):515–525, Dec 2002.

K. Ishii, F. Willoch, S. Minoshima, A. Drzezga, E. P. Ficaro, D. J. Cross, D. E. Kuhl, and M. Schwaiger. Statistical brain mapping of 18F-FDG PET in Alzheimer's disease: Validation of anatomic standardization for atrophied brains. *J Nucl Med*, 42(4):548–557, April 2001.

C. R. Jack, M. S. Albert, D. S. Knopman, G. M. McKhann, R. A. Sperling, M. C. Carrillo, B. Thies, and C. H. Phelps. Introduction to the recommendations from

the National Institute on Aging-Alzheimer's Association workgroups on diagnostic guidelines for Alzheimer's disease. *Alzheimer's & dementia: the journal of the Alzheimer's Association*, 7(3):257–262, 05 2011. doi: 10.1016/j.jalz.2011.03.004.

S. J. Kiebel, J. Ashburner, J.-B. Poline, and K. J. Friston. MRI and PET coregistration–a cross validation of statistical parametric mapping and automated image registration. *NeuroImage*, 5(4):271 – 279, 1997.

J. Kim and H. Park. Fast nonnegative tensor factorization with an active-set-like method. In *High-Performance Scientific Computing High-Performance Scientific Computing*, pages 311–326. Springer London, 2012. doi: 10.1007/978-1-4471-2437-5_16.

Y.-D. Kim and S. Choi. Nonnegaative tucker decomposition. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1–9. IEEE Computer Society, June 2007. doi: 10.1109/CVPR.2007.383405.

W. E. Klunk, H. Engler, A. Nordberg, Y. Wang, G. Blomqvist, D. P. Holt, M. Bergstrom, I. Savitcheva, G.-f. Huang, S. Estrada, B. Ausen, M. L. Debnath, J. Barletta, J. C. Price, J. Sandell, B. J. Lopresti, A. Wall, P. Koivisto, G. Antoni, C. A. Mathis, and B. Langstrom. Imaging brain amyloid in Alzheimer's disease with Pittsburgh Compound-B. *Annals of Neurology*, 55(3):306–319, Mar 2004. doi: 10.1002/ana.20009.

A. Kodewitz. Development of non-linear classifiers for the analysis of pet brain scans and images of skin cancer. Diploma thesis, University of Regensburg, October 2009.

A. Kodewitz, I. Keck, A. Tomé, J. M. Górriz, and E. Lang. Exploratory matrix factorization for PET image analysis. In M. Graña Romay, E. Corchado, and M. Garcia Sebastian, editors, *Hybrid Artificial Intelligence Systems*, volume 6076 of *Lecture Notes in Computer Science*, pages 460–467. Springer Berlin / Heidelberg, 2010. doi: 10.1007/978-3-642-13769-3_56.

T. G. Kolda. Multilinear operators for higher-order decompositions. Technical Report SAND2006-2081, Sandia National Laboratories, Albuquerque, NM and Livermore, CA, April 2006. URL `http://www.prod.sandia.gov/cgi-bin/techlib/access-control.pl/2006/062081.pdf`.

I. Kotsia, S. Zafeiriou, and I. Pitas. A novel discriminant non-negative matrix factorization algorithm with applications to facial image characterization problems. *IEEE Transactions on Information Forensics and Security*, 2(3-2):588–595, 2007.

J. B. Kruskal. Rank, decomposition, and uniqueness for 3-way and n-way arrays. In R. Coppi and S. Bolasco, editors, *Multiway Data Analysis*, volume 33 of *Biometrial Journal*. Etsevier Science Publishers B.V. (North-Holland), 1989.

J. L. Lancaster, L. H. Rainey, J. L. Summerlin, C. S. Freitas, P. T. Fox, A. C. Evans, A. W. Toga, and J. C. Mazziotta. Automated labeling of the human brain: a preliminary report on the development and evaluation of a forward-transform method. *Hum Brain Mapp*, 5(4):238–242, 1997. doi: 10.1002/(SICI)1097-0193(1997)5:4<238::AID-HBM6>3.0.CO;2-4.

J. L. Lancaster, M. G. Woldorff, L. M. Parsons, M. Liotti, C. S. Freitas, L. Rainey, P. V. Kochunov, D. Nickerson, S. A. Mikiten, and P. T. Fox. Automated Talairach atlas labels for functional brain mapping. *Human Brain Mapping*, 10(3):120–131, 2000.

D. D. Lee and H. S. Seung. Learning the parts of objects by non-negative matrix factorization. *Letters to Nature*, 401:788–791, 1999.

D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. In *NIPS*, pages 556–562, 2000.

H. Lee, Y.-D. Kim, A. Cichocki, and S. Choi. Nonnegative tensor factorization for continuous EEG classification. *International Journal of Neural Systems*, 17(4):305–317, 2007.

J. S. Lee, D. D. Lee, S. Choi, and D. S. Lee. Application of nonnegative matrix factorization to dynamic positron emission tomography. In *3rd International Conference on Independent Component Analysis and Blind Signal Separation*, pages 9–13, 2001.

L.-H. Lim and P. Comon. Nonnegative approximations of nonnegative tensors. *Journal of Chemometrics*, 23(7-8):14, 2009.

J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online dictionary learning for sparse coding. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pages 689–696, New York, NY, USA, 2009. ACM. doi: 10.1145/1553374.1553463.

J. A. Maldjian, P. J. Laurienti, and J. H. Burdette. Precentral gyrus discrepancy in electronic versions of the talairach atlas. *NeuroImage*, 21(1):450–455, 2004.

P. Markiewicz, J. Matthews, J. Declerck, and K. Herholz. Robustness of multivariate image analysis assessed by resampling techniques and applied to FDG-PET scans of patients with Alzheimer's disease. *NeuroImage*, 46(2):472 – 485, 2009. ISSN 1053-8119.

G. McKhann, D. Drachman, M. Folstein, R. Katzman, D. Price, and E. M. Stadlan. Clinical diagnosis of Alzheimer's disease: report of the NINCDS-ADRDA work group under the auspices of department of health and human services task force on Alzheimer's disease. *Neurology*, 34(7):939–944, Jul 1984.

G. M. McKhann, D. S. Knopman, H. Chertkow, B. T. Hyman, C. R. Jack, C. H. Kawas, W. E. Klunk, W. J. Koroshetz, J. J. Manly, R. Mayeux, R. C. Mohs, J. C. Morris, M. N. Rossor, P. Scheltens, M. C. Carrillo, B. Thies, S. Weintraub, and C. H. Phelps. The diagnosis of dementia due to Alzheimer's disease: Recommendations from the national institute on aging-alzheimer's association workgroups on diagnostic guidelines for Alzheimer's disease. *Alzheimer's & dementia: the journal of the Alzheimer's Association*, 7(3):263–269, May 2011. doi: 10.1016/j.jalz.2011.03.005.

N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. Equation of state calculations by fast computing machines. *Journal of Chemical Physics*, 21(6):1087–1092, 1953.

S. Minoshima, B. Giordani, S. Berent, K. A. Frey, N. L. Foster, and D. E. Kuhl. Metabolic reduction in the posterior cingulate cortex in very early Alzheimer's disease. *Annals of Neurology*, 42(1):85–94, 1997.

H.-J. Möller and M. B. Graeber. The case described by alois alzheimer in 1911. *European Archives of Psychiatry and Clinical Neuroscience*, 248:111–122, 1998. doi: 10.1007/s004060050027.

M. Mørup, L. K. Hansen, and S. M. Arnfred. Algorithms for sparse nonnegative Tucker decompositions. *Neural Computation*, 20(8):2112–2131, 2008. doi: 10.1162/neco.2008.11-06-407.

L. Mosconi, M. Brys, L. Glodzik-Sobanska, S. D. Santi, H. Rusinek, and M. J. de Leon. Early detection of Alzheimer's disease using neuroimaging. *Experimental Gerontology*, 42(1-2):129 – 138, 2007.

F. Nobili, D. Salmaso, S. Morbelli, N. Girtler, A. Piccardo, A. Brugnolo, B. Dessi, S. Larsson, G. Rodriguez, and M. Pagani. Principal component analysis of FDG PET in amnestic MCI. *European Journal of Nuclear Medicine and Molecular Imaging*, 35:2191–2202, 2008. doi: 10.1007/s00259-008-0869-z.

R. Opfer, S. Kabus, T. Schneider, I. C. Carlsen, S. Renisch, and J. Sabczynski. Follow-up segmentation of lung tumors in PET and CT data. *Proceedings of SPIE*, 7260, May 2009. doi: 10.1117/12.811599.

P. Paatero. A weighted non-negative least squares algorithm for three-way "PARAFAC" factor analysis. *Chemometrics and Intelligent Laboratory Systems*, 38(2):223 – 242, 1997. doi: 10.1016/S0169-7439(97)00031-2.

A. Pascual-Montano, J. Carazo, K. Kochi, D. Lehmann, and R. D. Pascual-Marqui. Nonsmooth nonnegative matrix factorization (nsNMF). *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28:403–415, 2006. doi: 10.1109/TPAMI.2006.60.

R. Peharz and F. Pernkopf. Sparse nonnegative matrix factorization with L0-constraints. *Neurocomputing*, 80(0):38 – 46, 2012. doi: 10.1016/j.neucom.2011.09.024. Special Issue on Machine Learning for Signal Processing 2010.

D. Perani and S. F. Cappa. Brain imaging in normal aging and dementia. In J. Grafman and F. Boller, editors, *Aging and dementia*, volume 6 of *Handbook of Neuropsychology*, pages 429–452. Elsevier, 2 edition, 2001.

M. F. Peterson and M. P. Eckstein. Looking just below the eyes is optimal across face recognition tasks. *Proceedings of the National Academy of Sciences*, November 2012. doi: 10.1073/pnas.1214269109.

M. Prince, R. Bryce, and C. Ferri. World Alzheimer report 2011. `http://www.alz.co.uk/research/world-report-2011`, September 2011.

B. Quost and T. Denœux. Learning from data with uncertain labels by boosting credal classifiers. In *Proceedings of the 1st ACM SIGKDD Workshop on Knowledge Discovery from Uncertain Data*, U '09, pages 38–47, New York, NY, USA, 2009. ACM. doi: 10.1145/1610555.1610561.

J. Ramírez, R. Chaves, J. M. Górriz, I. Álvarez, D. Salas-Gonzalez, M. López, and F. Segovia. Effective detection of the Alzheimer disease by means of coronal NMSE SVM feature classification. In *Advances in Neural Networks – ISNN 2009*, volume 5552, pages 337–344. Springer Berlin / Heidelberg, lecture notes in computer science edition, 2009.

P. Razifar. *Novel Approaches for Application of Principal Component Analysis on Dynamic PET Images for Improvement of Image Quality and Clinical Diagnosis.* PhD thesis, University of Uppsala, 2005.

J.-P. Royer, N. Thirion-Moreau, and P. Comon. Computing the polyadic decomposition of nonnegative third order tensors. *Signal Processing*, 91(9):2159 – 2171, 2011. doi: 10.1016/j.sigpro.2011.03.006.

P. Saxena, D. G. Pavel, J. C. Quintana, and B. Horwitz. An automatic threshold-based scaling method for enhancing the usefulness of Tc-HMPAO SPECT in the diagnosis of Alzheimer's disease. *Medical Image Computing and Computer-Assisted Interventation —MICCAI'98*, pages 623–630, 1998.

N. Scarmeas, C. G. Habeck, E. Zarahn, K. E. Anderson, A. Park, J. Hilton, G. H. Pelton, M. H. Tabert, L. S. Honig, J. R. Moeller, D. P. Devanand, and Y. Stern. Covariance PET patterns in early Alzheimer's disease and subjects with cognitive impairment but no dementia: utility in group discrimination and correlations with functional performance. *NeuroImage*, 23(1):35 – 45, 2004.

B. Schölkopf and A. J. Smola. *Learning with Kernels - Support Vector Machines, Regularization, and Beyond*. The MIT Press, 2002.

D. H. Silverman, editor. *PET in the Evaluation of Alzheimer's Disease and Related Disorders*. Springer-Verlag, 2009.

R. A. Sperling, P. S. Aisen, L. A. Beckett, D. A. Bennett, S. Craft, A. M. Fagan, T. Iwatsubo, C. R. Jack, J. Kaye, T. J. Montine, D. C. Park, E. M. Reiman, C. C. Rowe, E. Siemers, Y. Stern, K. Yaffe, M. C. Carrillo, B. Thies, M. Morrison-Bogorad, M. V. Wagster, and C. H. Phelps. Toward defining the preclinical stages of Alzheimer's disease: Recommendations from the national institute on aging-alzheimer's association workgroups on diagnostic guidelines for Alzheimer's disease. *Alzheimer's & dementia: the journal of the Alzheimer's Association*, 7(3):280–292, 05 2011. doi: 10.1016/j.jalz.2011.03.003.

V. Stouten, K. Demuynck, and H. V. Hamme. Automatically learning the units of speech by non-negative matrix factorisation. In *Proceedings of Interspeech*, pages 1937–1940, August 2007.

C. Strobl and A. Zeileis. Danger : High power ! – exploring the statistical properties of a test for random forest variable importance. Technical Report 017, Department of Statistics, University of Munich, 2008. URL `http://epub.ub.uni-muenchen.de/2111/`.

C. Strobl, A.-L. Boulesteix, A. Zeileis, and T. Hothorn. Bias in random forest variable importance measures: Illustrations, sources and a solution, 2006. URL `http://epub.ub.uni-muenchen.de/1858/`.

J. Talairach and P. Tournoux. *Co-Planar Stereotaxic Atlas of the Human Brain*. Thieme, 1988.

C. Thiel. Classification on soft labels is robust against label noise. *Language Resources And Evaluation*, pages 65–73, 2008. doi: 10.1007/978-3-540-85563-7_14.

L. R. Tucker. Some mathematical notes on three-mode factor analysis. *Psychometrica*, 31(3):279–311, September 1966.

M. A. Turk and A. P. Pentland. Face recognition using eigenfaces. *Proceedings 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 591(1):586–591, 1991. doi: 10.1109/CVPR.1991.139758.

M. Welling and M. Weber. Positive tensor factorization. *Pattern Recognition Letters*, 22(12):1255 – 1261, 2001. doi: 10.1016/S0167-8655(01)00070-8. Selected Papers from the 11th Portuguese Conference on Pattern Recognition - RECPAD2000.

A. Wimo and M. Prince. World alzheimer report 2010. `http://www.alz.co.uk/research/world-report`, September 2010.

I. Yakushev, A. Hammers, A. Fellgiebel, I. Schmidtmann, A. Scheurich, H.-G. Buchholz, J. Peters, P. Bartenstein, K. Lieb, and M. Schreckenberger. SPM-based count normalization provides excellent discrimination of mild Alzheimer's disease and amnestic mild cognitive impairment from healthy aging. *NeuroImage*, 44(1):43 – 50, 2009. doi: 10.1016/j.neuroimage.2008.07.015.

H. Yang and G. He. Online face recognition algorithm via nonnegative matrix factorization. *Information Technology Journal*, 9:1719–1724, 2010.

S. Zafeiriou, A. Tefas, I. Buciu, and I. Pitas. Exploiting discriminant information in nonnegative matrix factorization with application to frontal face verification. *IEEE Transactions on Neural Networks*, 17(3):683–695, 2006. doi: 10.1109/TNN.2006.873291.