A Multivariate Approach to Functional Neuro Modeling

Ph.D. Thesis

Niels Jacob Sand Mørch

LYNGBY 1998

IMM-PHD-1998-

IMM

IMM
DEPARTMENT OF MATHEMATICAL MODELLING

Technical University of Denmark
DK-2800 Lyngby – Denmark

1998-05-15
nm

A Multivariate Approach to Functional Neuro Modeling

Ph.D. Thesis

Niels Jacob Sand Mørch

LYNGBY 1998

IMM-PHD-1998-

IMM

ISSN 0909–3192

Printed by IMM, Technical University of Denmark

Abstract

This Ph.D. thesis, *A Multivariate Approach to Functional Neuro Modeling*, deals with the analysis and modeling of data from functional neuro imaging experiments. A multivariate dataset description is provided which facilitates efficient representation of typical datasets and, more importantly, provides the basis for a generalization theoretical framework relating model performance to model complexity and dataset size. Briefly summarized the major topics discussed in the thesis include:

- An introduction of the representation of functional datasets by pairs of neuronal activity patterns and overall conditions governing the functional experiment, via associated micro- and macroscopic variables. The description facilitates an efficient microscopic re-representation, as well as a handle on the link between brain and behavior; the latter is obtained by hypothesizing variations in the micro- and macroscopic variables to be manifestations of an underlying system.

- A review of two microscopic basis selection procedures, namely principal component analysis and independent component analysis, with respect to their applicability to functional datasets.

- Quantitative model performance assessment via a generalization theoretical framework centered around measures of model generalization error. Only few, if any, examples of the application of generalization theory to functional neuro modeling currently exist in the literature.

- Exemplification of the proposed generalization theoretical framework by the application of linear and more flexible, nonlinear microscopic regression models to a real-world dataset. The dependency of model performance, as quantified by generalization error, on model flexibility and training set size is demonstrated, leading to the important realization that no uniformly optimal model exists.

- Model visualization and interpretation techniques. The simplicity of this task for linear models contrasts the difficulties involved when dealing with nonlinear models. Thus, a novel visualization technique for nonlinear models is proposed.

A single observation emerges from the thesis as particularly important; optimal model flexibility is a function of both the complexity and the size of the dataset at hand. This is something that has not received appropriate attention by the functional neuro modeling community so far. The observation implies that optimal model performance rarely is achieved with *black-box* models; rather, model flexibility must be matched to the specific functional dataset. The potential advantage is a model that more precisely approximates the true nature of the relationship between brain and behavior, thus paving the way for increased insight into the function of the human brain.

Resumé

(Abstract in Danish)

Nærværende Ph.D. afhandling, *A Multivariate Approach to Functional Neuro Modeling*, omhandler analyse og modellering af data fra metoder til funktionel afbildning af den menneskelige hjerne. En multivariat datasæt beskrivelse introduceres, hvilket tillader effektiv repræsentation af typiske datasæt og, endnu vigtigere, udgør fundamentet for et generaliserings-teoretisk begrebsapparat, der sammenknytter en models ydeevne med dens kompleksitet og størrelsen af datasættet. Emnerne behandlet i afhandlingen omfatter:

- En introduktion til repræsentation af funktionelle datasæt ved hjælp af par af neuronale aktivitets mønstre og generelle vilkår for det funktionelle eksperiment, via tilhørende såkaldte mikro- og makroskopiske variable. Beskrivelsen tillader effektiv mikroskopisk repræsentation, samt indsigt i sammenhængen mellem hjerne og handling; det sidste gennem en antagelse om at variationer i de mikro- og makroskopiske variable er manifestationer af et underliggende system.

- Gennemgang af to procedurer til udvælgelses af en mikroskopisk basis, nemlig *principal component analysis* og *independent component analysis*, specielt med hensyn til deres anvendelighed på funktionelle datasæt.

- Kvantitativ vurdering af model-ydeevne ved hjælp af et generaliserings-teoretisk begrebsapparat baseret på modellers generaliseringsfejl. Kun få, om nogen, eksempler på anvendelsen af generaliseringsteori indenfor funktionel hjerne modellering findes i litteraturen.

- Praktisk eksemplificering af det foreslåede begrebsapparat med anvendelse af lineære samt mere fleksible, ulineære mikroskopiske regressions modeller. Model-ydeevnens, som kvantiseret ved hjælp af generaliseringsfejl, afhængighed af model fleksibilitet og datasættets størrelse demonstreres, hvilket leder til den væsentlige erkendelse, at der ikke findes én model, der er universelt bedre end alle andre.

- Model visualiserings- og fortolkningsteknikker. Enkeltheden af denne opgave for lineære modeller står i skarp kontrast til vanskelighederne forbundet med visualisering af ulineære modeller. En ny visualiserings teknik for ulineære modeller bliver derfor foreslået.

Et enkelt faktum kommer til at fremstå som særligt vigtigt; optimal model fleksibilitet afhænger af både kompleksiteten og størrelsen af det datasæt modellen anvendes på. Dette er en ting, der hidtil ikke har været genstand for tilstrækkelig opmærksomhed indenfor det funktionelle modelleringsfelt. Observationen medfører at optimal model-ydeevne ikke altid kan opnåes med en *standard* model; istedet bør model fleksibilitet tilpasses det enkelte funktionelle datasæt. Den potentielle fordel er en model der bedre approksimerer den underliggende sammenhæng mellem hjerne og handling, og således viser vejen mod øget indsigt i hjernens virkemåde.

Preface

This thesis serves as partial fulfillment of the requirements for the Ph.D. degree. The work has been funded by the Danish Research Academy and carried out partly at the Technical University of Denmark, and partly at the Copenhagen University Hospital.

> "Causes shall not be multiplied beyond necessity",
>
> *William of Occam (1285–1349)*

These words wisely state that the simplest explanation is the best. The principle is known as *Occam's razor* and applies equally well to a Ph.D. thesis title as to everything else: a title should be just long enough to clearly and concisely convey the contents of the work that it names. The title of the present work, "A Multivariate Approach to Functional Neuro Modeling" is, however, rather long. Efforts to find a shorter, equally precise title were unsuccessful, owing to the fact that all the words are important in understanding the intention of the work:

Neuro … It is all about the brain. In fact, the title may even be too short since it holds no indication of the fact that we investigate only the *human* brain.

Functional … We are not concerned with the anatomical structures of the brain, at least not per se. Rather, we investigate the *functional* behavior of the brain, implying that we engage in the study of the *living* human brain. Specifically, images of (approximated) neuronal activity form the basis on which the approach rests.

A(n) … **Approach to** … Clearly, the views presented here constitute but a single approach to understanding the living human brain. Many other approaches exist and some have considerable overlap with the one presented here.

Multivariate … The images and other measures that form the basis of the approach herein can be regarded as either *univariate* or *multivariate* stochastic variables. Many multivariate methods as well as several univariate approaches have proved themselves very viable. The view presented here, however, is strictly multivariate.

Modeling … The complex behavior exhibited by man is remarkable. From a biochemical point of view, so is the human brain. However interesting the two phenomena may seem in isolation, the really interesting questions, with equally significant answers, arise when hypothesizing that brain and behavior can be linked. It is exactly this link that we study and attempt to model.

With these guidelines in place the scene has been set and we can take a closer look on the way the thesis is organized.

Thesis overview

The thesis is organized into seven chapters and six appendices. The first two chapters serve as an introduction to functional neuro imaging and system modeling, while the next four form the main part of the thesis, concerned with multivariate analysis, modeling and visualization of functional datasets. In addition to smaller illustrative examples presented as the thesis progresses, a larger real-world dataset is analyzed continuously as methods are derived and described. In more detail the contents of the individual chapters and appendices are:

Chapter 1 briefly describes the human brain. Two ways of imaging its function (functional neuro imaging modalities) are introduced and characteristics of functional datasets given.

Chapter 2 reviews the concepts of real world systems and mathematical models thereof. Signals, inputs and outputs are defined in the context of functional datasets. The chapter concludes with a discussion of the validity of a system hypothesis in brain science.

Chapter 3 formalizes a number of vector spaces relevant to the analysis of functional datasets, and further investigates coordinate transformation methods in the space spanned by the set of preprocessed image volumes.

Chapter 4 discusses modeling as joint density estimation and focuses on generic aspects of model performance, in particular generalization and model complexity control.

Chapter 5 exemplifies the joint density estimation of chapter 4 by employing linear models in order to relate properties of macroscopic and microscopic variables. A duality between two types of linear models is shown to exist.

Chapter 6 considers nonlinear models in an attempt to improve model performance over that achievable with linear models. The complexities introduced by nonlinear models with respect to parameter estimation are assessed.

Chapter 7 summarizes the work presented and outlines possible conclusions. Suggestions for further work are also provided.

Appendix A describes the functional neuro imaging experiment and resulting dataset used to illustrate the various analysis and modeling techniques.

Appendix B provides the basics of information theory as needed to derive and understand the concepts underlying independent component analysis described in chapter 3.

Appendix C algebraically estimates the expected generalization error as defined in chapter 4.

Appendix D reproduces the author's contribution to the 1995 International Conference on Neural Networks in Perth, Australia.

Appendix E reproduces the author's contribution to the Second International Conference on Functional Mapping of the Human Brain 1996, in Boston, USA.

Appendix F reproduces the author's contribution to the 15th International Conference on Information Processing in Medical Imaging 1997 in Vermont, USA.

This list almost concludes the prefatory remarks. Before getting into the nitty-gritty, however, heartfelt appreciation must be expressed.

Acknowledgments

The author wishes to extend thanks to the Danish Research Academy for funding in the three-year period over which the Ph.D. work has been performed. Further, both the Department for Mathematical Modelling (IMM) at the Technical University of Denmark (DTU), and the Neurobiology Research Unit (NRU), Copenhagen University Hospital (RH) have provided excellent working environments, characterized by their enjoyable atmosphere and high academic standards.

The duties of supervision have been superbly carried out by Associate Professor Lars Kai Hansen from IMM, DTU, and Professor Olaf B. Paulson and Research Associate Claus Svarer from NRU, RH. All have provided invaluable advice on both academic and practical matters. Their vast experience in their respective fields have been of immense value. Also, many other staff members at both IMM, DTU as well as NRU, RH have been of great help, in particular Jan Larsen and Ian Law. Extensive collaboration with several foreign research institutions, partly made possible via grant P20 MH57180 from the US Human Brain Project, has been a source of both academic and personal development. Specifically, the author wishes to thank the staff at the Minneapolis VA Medical Center (MVAMC) for many fruitful discussions; both by means of telecommunication and during the periods of stay in Minneapolis. Warm thanks go to Stephen C. Strother for his always open-minded attitude and our many discussions on everything from manifolds in high-dimensional vector spaces to the more dubious details of the English language. For making my six months at the Brain Image Analysis Laboratory (BIAL) at the University of California, San Diego (UCSD) a productive and great time, both professionally and personally, warm thanks go to Professor Terry Jernigan and the rest of the laboratory staff.

Thanks also go to my fellow Ph.D. students for many hours of fun. Special thanks go to Ulrik Kjems, Mads Hintz-Madsen, and Peter Alshede Philipsen for sharing office space with me and for keeping up with my mess and peculiarities. Finally, many heartfelt thanks go to my family and friends for their never ending support and encouragement during the many hours of frustrated writing.

Lyngby, May 1998,

Niels Mørch

Nomenclature

An attempt has been made to use symbols, operators, and names of variables consistently throughout the text, such that e.g. $\mathbf{z}$ almost always denotes a vector of projections. However, exceptions do exist so apart from the general guidelines below no comprehensive index of notation and meaning is provided; instead the introduction of naming conventions is left to the chapter where they first occur.

Vectors and matrices

Unless otherwise stated all vectors are column vectors, denoted by boldface lowercase letters

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = [x_1 \; x_2 \; \ldots \; x_n]^\top \quad ,$$

where $^\top$ denotes the vector *transpose*. Matrices are denoted by boldface uppercase letters

$$\mathbf{X} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1m} \\ x_{21} & x_{22} & \cdots & x_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nm} \end{bmatrix} = [\mathbf{x}_1 \; \mathbf{x}_2 \; \cdots \; \mathbf{x}_m] = \begin{bmatrix} \mathbf{x}_{(1)}^\top \\ \mathbf{x}_{(2)}^\top \\ \vdots \\ \mathbf{x}_{(n)}^\top \end{bmatrix} \quad .$$

Here $\mathbf{x}_j$ is the j'th column of $\mathbf{X}$, and $\mathbf{x}_{(i)}$ the i'th row of $\mathbf{X}$ arranged in a column vector. As for vectors $^\top$ denotes the matrix transpose

$$\mathbf{X}^\top = \left[\mathbf{x}_{(1)} \; \mathbf{x}_{(2)} \; \cdots \; \mathbf{x}_{(n)}\right] \quad .$$

A matrix composed from individual scalars x_{ij} is written as $\mathbf{X} = (x_{ij})$, whereas the scalar in the j'th row and i'th column of matrix $\mathbf{X}$ is denoted $\mathbf{X}_{ij}$.

A few special vectors and matrices deserves mention here. In particular, all elements of the i'th *unit vector* are zero except for the i'th element

$$\mathbf{e}_i = [0 \; \cdots \; 1 \; \cdots \; 0]^\top \quad .$$

The d-dimensional *identity matrix* consists of the unit vectors from order one up to d

$$\mathbf{I}_d = [\mathbf{e}_1 \; \mathbf{e}_2 \; \cdots \; \mathbf{e}_d] = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix} \quad ,$$

where the subscript is omitted if the dimension is clear. The identity matrix is a special example of a *diagonal matrix*, of which all off-diagonal elements are zero.

Operators

With few exceptions operators are typeset non-slanted to make them stand out in equations. Unless the operator itself is a pair of delimiters (like the determinant operator below) the operator is followed by its arguments enclosed in brackets. The type of brackets vary from operator to operator. Operators not mentioned in the list below or other sections of this nomenclature are introduced in the text where they first appear.

$\mathrm{rank}\,(\mathbf{A})$	rank of matrix $\mathbf{A}$, i.e. the maximum number of linearly independent columns.
$\mathrm{diag}\,[\mathbf{x}]$	diagonal matrix with the elements of vector $\mathbf{x}$ in its diagonal. For a matrix, $\mathrm{diag}\,[\mathbf{A}]$ is the vector of diagonal elements, $\mathbf{A}_{ii}$.
$\lvert\mathbf{A}\rvert$	determinant of the square matrix $\mathbf{A}$.
$\lVert\mathbf{A}\rVert_2$	matrix norm of $\mathbf{A}$, which does not need to be square. Corresponds to the largest singular value of $\mathbf{A}$.
$\langle z\rangle$	mean of the stochastic variable z; also labeled $\bar{z}$.
$\mathrm{V}\,[z]$	variance of the stochastic variable z.

Vector spaces

Uppercase caligraphic letters are used to denote vector spaces. In particular, d-dimensional *Euclidean space* is labeled $\mathcal{R}^d$. The subspace operator is $\subset$ so that

$$\mathcal{S} \subset \mathcal{R}^d$$

means that $\mathcal{S}$ is a subspace of $\mathcal{R}^d$. The dimension of a vector space is denoted by $\dim(\cdot)$, so we have

$$\dim(\mathcal{R}^d) = d \quad .$$

Vector space notation is further explained in section 3.1.

Other symbols

A *set* definition is enclosed in curly braces, as in

$$\{\mathbf{x}_n \mid n = 1, \ldots, N\} \quad ,$$

and named using an uppercase sans-serif letter, e.g. D. For a stochastic variable the symbol $\sim$ means "distributed as", e.g. $z \sim N(0, \sigma^2)$. The same symbol is also used to mean "in the range of" when specifying deterministic values, e.g. $d \sim 10^4$. Finally, estimates are generally accented with a "hat", as in $\hat{\mathbf{x}}$.

Acronyms

In the text some commonly used terms are abbreviated. At its first occurrence a term is written out in full, followed by the acronym in parentheses. The acronym is subsequently used instead of the term, except where context dictates otherwise, e.g. in the beginning of a sentence. A full list of acronyms, capitalized as they appear in the text and followed by the term they abbreviate, completes the nomenclature.

AIR	Automated image registration
ANN	Artificial neural network
ANOVA	Analysis of variance
BOLD	Blood oxygenation level dependent
BS	Bootstrapping
CBF	Cerebral blood flow
c.d.f.	Cumulative density function
CPH/SAC	Copenhagen saccade PET dataset
CV	Cross-validation
EOG	Electrooculography
fMRI	Functional magnetic resonance imaging
FWHM	Full width, half maximum
GLM	General linear model
ICA	Independent component analysis
LED	Light emitting diode
LM	Levenberg-Marquardt
LRT	Likelihood ratio test
MAP	Maximum a posteriori
ML	Maximum likelihood
MLP	Multi-layer perceptron
MRI	Magnetic resonance imaging
MSE	Mean square error
OBD	Optimal brain damage
OBS	Optimal brain surgeon
PCA	Principal component analysis
p.d.f.	Probability density function
PET	Positron emission tomography
RBF	Radial basis function
rCBF	Regional cerebral blood flow
RF	Radio frequency
SPM	Statistical parametric mapping

SSM Scaled subprofile model

SVD Singular value decomposition

Contents

Chapter 1

Introduction

Is there a link between brain and behavior and, perhaps, the mind? Many years of scientific investigations indicate that indeed such links exist. Most work has been concerned with the first of the two; the link between brain and behavior. This is the possible relation between the neuronal activity of the human brain and "mechanical" actions performed by humans, such as walking and performing visual recognition of faces. It is this link we shall investigate and attempt to quantify. The existence of a link between brain and mind is a different and much more involved question since it involves perceptions derived from mental entities—it is essentially concerned with consciousness and self and we will not address it here.

1.1 Measuring brain function

The human brain is amazing. It enables us to perform complex tasks in a great diversity of situations, often without apparent effort. Some neuro-biologists like to say that the sole function of the rest of the human body is to support the brain. This intriguing organ deserves thorough investigation, as a deeper understanding will provide much more than simply better chances of treating brain-related disease; it potentially opens a window to the *mind*.

Despite the prospects of tremendous insights it is only in the last few decades scientists attempting to link brain and behavior have been able to study the *normal, living* human brain in a non-invasive manner. In earlier times one had to assent to the study of the unfortunate individuals who happened to suffer brain injury. One particularly well known such case is that of Phenias T. Gage who in 1848, while setting up a charge of explosives during railroad construction, suffered severe head damage when the charge exploded and blew the tamping iron he was using straight through the front of his head (Nolte, 1993). He survived the incident and soon regained his strength. However, just as he had been hard-working and responsible before the accident, his behavior now became tact-less and impulsive. Unfortunate for the poor man as it was the episode gave valuable knowledge about the function of the prefrontal cortex.

Attempting to understand the function of (part of) the human brain may seem a daunting and almost impossible task. Indeed, it is hard, but examining ways to describe, or *quantify*, behavior is a first step. For example, consider what happens in the brain when light is flashed in a person's eyes. To assess this we might ask the subject a series of questions. We could ask her to discriminate between flashes occurring with different frequencies, colors, or intensities. This approach implicitly relies on the subject's consciousness, since it depends on her personal experience of the stimulus. This conscious-filtered way of describing behavior has a major drawback: There are many types of brain activity which are inaccessible to introspection. (There may also be other practical problems, e.g. the inability of a young child to verbally convey its experiences.) In other words, simply observing the *external* manifestations of brain function, filtered through the conscious mind, is insufficient in quantifying the link between brain and behavior. Instead, methods of measuring the "activity" inside the brain are needed.

1.1.1 Functional neuro imaging

A large number of techniques (modalities) exist for in vivo imaging of the human brain. While some, such as magnetic resonance imaging (MRI), reflect anatomical (tissue dependent) information, *functional* neuro imaging methods are characterized by their facilitation of indirect measures of the neuronal firing patterns of the brain (Malonek and Grinvald, 1996). The two techniques described below quantify the spatial and temporal distribution of blood flow and oxygenation, respectively, and thus indirectly provide three-dimensional (3D) image volumes (scans) of brain activity.

1.1.1.1 Positron emission tomography

Positron emission tomography (PET) relies on a positron emitting tracer to quantify an indirect measure of neuronal activity (Phelps, 1986). By using a tracer that is involved in biochemical processes in the brain the processes can be located and quantified. A commonly used tracer is [^{15}O] labeled water, since water is transported freely between blood and brain tissue, i.e. over the blood-brain barrier. Water injected into the blood will reach the brain and be distributed according to the regional cerebral blood flow (rCBF).

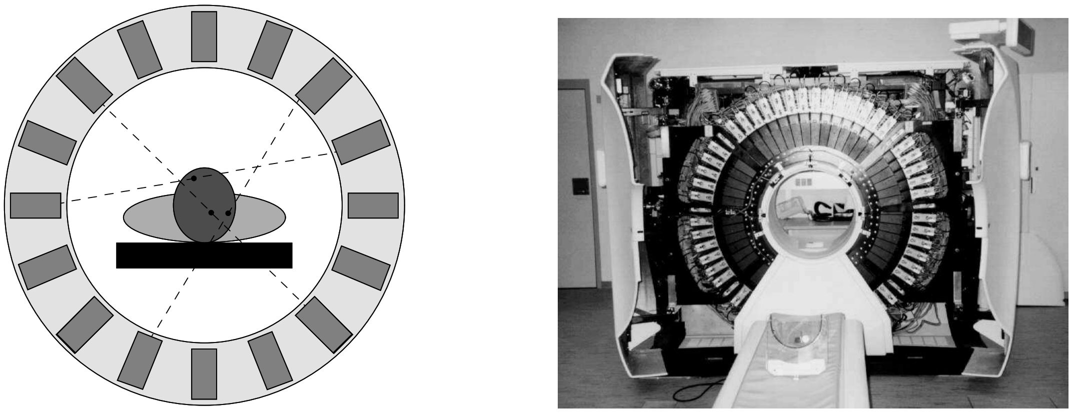

Figure 1.1: PET scanning. *Left panel*: Gamma ray coincidence detection. *Right panel*: The Advance General Electric PET scanner at the National University Hospital in Copenhagen. The front is open and the detector rings are clearly visible.

During the decay of the tracer positrons are emitted. After traveling a short distance[1] a positron annihilates with an electron from the surrounding tissue and the mass is converted

[1]The distance traveled depends on the isotope used. For [^{15}O] labeled water it is in the order of millimeters.

to electromagnetic energy in the form of two gamma quanta, emitted simultaneously in almost opposite[2] directions. By placing gamma ray detectors around the subject the emitted quanta can be detected and used to estimate the tracer distribution. Simultaneous detection in two detectors localize the annihilation on the line connecting them. By organizing detectors in rings around the subject and counting simultaneous detections the so-called *sinogram* is obtained. The sinogram is a collection of projections of the distribution of the tracer inside the detector rings. By employing the inverse Radon transform (Radon, 1917) the 3D distribution of the tracer can be reconstructed from the sinogram yielding an estimate of rCBF. The left panel of figure 1.1 outlines the principle of PET scanning. Gamma quanta emitted by positron annihilations occurring in the subject's brain are absorbed in the ring of detectors. Precise timing is used to determine the coincidences (simultaneous detections) that form the sinogram.

It is important to realize that the reconstructed tracer distribution deviates from the true distribution. The discrepancy is the result of confounding factors and propagates to models based on the estimated distribution and thus affects the conclusions drawn from such models. The confounding factors are many, but suffice it here to mention a few:

- The distribution of the distance traveled by the positrons before annihilation effectively results in blurring of the estimated tracer distribution image. As mentioned above the blurring depends on the positron energy of the isotope used.

- Due to the non-zero momentum of the emitted positrons the annihilation results in the emission of gamma quanta that are not exactly opposite. Thus, assuming that an annihilation occurred on the line connecting two detectors in which gamma quanta are detected simultaneously is incorrect, although the errors introduced in this way are small.

- Absorption and scatter of the gamma rays occur in the tissue before detection in the detector rings. These effects can be partially corrected for by acquiring a *transmission scan* prior to tracer injection using an external gamma source for each subject. The transmission scan is used to correct subsequent histograms from the same subject.

- Limitations in the detectors. The size, homogeneity, speed and separation of the detectors all affect histogram acquisition and limit both the spatial and temporal resolution of the reconstructed distributions.

- The need for a sufficient number of *counts* (detected annihilations) across the histogram coupled with the limited amount of radiation allowed for one subject argues in favor of isotopes with short half-lives. In practice [^{15}O] labeled water has a half-life of approximately two minutes.

A more rigorous treatment of the theory and practical issues relating to PET scanning can be found in (Phelps, 1986). The Radon transform and it's application to PET image reconstruction is treated in e.g. (Toft, 1996).

1.1.1.2 Functional magnetic resonance imaging

Magnetic resonance imaging (MRI) has traditionally been used to image anatomy and pathology. Independent of an injected radioactive tracer MRI is based on the magnetic

[2]The directions are not exactly opposite because of the non-zero momentum of the positron.

spin properties of hydrogen, which force the protons to align themselves with any externally applied magnetic field (Kramer and Buonanno, 1985). In the presence of a magnetic field gradient protons in selected slices of the brain volume can be excited by the application of a magnetic radio frequency (RF) pulse. The excitation induces a precessing of the proton dipoles, as indicated in the left panel of figure 1.2. By measuring the relaxation characteristics of the resulting macroscopic magnetization after the RF signal has been turned off and using phase and frequency modulation techniques it is possible to obtain an image volume that reflects the magnetic properties of the brain tissue. Macroscopic magnetization is illustrated in the right panel of figure 1.2.

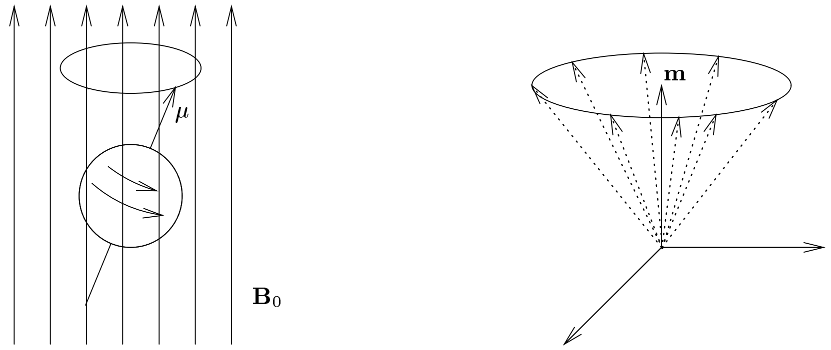

Figure 1.2: Principles of MRI scanning. *Left panel*: The magnetic dipole μ of a proton precesses (wobbles) in the presence of an external magnetic field $\mathbf{B}_0$. *Right panel*: A non-zero macroscopic magnetic moment $\mathbf{m}$ arises when the magnetic dipoles of an ensemble of precessing protons is unevenly distributed.

The application of MRI in functional neuro imaging (fMRI) was first reported by (Ogawa et al., 1990) using the blood oxygenation level dependent (BOLD) technique. The BOLD technique measures the increase in oxyhemoglobin content and its application to human subjects was reported simultaneously by several researchers, e.g. (Bandettini et al., 1992). It was demonstrated how the MR signal can be modulated by changes in rCBF and oxygen utilization as a consequence of the altered deoxyhemoglobin content; the regional onset of neuronal activity causes a rapid local increase in oxygen consumption while the increase in rCBF lags behind. The result is a local increase (in both space and time) in deoxyhemoglobin and a subsequent local alteration of the magnetic field leading to a negative BOLD signal that peaks after around 2.5 sec. The ensuing rCBF increase exceeds the elevated level of oxygen consumption; this overshoot increases oxyhemoglobin content in an area large relative to the area of initial oxygen consumption increase.

Compared to positron emission tomography fMRI holds two major advantages:

- It has considerable higher temporal resolution for comparable in-plane (single slice) spatial resolution.
- It is non-invasive and can therefore be repeated many times for a single subject.

The technique is, on the other hand, hampered by the compound and indirect nature of the signal it measures; since the effect of an rCBF increase is most readably observable in the veins problems with signal localization are likely. For a review of fMRI (Kim and Ugurbil, 1997) is a good source.

1.1.2 Preprocessing

To improve the statistical power of subsequent modeling from functional neuro imaging datasets it is desirable to have a large number of observations, each with minimal influence from confounding factors. Meeting the first of these goals poses a problem, at least when working with PET, since the amount of radiation an individual can be subjected to is limited. This means that datasets must consist of image volumes from more than one subject, thus potentially introducing inter-subject variation.

To minimize the effect of inter-subject variation and other confounding factors as discussed above, a standardized set of processing steps is applied to all acquired functional neuro imaging datasets before further analysis and modeling is performed (Strother et al., 1995a; Friston, 1994). The processing steps comprise what we call *standard preprocessing* and essentially consists of realignment, stereotactic normalization, smoothing, global CBF normalization, and masking as depicted in figure 1.3.

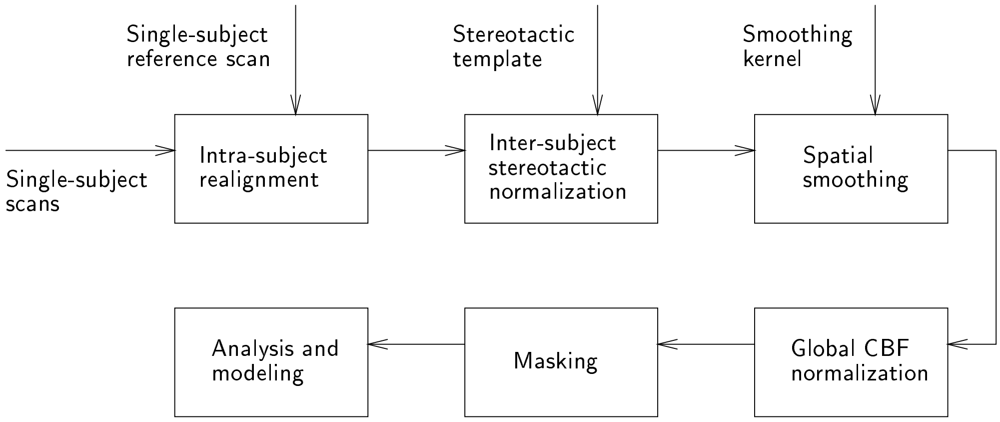

Figure 1.3: Standard preprocessing steps include intra-subject realignment, inter-subject stereotactic normalization, spatial smoothing, global CBF normalization, and masking of non-cerebral areas.

1.1.2.1 Intra-subject realignment

Realignment involves estimation and removal of intra-subject movement and is performed relative to a reference scan for each subject. Often a scan selected randomly[3] from the set of scans for each subject serves this purpose. Realignment transformations are rigid, i.e. combinations of translations and rotations. One robust intra-subject single modality approach is automated image registration (AIR) which is based on image ratio measures (Woods et al., 1993). Despite the very good performance of most intra-subject realignment schemes problems may occur; if movement between scans is large the overlap between the realigned image volumes is significantly reduced, causing masking problems as discussed later. Monitoring realignment parameters (translation and rotation) allows poorly aligned scans to be discarded from further analysis.

[3]The reference scan is selected randomly to avoid the introduction of systematic effects if the scan acquisition sequence is improperly randomized.

1.1.2.2 Inter-subject stereotactic normalization

Stereotactic normalization aims to minimize inter-subject anatomical variation, i.e. achieve the best possible coregistration of homologous areas across subjects. This is obtained by reducing differences in position, size, and shape on a subject by subject basis. Assuming correspondence, at least at a certain scale, between functional and structural anatomy (meaning that different subjects employ the same anatomical brain structures to perform a given task) stereotactic normalization is the identification of a 3D deformation that makes a subject's anatomy match a template anatomy. Template anatomy space is often referred to as *stereotactic space* (Talairach and Tournoux, 1988), hence the name stereotactic normalization.

Stereotactic normalization is hampered by the difficulties involved in quantifying inter-subject functional correspondence. Clearly, overall anatomical structures must be mapped correctly. Very detailed mapping of fine anatomical structures, however, may both be impossible and lead to lower performance of subsequent analysis and modeling procedures. This is because inter-subject variation is a mixture of measurement noise, anatomical variation and functional variation. Medium scale template based minimization of variation increases coregistration of homologous functional areas, leading to a decrease in functional variation. Extending the structural match to very small scales, however, may actually introduce variation rather than remove it, since inter-subject topographical differences may exist at these scales. This phenomenon is quantitatively investigated in (Kjems et al., 1997), in which a flexible, nonlinear stereotactic normalization procedure is proposed. See also (Woods et al., 1992; Kjems et al., 1996).

1.1.2.3 Spatial smoothing

The signal-to-noise ratio of functional neuro images can often be improved by spatial smoothing (low-pass filtering). This stems from the fact that the scale over which blood flow and oxygenation (the signal) varies is several millimeters, whereas noise typically contains higher spatial frequencies. In PET the noise is determined by the process of reconstructing the spatial distribution of blood flow from the sinogram. In fMRI the noise is approximately independent for each voxel (volume element). In both cases the spatial noise frequencies are high relative to the spatial signal frequencies. As discussed above spatial smoothing is also employed in order to match the spatial frequency contents (the image resolution) to the scale where functional anatomy can be assumed homologous, i.e. to reduce residual topographical variation. It is hard to identify an optimal smoothing kernel. In fact, it may be optimal to use a spatially varying kernel. Typically, however, a fixed Gaussian smoothing kernel with full width half maximum (FWHM) of 8–80 millimeters is used.

1.1.2.4 Global cerebral blood flow normalization

The change in estimated cerebral blood flow has two components: a global, regional-independent change and a local, regional-dependent change. The variation due to global differences must be removed in order to analyze and model the regional effects. Global variation is introduced by things like differences in injected tracer dose (for PET), gender, cardiac output, and the weight of the subject. Considerable investigation has gone into the global normalization issue (Fox and Mintun, 1989; Friston, 1994; Friston et al., 1990; Moeller et al., 1987), but complete consensus has not yet been achieved. However, it has become standard practice to normalize scans by division by the scan mean, possibly preceded by explicit multiplicative correction of measured factors, such as subject weight and injected tracer dose.

1.1.2.5 Masking

Volume images yielded by functional neuro imaging are intended for analysis and modeling of cerebral neuronal activity. However, large parts of the images cover non-cerebral areas: the skin, the skull, the ventricles, the empty space outside the head, etc. Further analysis of functional data from these areas makes little sense and so they should be masked out. To this end an intra-cerebral voxel mask volume is created for each scan. The procedure is semi-automatic and based on thresholding and the anatomical knowledge of a trained operator. For analysis and modeling of functional datasets containing more than a single scan voxels not present in all scans are discarded, i.e. analysis proceeds using a *common* mask which is the intersection of the individual scan masks.

1.2 Experimental design

Scan acquisition is only one part of a functional experiment. In the design of the experiment the researcher must attempt to induce signal changes in the neuro-physiological system of interest in an optimized fashion. This is achieved by selecting and controlling behavioral variables that govern the selected neuro-physiological system, thus introducing controlled experimental variation. In this process it is pivotal to optimize the signal-to-noise ratio: large changes in the level of neuronal activity must be induced in the system of interest (the signal) while minimizing interference from other neuro-physiological systems (behavioral noise). Since interaction between cognitive modules is abundant good experiments are difficult to design. One approach often taken is the *categorical* design of cognitive subtraction, also known as baseline-activation paradigms. Scans are acquired in two conditions: the activated state, where subjects perform a task designed to activate the system of interest, and the baseline state, which mimics the activated state except that cerebral activity in the system of interest is attempted minimized. In this way experimental variation between the group of baseline scans and the group of activated scans is introduced primarily by the neuro-physiological system of interest. In contrast the *parametric* design induces neuro-physiological signal changes by parametric, as opposed to categorical, variation of selected behavioral variables.

Regardless of experimental design we need to be able to include all relevant factors quantitatively in the analysis. We therefore introduce some notational conventions.

1.2.1 Microscopic variables

After scan acquisition and standard preprocessing the result is a set of spatially-distributed multivariate stochastic variables that quantify neuronal activity. We call this the set of *microscopic* variables. Each scan is conveniently arranged in a vector which we usually denote $\mathbf{x}$. The microscopic variables of a functional experiment with N observations can be further organized into the matrix $\mathbf{X} = [\mathbf{x}_1 \; \mathbf{x}_2 \; \cdots \; \mathbf{x}_N]$, which we call the microscopic *data matrix*.

1.2.2 Macroscopic variables

The general conditions governing the experiment are described by a corresponding set of *macroscopic* variables. These quantify relevant factors other than the neuronal activity distribution, and may include experimental manipulations, such as labels distinguishing baseline scans from activation scans, demographic and physiological measures, such as age and heart-rate, respectively, and behavioral measurements used to monitor task performance. The macroscopic variables are in general multivariate and stochastic. Like above, they are conveniently arranged in a vector which we denote $\mathbf{g}$. The macroscopic data matrix becomes $\mathbf{G} = [\mathbf{g}_1\ \mathbf{g}_2\ \cdots\ \mathbf{g}_N]$, accordingly.

1.3 Summary

Functional neuro imaging facilitates indirect quantitative spatially-distributed measures of brain function at a microscopic level. A functional experiment consists of such microscopic measurements of neuronal activity along with variables governing the macroscopic conditions under which the experiment is performed. Depending on features of the experimental design, as well as imaging modality and associated limitations, uncontrolled anatomical and topographical variations can be partially eliminated by the application of a set of standard preprocessing steps, designed to deal primarily with intra- and inter-subject alignment issues. Functional neuro modeling aims to investigate the existence of a link between brain and behavior by attempting to relate the preprocessed sets of micro- and macroscopic variables.

Chapter 2

The system hypothesis

If we can regard the human brain as a system with associated observable quantities, the process of analyzing and *understanding* it belongs to the realm of system modeling. To assess the usefulness of this approach, however, we need clear definitions of the concepts of systems and models. These in turn provide the basis for reflections on the validity of a system modeling approach in functional brain science. This chapter attempts to provide both.

2.1 Systems

A *system* can be defined as an object with variables that interact with each other and parts of the world that are external to the system itself (Ljung, 1987). The system must be clearly delimited to facilitate the distinction between it and the rest of the world, but the delimitation needs not be physical; the notion of a system is a broad concept that applies equally well to mechanical arrangements characterized by the interaction of physical forces and to, say, a display driver for a computer operating system characterized by the flow and interaction of information.

2.1.1 Signals, inputs and outputs

Here we consider only systems with real-world manifestations, i.e. systems that produce observable quantities. Systems without observable manifestations are possible, but not very interesting because we are unable to measure their behavior. The observable quantities are usually called system *outputs*, and are a subset of the total set of system variables or *signals*. We generally distinguish between output, input, and latent variables relative to the system:

Outputs are the observable signals of interest to the observer.

Inputs are external signals that affect system behavior. Normal inputs can be observed and manipulated directly by the observer, whereas *disturbances* cannot be manipulated and are observed only indirectly through their influence on the outputs.

Latent signals are internal to the system itself and cannot be directly observed. Their existence is evidenced solely by their influence on the system outputs.

Figure 2.1 contains two simple sketches of the system that governs human speech. The output is sound vibrations (changes in air pressure) and possible disturbances include characteristics of foreign bodies on the vocal chord. If the shape of the vocal tract can be quantified and measured it can be regarded as system input, as indicated in the left panel. If the shape of the vocal tract is unmeasurable, on the other hand, an extension of the system boundaries, as indicated in the right panel, to include the parts of the central nervous system that relate to speech generation causes the variables that describe the shape of the vocal tract to become latent. From this example it is clear that system boundaries to some extent determine the classification of system signals.

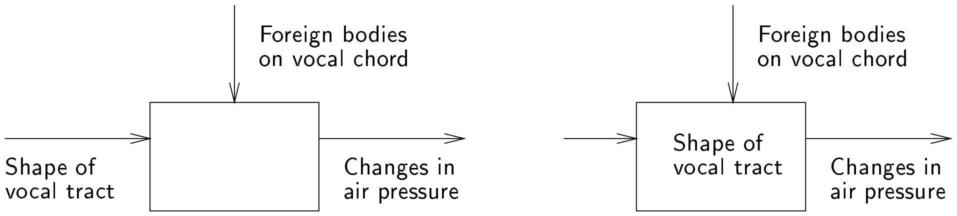

Figure 2.1: Sketches of the human speech system. *Left panel*: Regarding the shape of the vocal tract as system input. *Right panel*: By extending the system boundaries to include the parts of the central nervous system that relate to speech generation the shape of the vocal tract becomes internal to the system, i.e. it is described by latent variables.

2.1.2 Brain science and the system hypothesis

When dealing with data from experiments aimed at understanding human brain function it is convenient to *assume* that the brain can be described as a system, i.e. to employ what we will call the *system hypothesis*. It means that the brain is viewed as an unknown "black-box" with associated, measurable signals. The signals and their characterization varies depending on what we include in the definition of the brain system, just as in the simple case above. From the set of macroscopic variables that was defined in section 1.2 we are able to identify

The inputs as the variables that characterize the task to be performed. They could include task labels (baseline or activation), and parameters that quantify task difficulty.

The outputs as behavioral measures that describe task performance and other manifestations of the neuronal activity that takes place, such as limb movement.

Disturbances to include influences that cannot be manipulated, both measurable ones like age and body weight, and unmeasurable ones.

If we define the system "black-box" to be the human body and measurements are performed only by external observation, the inputs, outputs and disturbances are described as above. This is the case in traditional psychological experiments. In this setting the microscopic variables of section 1.2, i.e. the spatial distribution of neuronal activity, are latent signals, as they are internal to the system and their existence reflected only by their influence on the system outputs, i.e. the movements and other actions of the subject. However, the application of functional neuro imaging allows us to shift the system boundary by regarding the images of estimated neuronal activity (the microscopic variables) as system outputs, thus opening the door to quantitative analysis and, potentially, deeper understanding of the brain. The shift identifies the previously defined microscopic variables $\mathbf{x}$ as system outputs as mentioned, and the macroscopic variables $\mathbf{g}$ as system inputs[1]. With associated marginal probability density functions (p.d.f.'s) $\mathbf{x} \sim \mathrm{p}(\mathbf{x})$ and $\mathbf{g} \sim \mathrm{p}(\mathbf{g})$ the system is, in terms of the observable quantities, governed by the joint input-output distribution

$$\mathrm{p}(\mathbf{x}, \mathbf{g}) = \mathrm{p}(\mathbf{g}|\mathbf{x})\mathrm{p}(\mathbf{x}) = \mathrm{p}(\mathbf{x}|\mathbf{g})\mathrm{p}(\mathbf{g}) \quad . \tag{2.1}$$

In this presentation we are, in essence, concerned with estimation and interpretation of properties of this distribution.

It is important to keep in mind that the value of models based on the notion of a brain system is heavily dependent on the assumption that such a system exists. In other words, we assume that some sort of systematic relation exists between the input, output, and latent variables defined above. The existence of such a link between brain and behavior is evidenced by many aspects of modern neuro science—see (Frackowiak et al., 1997) for a discussion. The extent to which the brain system hypothesis warrants a similar hypothesis regarding the existence of a link between brain and *mind* is, however, a different and more involved question with deep philosophical aspects. The topic shall not be addressed herein.

2.2 Models

The general nature of the system definition above emphasizes the important role the concept plays in modern science; by assuming an underlying system we are able to quantitatively investigate the relationships between its variables. Such assumed relationships between system variables are called *models*, and the process of identifying and estimating relevant properties accordingly denoted *system modeling* (Ljung, 1987). The nature of the system determines the mathematical sophistication of the model needed to obtain satisfactory agreement between the two. Many greatly diverse fields rely on system modeling for problem solving.

In line with (2.1) we model the relation between system variables by a parameterized distribution estimate

$$\hat{\mathrm{p}}(\mathbf{x}, \mathbf{g}) = \mathrm{p}(\mathbf{x}, \mathbf{g}|\mathbf{w}) \quad , \tag{2.2}$$

where $\mathbf{w}$ denotes a set of adjustable parameters used to approximate $\mathrm{p}(\mathbf{x}, \mathbf{g})$. By identifying a proper set of parameters $\mathbf{w}^*$ we seek to make the model behave like the system

$$\mathrm{p}(\mathbf{x}, \mathbf{g}|\mathbf{w}^*) \simeq \mathrm{p}(\mathbf{x}, \mathbf{g}) \quad . \tag{2.3}$$

For this approach to succeed we must not only assume that the observed data has been generated according to some well-defined mathematical rules, i.e. that a "true" system exists; the corresponding set of "true" parameters must also fall within the set of relationships that the parameterized model can implement. In the context of functional neuro science the first of these assumption is difficult to validate; we can only compare certain aspects of the real-world system and our mathematical abstraction, but not establish exact

[1] In the following we will not consider disturbances, i.e. observable inputs like age and gender that cannot be manipulated experimentally, a special class of inputs, but rather include them in the class of normal inputs, such as those characterizing task difficulty.

connections between them. In practice the system hypothesis involves the assumed existence of a system that is governed completely by a set of mathematical rules. The second prerequisite for successful modeling relates to model complexity and will be discussed in later chapters.

2.2.1 Modeling from data

In general knowledge about a system comes from observations. We could also say that systems are modeled from data. This is true both for mental models of everyday tasks such as moving one's body around the physical world, and for more mathematical models like the distribution estimate in (2.2). Two rather different approaches to modeling can be employed; for

A known system we may have well-tested models for all sub-systems comprising the overall system. In a mechanical system such sub-systems could be springs and electric motors for which valid models already exist. Proper values for the parameters of the sub-systems models may even be known in advance (maybe they are provided by the vendor[2]) so the complete system can be accurately modeled by joining together the individual sub-system models. This building-block approach is generally referred to simply as *modeling*.

An unknown system, where we have no knowledge other than what we observe, measurements of the inputs and outputs are used to infer a model. This modeling-from-data approach is often called *system identification* and is what we shall focus on here[3].

The process of modeling from data involves both data acquisition, application of candidate models, and model performance assessment. In the current context the first of these steps entails experimental design and functional brain scanning, as discussed in sections 1.2 and 1.1, respectively. This results in a number of simultaneous observations of $\mathbf{x}$ and $\mathbf{g}$,

$$\mathsf{D} = \{(\mathbf{x}_n, \mathbf{g}_n) \mid n = 1, \dots, N\} \quad , \tag{2.4}$$

arranged in a dataset. The dataset used to estimate the model parameters $\mathbf{w}$ is called a *training set*. In the remainder of the thesis a number of selected—the list is by no means exhaustive—candidate models are introduced and analyzed with respect to parameter estimation. Further, we concentrate on model performance assessment and model validation, i.e. the problem of choosing the best model from a set of candidates.

2.2.2 Using functional brain models

From the above it is clear that the identification and estimation of a suitable functional brain model is a difficult task with many open ends. Should we, however, prove successful we must address the usefulness of the resulting model (Mørch and Thomsen, 1994; Lundsager and Kristensen, 1996):

[2]The term "vendor" is perhaps somewhat improper; while the vendor of electric motors may supply relevant parameter information, the issue of a sub-system "vendor" in the context of the hypothesized brain-system is something the author will leave for philosophers to discuss.

[3]From a system modeling point of view a more appropriate thesis title may in fact be: "A Multivariate Approach to Functional Neuro System Identification". However, the generic meaning of "modeling" let us to the use of that word over "system identification".

Prediction The model allows for prediction of system signals. In this way a correct model facilitates robust estimation of e.g. normal neuronal activity in the visual cortex during different kinds of visual stimulation. This in turn enables identification of abnormalities and possibly enhanced treatment of disease.

Interpretation A successful model potentially provides insight into the function of the brain; the model can be investigated in an attempt to identify the features that the model emphasizes. This process is closely connected to model visualization which is a topic of chapters 5 and 6.

The second of these uses is currently being pursued in many research laboratories across the globe, steadily increasing our knowledge about human brain function. Further, examples of the first use listed above are beginning to appear in clinical environments, aiding neurosurgeons in their quest not to harm normal tissue during surgery.

2.3 Summary

In a neuro-biological context the system hypothesis entails the assumption of the human brain being a system in a mathematical sense; well-defined relationships among the associated signals are assumed to exist. Depending on the specific classification of signals the system hypothesis implies that insight into human behavior (but not necessarily the human mind) may be gained by modeling properties of the joint probability distribution of system inputs and outputs. These, in turn, may be approximated by the micro- and macroscopic variables of functional datasets.

Chapter 3

Coordinate transformations

In datasets from typical functional experiments the number of elements (voxels) in the microscopic variables exceeds the number of observations by orders of magnitude, which means that the original representation is highly inefficient. This chapter briefly reviews relevant parts of linear algebra as it applies to the problem of coordinate transformations, and introduces and evaluates a number of basis selection procedures.

3.1 Euclidean vector spaces

Denote by $\mathcal{R}^d$ the d-dimensional Euclidean space, i.e. the set of all real d-dimensional vectors $\mathbf{x} = [x_1 \; x_2 \; \ldots \; x_d]^\mathsf{T}$. We say that $\mathbf{x}$ *belongs to* or *falls in* $\mathcal{R}^d$ and write $\mathbf{x} \in \mathcal{R}^d$. The vectors in the microscopic data matrix $\mathbf{X} = [\mathbf{x}_1 \; \mathbf{x}_2 \; \cdots \; \mathbf{x}_N]$ all fall in the Euclidean space with dimension, d, equal to the number of elements in each vector, i.e. $\mathbf{x}_n \in \mathcal{R}^d$. This space we shall denote *input space* and label $\mathcal{I}$ (Mørch et al., 1997),

$$\mathcal{I} \stackrel{\text{def}}{=} \mathcal{R}^d \quad . \tag{3.1}$$

3.1.1 Subspaces, bases and projections

Consider again the set of microscopic vectors. The *span* of these vectors is the set of all vectors $\mathbf{x} \in \mathcal{R}^d$ that can be generated from linear combinations of the set,

$$\operatorname{span}(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N) = \left\{ \mathbf{x} \mid \mathbf{x} = \sum_{n=1}^{N} a_n \mathbf{x}_n \right\} \quad . \tag{3.2}$$

Itself being a vector space the span is a *subspace* of $\mathcal{R}^d$. Since it is spanned by the observed (microscopic) signals we call it *signal space* and label it $\mathcal{S}$. With the microscopic vectors arranged in the data matrix $\mathbf{X} = [\mathbf{x}_1 \; \mathbf{x}_2 \; \cdots \; \mathbf{x}_N]$ we can also express signal space as the *range* of the data matrix

$$\mathcal{S} \stackrel{\text{def}}{=} \operatorname{span}(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N) = \langle \mathbf{X} \rangle \quad . \tag{3.3}$$

A set of linearly independent vectors $\mathbf{B} = [\mathbf{b}_1 \; \mathbf{b}_2 \; \cdots \; \mathbf{b}_B]$ forms a *basis* for a subspace when every vector in the subspace can be written as a linear combination of the basis vectors

$$\mathbf{x} = \sum_{b=1}^{B} a_b \mathbf{b}_b = \mathbf{B}\mathbf{a} \quad , \tag{3.4}$$

where $\mathbf{a}$ contains the *coordinates* of $\mathbf{x}$ with respect to basis $\mathbf{B}$ (we say that $\mathbf{a}$ is the *representation* of $\mathbf{x}$ using basis $\mathbf{B}$) (Scharf, 1991). Note that the microscopic data matrix itself forms a basis for signal space, provided that all microscopic vectors are linearly independent. From the definition it is further clear that the d-dimensional identity matrix, $\mathbf{I}_d = [\mathbf{e}_1 \ \mathbf{e}_2 \ \cdots \ \mathbf{e}_d]$, forms a basis for signal space[1]. We denote it the Euclidean basis; coordinates are usually given with respect to this basis and we do not differentiate between a vector and its coordinates. When dealing with more than one basis, however, ${}_{(\mathbf{B})}\mathbf{x}$ indicates that the coordinates of vector $\mathbf{x}$ are given with respect to basis $\mathbf{B}$. The coordinates of one basis, $\mathbf{A}$, with respect to another, $\mathbf{B}$, arranged in the so-called *coordinate transformation matrix* ${}_{(\mathbf{B})}\mathbf{M}_{(\mathbf{A})}$ facilitates the change in representation from the first basis to the second,

$$ {}_{(\mathbf{B})}\mathbf{x} = {}_{(\mathbf{B})}\mathbf{M}_{(\mathbf{A})} \ {}_{(\mathbf{A})}\mathbf{x} \quad , \tag{3.5} $$

i.e. a coordinate transformation[2]. If the rank of the coordinate transformation matrix equals the dimension of the span of the original basis $\mathbf{A}$,

$$ \operatorname{rank}\left({}_{(\mathbf{B})}\mathbf{M}_{(\mathbf{A})}\right) = \dim(\operatorname{span}(\mathbf{A})) \quad , \tag{3.6} $$

which is the case when ${}_{(\mathbf{B})}\mathbf{M}_{(\mathbf{A})}$ is regular, the bases span the same space.

Consider a vector, $\mathbf{x}$, in signal space, represented using the d-dimensional Euclidean basis. The *projection* of the vector onto the subspace spanned by a basis $\mathbf{B}$ is

$$ {}_{(\mathbf{B})}\hat{\mathbf{x}} = \left(\mathbf{B}^\top\mathbf{B}\right)^{-1}\mathbf{B}^\top\mathbf{x} \quad , \tag{3.7} $$

where the projection is represented using $\mathbf{B}$. The projection is a special case of (3.5) with ${}_{(\mathbf{B})}\mathbf{M}_{(\mathbf{I}_d)} = \left(\mathbf{B}^\top\mathbf{B}\right)^{-1}\mathbf{B}^\top$. If $\operatorname{rank}(\mathbf{B}) < d$, i.e. the dimension of the space spanned by basis $\mathbf{B}$ is smaller than that of basis $\mathbf{I}_d$, (3.6) does not hold and the projection in (3.7) reduces the dimension of $\mathbf{x}$'s coordinate vector by ignoring the parts of $\mathbf{x}$ that fall in the part of Euclidean space that is orthogonal to the space spanned by basis $\mathbf{B}$. If the basis vectors are mutually orthogonal and all of unit length then $\mathbf{B}$ is an *orthogonal* matrix that forms an *orthonormal* basis. It means that $\mathbf{B}^\top\mathbf{B} = \mathbf{I}$ so (3.7) simplifies to

$$ {}_{(\mathbf{B})}\hat{\mathbf{x}} = \mathbf{B}^\top\mathbf{x} \quad . \tag{3.8} $$

3.1.2 Ill-posed datasets

Even after preprocessing where extra-cerebral voxels are removed the number of elements in the microscopic vectors is typically quite large, often 40000 or more, i.e. $d \sim 10^4$. The number of observations, on the other hand, is orders of magnitude smaller, typically $N \sim 10^2$. This means that signal space is a low-dimensional subspace of input space

$$ \dim(\mathcal{S}) < \dim(\mathcal{I}) \quad \Leftrightarrow \quad \mathcal{S} \subset \mathcal{I} \quad . \tag{3.9} $$

We say that the microscopic data matrix is *ill-posed* (Lautrup et al., 1994; Mørch et al., 1997). The situation is illustrated in figure 3.1, where the three ($d = 3$) dashed vectors represent input space (3D Euclidean space). With only two ($N = 2$) observations in the microscopic data matrix signal space is the plane indicated in gray. No information about the parts of input space that are orthogonal to signal space is available.

[1]The basis vector $\mathbf{e}_i$ is the unit vector parallel to the i'th Euclidean axis, i.e. $\mathbf{e}_i = [0 \ \cdots \ 1 \ \cdots \ 0]^\top$.

[2]The notation is inspired by (Hansen et al., 1987) and may appear unnecessary complex. We shall not, however, make much use of it in the following.

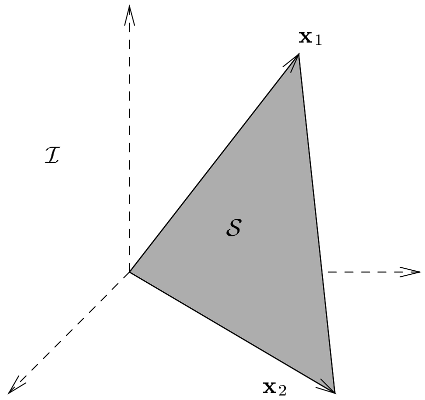

Figure 3.1: Ill-posed microscopic data matrix. With a three-dimensional input space (represented by the dashed vectors), signal space spanned by the two observations in the microscopic data matrix is the plane indicated in gray. The dataset contains no information about the parts of input space that are orthogonal to signal space because $\dim(\mathcal{S}) < \dim(\mathcal{I})$.

Now, the rank of a basis matrix is equal to the dimensionality of the space it spans, so $\text{rank}\,(\mathbf{X}) = \dim(\mathcal{S}) = \min(d, N) = N$ for an ill-posed data matrix. This signifies that the Euclidean basis is a poor choice when it comes to representing the microscopic observations efficiently in the low-dimensional signal space. By employing a projection as in (3.7) the dimensionality of the representation can be reduced. To avoid discarding information, however, we have to make sure that none of the microscopic variables fall in parts of input space that are orthogonal to the space spanned by the basis onto which we project. The basis must, in other words, span signal space.

The data matrix $\mathbf{X}$ by definition forms a basis for signal space and is an obvious candidate when looking for an efficient, loss-free dimensionality reduction of the microscopic variables

$$\mathbf{v} = {}_{(\mathbf{X})}\hat{\mathbf{x}} = \left(\mathbf{X}^\intercal\mathbf{X}\right)^{-1}\mathbf{X}^\intercal\mathbf{x} \quad , \tag{3.10}$$

which reduces[3] the dimensionality from d to N. The projection eases later analysis because of the more efficient representation of the microscopic vectors, but that is not the only reason why projection onto a signal space spanning basis is of interest; we may aim to identify a subset of *informative* projections.

3.1.3 Model space

The low number of observations, N, in a functional experiment is a confounding factor in analysis and modeling, since it limits the available degrees of freedom. It makes little sense to estimate more than N model parameters, in fact models with substantially fewer parameters often perform better, as we shall see later. In the context of vector spaces this translates to the problem of identifying a subspace of signal space, which we call *model space* and label $\mathcal{M}$, that retains the significant parts of the information in the microscopic variables. The pivotal point here is the interpretation of the word "significant".

Recall from chapter 2 that models are based on estimated properties of the joint micro- and macroscopic probability distribution. To limit the number of model parameters we

[3]Provided that all the microscopic vectors are linearly independent.

perform dimensionality reduction via a coordinate transformation *prior* to modeling, which means that information lost by projecting the microscopic variables onto a poorly selected model space basis is inaccessible in the modeling process. This in turn implies that model space should retain the parts of signal space needed to successfully model the relationships between the microscopic and macroscopic variables. It follows that model space depends on what macroscopic variables we focus our attention on. We need to have this in mind when evaluating different basis selection procedures—in the next chapter we will address that problem quantitatively. For now, however, the representation of the microscopic variables by the candidate basis vectors, i.e. the projections, will suffice to illuminate important aspects of different basis selection procedures.

3.2 Principal component analysis

The aim of principal component analysis (PCA) is to find a ranked set of orthogonal basis vectors in signal space, called principal axes, that account for as much as possible of the variance of the microscopic variables (Jackson, 1991; Mardia et al., 1979). The first principal axis is along the direction of maximum variance in signal space, the second along the direction of maximum variance in the subspace of signal space that is orthogonal to the first principal component, and so on. Before addressing the problems imposed on PCA by the ill-posed nature of the microscopic data matrix we review the basic properties of the transformation.

3.2.1 Definition and basic properties

Consider the sample covariance matrix of the microscopic variables

$$\mathbf{S} = \frac{1}{N}\sum_{n=1}^{N}(\mathbf{x}_n - \bar{\mathbf{x}})(\mathbf{x}_n - \bar{\mathbf{x}})^\top = \frac{1}{N}\mathbf{X}\mathbf{X}^\top - \bar{\mathbf{x}}\bar{\mathbf{x}}^\top \quad , \tag{3.11}$$

where $\bar{\mathbf{x}}$ is the vector of means. Spectral decomposition of $\mathbf{S}$ yields

$$\mathbf{S} = \mathbf{E}\mathbf{L}\mathbf{E}^\top \quad , \tag{3.12}$$

where $\mathbf{E} = [\mathbf{e}_1\ \mathbf{e}_2\ \cdots\ \mathbf{e}_d]$ is an orthogonal matrix of principal axes and $\mathbf{L}$ is a diagonal matrix of the eigenvalues of $\mathbf{S}$, $l_1 \geq l_2 \geq \ldots \geq l_d \geq 0$. This defines the principal component transformation as the coordinate transformation resulting from the projection onto $\mathbf{E}$, as in (3.7),

$$\mathbf{z} = \left(\mathbf{E}^\top\mathbf{E}\right)^{-1}\mathbf{E}^\top\left(\mathbf{x} - \bar{\mathbf{x}}\right) = \mathbf{E}^\top\left(\mathbf{x} - \bar{\mathbf{x}}\right) \quad , \tag{3.13}$$

where the second equality follows from $\mathbf{E}^\top\mathbf{E} = \mathbf{I}$ for the orthogonal matrix $\mathbf{E}$. The transformed variables, $\mathbf{z}$, are called the principal components. It follows from (3.12) that their sample covariance matrix is diagonal

$$\mathbf{S}_\mathbf{z} = \mathbf{E}^\top\mathbf{S}\mathbf{E} = \mathbf{E}^\top\mathbf{E}\mathbf{L}\mathbf{E}^\top\mathbf{E} = \mathbf{L} \tag{3.14}$$

so their elements are uncorrelated. Using (3.13) and (3.14) a few properties of the principal components are evident

$$\langle z_i \rangle = 0 \tag{3.15}$$

$$\mathrm{V}\left[z_i\right] = l_i \tag{3.16}$$

$$\mathrm{V}\left[z_1\right] \geq \mathrm{V}\left[z_2\right] \geq \ldots \geq \mathrm{V}\left[z_d\right] \geq 0 \quad . \tag{3.17}$$

The variance ranking of the principal component elements in (3.17) indicates that the last few principal axes often account only for a very small fraction of the total variance, suggesting that they can be ignored. Further dimensionality reduction is achieved by doing so, but the transformation is no longer loss-free. Ignoring the last principal axes corresponds to the identification of model space as the space spanned by the first few principal axes. The variance relative to the total variance

$$\tilde{l}_i = \frac{\mathrm{V}[\mathbf{z}_i]}{\sum_{j=1}^{d} \mathrm{V}[\mathbf{z}_j]} = \frac{l_i}{\sum_{j=1}^{d} l_j} \tag{3.18}$$

is a measure of the fraction of variance accounted for by the individual axes and can be used to determine the number of axes to retain. Model space identification will be addressed in more detail in chapter 4.

If the microscopic variables are multivariate Gaussians, $\mathbf{x}_n \sim N(\mu, \Sigma)$, so are the principal components

$$\mathbf{z}_n \sim N(\mathbf{0}, \Sigma_{\mathbf{z}}) \;, \qquad \hat{\Sigma}_{\mathbf{z}} = \operatorname{diag}[l_1, l_2, \ldots, l_d] \quad . \tag{3.19}$$

They are also uncorrelated and since uncorrelated Gaussians are independent, we find that the principal components of linearly mixed multivariate Gaussians have independent elements. Note, however, that PCA is just one of many transformations that result in uncorrelated elements in the transformed variables, so the identified independent components are not unique. Principal component analysis is more rigorously treated in e.g. (Mardia et al., 1979) and (Jackson, 1991).

3.2.2 Singular value decomposition

The sample covariance matrix in (3.11) is $d \times d$ which is huge for an ill-posed microscopic data matrix. This poses practical problems when computing the eigenvalues and eigenvectors that comprise the principal component transformation. However, the covariance matrix will be highly singular when the data matrix is ill-posed, since

$$\operatorname{rank}(\mathbf{S}) = \operatorname{rank}(\mathbf{X}\mathbf{X}^\top) - \operatorname{rank}(\bar{\mathbf{x}}\bar{\mathbf{x}}^\top) = \operatorname{rank}(\mathbf{X}) - 1 = N - 1 \quad . \tag{3.20}$$

This means that $\mathbf{S}$ has only $N-1$ non-zero eigenvalues; the eigenvectors corresponding to the remaining eigenvalues of zero have no relevance. The first $N-1$ eigenvectors are, in other words, enough to span signal space[4]. This observation can be utilized to avoid computing the large number of irrelevant eigenvectors.

Starting from the *centered* data matrix,

$$\mathbf{D} = [\mathbf{d}_1 \, \mathbf{d}_2 \, \cdots \, \mathbf{d}_N] \;, \qquad \mathbf{d}_n = \mathbf{x}_n - \bar{\mathbf{x}} \quad , \tag{3.21}$$

and observing that a rescaling of $\mathbf{S}$ changes only the eigenvalues in (3.12) while leaving the eigenvectors the same, we can rewrite the eigenvector equation as the spectral decomposition of $\mathbf{S}_N = N\mathbf{S} = \mathbf{D}\mathbf{D}^\top$

$$\mathbf{S}_N = \mathbf{E}\mathbf{L}_N\mathbf{E}^\top \quad , \tag{3.22}$$

[4]The need for only $N-1$ basis vectors and not N is the result of the mean removal in (3.11).

where $\mathbf{L}_N = N\mathbf{L}$ relates the eigenvalues of (3.12) and (3.22). Now, by considering the singular value decomposition (SVD) of $\mathbf{D}$

$$\mathbf{D} = \mathbf{U}\mathbf{G}\mathbf{V}^\top \quad , \tag{3.23}$$

where $\mathbf{U}$ and $\mathbf{V}$ are orthogonal matrices with the left and right singular vectors, respectively, and $\mathbf{G}$ is a diagonal matrix with the singular values, we find

$$\begin{aligned} \mathbf{S}_N &= \mathbf{D}\mathbf{D}^\top & (3.24)\\ &= \mathbf{U}\mathbf{G}\mathbf{V}^\top\mathbf{V}\mathbf{G}^\top\mathbf{U}^\top & (3.25)\\ &= \mathbf{U}\mathbf{G}^2\mathbf{U}^\top \quad . & (3.26) \end{aligned}$$

Thus we have the identities

$$\mathbf{E} = \mathbf{U} \ , \qquad \mathbf{L} = \frac{1}{N}\mathbf{L}_N = \frac{1}{N}\mathbf{G}^2 \quad , \tag{3.27}$$

while the matrix of principal components becomes

$$\begin{aligned} \mathbf{Z} &= [\mathbf{z}_1\ \mathbf{z}_2\ \cdots\ \mathbf{z}_N] & (3.28)\\ &= \mathbf{E}^\top\mathbf{D} & (3.29)\\ &= \mathbf{U}^\top\mathbf{U}\mathbf{G}\mathbf{V}^\top & (3.30)\\ &= \mathbf{G}\mathbf{V}^\top \quad . & (3.31) \end{aligned}$$

Both the principal axes, the principal components, and the PCA eigenvalues can, in other words, be found from the singular value decomposition of $\mathbf{D}$. The SVD can be computed much more efficiently than spectral decomposition of the huge sample covariance matrix $\mathbf{S}$, so (3.23) has important practical implications.

3.3 Independent component analysis

We have seen how the principal components have independent elements when computed from a set of linearly mixed multivariate Gaussians. Generally, however, uncorrelated elements of multivariate stochastic variables are not independent; it holds for multivariate Gaussians only because they are completely parameterized by the mean vector and the covariance matrix. To achieve independence in the general case we turn to independent component analysis (ICA).

3.3.1 Model assumptions

Let $s_i \mid i = 1, \ldots, N$ be *statistically independent* random variables arranged in a vector of *source signals* $\mathbf{s} = [s_1\ \ldots\ s_N]^T$. We assume the s_i's to have zero mean and unit variance. Due to the independence assumption the probability density function (p.d.f.) of $\mathbf{s}$ factors out as

$$f(\mathbf{s}) = \prod_{i=1}^{N} f_i(s_i) \quad , \tag{3.32}$$

where $f_i(s_i)$ is the p.d.f. of s_i. Further, assume that a vector of N observable *linear mixtures* is produced

$$\mathbf{x} = \mathbf{A}\mathbf{s} \tag{3.33}$$

via the invertible $N \times N$ *mixing matrix* $\mathbf{A}$. The aim of independent component analysis (Comon, 1994) and it's application to blind source separation (Jutten and Herault, 1991) is, based on the above assumptions and on realizations of $\mathbf{x}$, to estimate $\mathbf{A}$. Equivalently we may look for an *unmixing matrix* $\mathbf{W}$ such that

$$\mathbf{u} = \mathbf{W}\mathbf{x} = \mathbf{W}\mathbf{A}\mathbf{s} = \mathbf{C}\mathbf{s} = \hat{\mathbf{s}} \quad , \tag{3.34}$$

is an estimate of the source signals. Here the combined mixing-unmixing matrix $\mathbf{C}$ is called the *system matrix*. Since the ordering of the source signals is by mere convention, they can at best be recovered up to a permutation. Further, the product of each source signal s_i and the corresponding column in the mixing matrix $\mathbf{a}_i$ remains constant when s_i is multiplied by a scalar factor λ_i as long as $\mathbf{a}_i$ is correspondingly multiplied by λ_i^{-1}. Even with the normalization assumption of zero mean and unit variance for each source signal, the signs of the source signal estimates therefore remain undetermined. Thus, if we by a quasi-identity matrix $\mathbf{I}_q$ define the product of a permutation matrix with a diagonal matrix with unit-norm diagonal elements, we seek to identify a $\mathbf{W}$ that satisfies

$$\mathbf{C} = \mathbf{W}\mathbf{A} = \mathbf{I}_q \quad . \tag{3.35}$$

Due to the sign and ordering indeterminations the parameter space $\mathcal{W} = \langle \mathbf{I}_{N\times N} \rangle$ has a high degree of symmetry; many $\mathbf{W}$'s result in the same effective system matrix.

In the present context of functional neuro imaging equation (3.33) has two possible interpretations; we may assume independence in either time or space. In the first case the observed signal in a single voxel over time (across scans) is assumed to be a linear mixture of independent time (scan) profiles. This time-delayed correlation approach is investigated in (Molgedey and Schuster, 1997; Hansen and Larsen, 1998). Alternatively, independence in space implies that the microscopic variables are generated as linear mixtures of N spatial patterns of neuronal activity with independent p.d.f.'s, as discussed in (McKeown et al., 1998). It is this latter approach we shall employ here. A discussion of the extent to which the assumption of spatial independent source patterns is realistic is deferred to section 3.4.2. As a notational aside we observe that the mixtures $\mathbf{x}$ in (3.33) in the latter case are N-dimensional and thus correspond to *rows* of the microscopic data matrix $\mathbf{X}$.

3.3.2 Maximum likelihood estimation

Consider N independent realization $\mathbf{x}_1, \dots, \mathbf{x}_N$ of the vector of mixtures, distributed as

$$f(\mathbf{x}) = f(\mathbf{A}\mathbf{s}) \quad . \tag{3.36}$$

Let $f_{\mathbf{W}}(\mathbf{x})$ denote a model of the density $f(\mathbf{x})$ parameterized by $\mathbf{W} \in \check{\mathcal{W}}$, where $\check{\mathcal{W}}$ is the subspace of $\mathcal{W}$ that consists of all invertible $N \times N$ matrices. According to (B.7) the negative log-likelihood that the sample is drawn from $f_{\mathbf{W}}(\mathbf{x})$ converges in probability to the cross-entropy

$$\mathrm{L}(\mathbf{W}) = \langle -\log f_{\mathbf{W}}(x) \rangle_{f(x)} \tag{3.37}$$

as N goes to infinity. By setting

$$f_{\mathbf{W}}(\mathbf{x}) = \frac{f_{\mathbf{W}}(\mathbf{x})}{f(\mathbf{x})} f(\mathbf{x}) \tag{3.38}$$

in the above and employing (B.9), the asymptotic normalized negative log-likelihood becomes

$$\mathrm{L}(\mathbf{W}) = \mathrm{K}\left[f(\mathbf{x}); f_{\mathbf{W}}(\mathbf{x})\right] + \mathrm{H}\left[f(\mathbf{x})\right] \quad . \tag{3.39}$$

Since the entropy of the mixtures,

$$\mathrm{H}\left[f(\mathbf{x})\right] = \mathrm{H}\left[f(\mathbf{As})\right] \quad , \tag{3.40}$$

is independent on the unmixing matrix $\mathbf{W}$, maximum likelihood estimation involves minimization of the cost function

$$\mathrm{ML}(\mathbf{W}) = \mathrm{K}\left[f(\mathbf{x}); f_{\mathbf{W}}(\mathbf{x})\right] \quad . \tag{3.41}$$

Simplifying the notation by using $\mathbf{x}$ for $f(\mathbf{x})$ and $\tilde{\mathbf{x}}$ for $f_{\mathbf{W}}(\mathbf{x})$ we get

$$\begin{aligned} \mathrm{ML}(\mathbf{W}) &= \mathrm{K}\left[\mathbf{x}; \tilde{\mathbf{x}}\right] && (3.42)\\ &= \mathrm{K}\left[\mathbf{x}; \mathbf{W}^{-1}\tilde{\mathbf{s}}\right] && (3.43)\\ &= \mathrm{K}\left[\mathbf{Wx}; \tilde{\mathbf{s}}\right] \quad , && (3.44) \end{aligned}$$

where we have used the relationship (3.34). In (3.44) the p.d.f.

$$\mathrm{p}(\tilde{\mathbf{s}}) = \prod_{i=1}^{N} p_i(\tilde{s}_i) \quad , \tag{3.45}$$

of the random variable $\tilde{\mathbf{s}}$ approximates the joint p.d.f. of the source signals, $f(\mathbf{s})$.

Now, if we let $g_i(\tilde{s}_i)$ be the cumulative density function (c.d.f.) of $p_i(\tilde{s}_i)$

$$g_i(\tilde{s}_i) = \int_{-\infty}^{\tilde{s}_i} p_i(\tilde{s}_i)\, d\tilde{s}_i \quad , \tag{3.46}$$

and employ (B.14) and (B.15), the cost function in (3.44) becomes

$$\begin{aligned} \mathrm{ML}(\mathbf{W}) &= \mathrm{K}\left[\mathbf{Wx}; \tilde{\mathbf{s}}\right] && (3.47)\\ &= \mathrm{K}\left[\mathbf{g}(\mathbf{Wx}); \mathbf{g}(\tilde{\mathbf{s}})\right] && (3.48)\\ &= \mathrm{K}\left[\mathbf{g}(\mathbf{Wx}); \mathbf{1}(\tilde{\mathbf{s}})\right] && (3.49)\\ &= -\mathrm{H}\left[\mathbf{g}(\mathbf{Wx})\right] \quad . && (3.50) \end{aligned}$$

Denoting $\mathbf{y} = \mathbf{g}(\mathbf{Wx}) = \mathbf{g}(\mathbf{u})$ we have shown that maximum likelihood estimation of $\mathbf{W}$ is achieved by maximizing[5] the joint entropy of the nonlinearly transformed source signal estimates $\mathbf{y}$. For more details, see (Cardoso, 1997).

3.3.3 Information maximization

The result above can also be derived using the information maximization principle, as in e.g. (Bell and Sejnowski, 1995a; Bell and Sejnowski, 1995b). To briefly sketch the approach we re-iterate the aim of ICA: Identify $\mathbf{W}$ to obtain *statistically independent* source signal estimates $\mathbf{u}$. Independence is achieved when

$$f(\mathbf{u}) = \prod_{i=1}^{N} f(u_i) \quad \Leftrightarrow \quad \text{all } u_i\text{'s are statistically independent,} \tag{3.51}$$

[5]Obviously, maximization of the entropy corresponds to minimization of the negative entropy.

which can be quantified by the Kullback-Leibler entropy

$$\mathrm{K}\left[f(\mathbf{u}); \prod_{i=1}^{N} f(u_i)\right] \quad . \tag{3.52}$$

With the straight-forward generalization of mutual information (B.19) to more than two variables

$$\mathrm{I}\left[\mathbf{u}\right] = \left\langle \log \frac{f(\mathbf{u})}{\prod_{i=1}^{N} f(u_i)} \right\rangle_{f(\mathbf{u})} \tag{3.53}$$

we see that minimum Kullback-Leibler divergence (3.52) is achieved when the mutual information (3.53) is minimized with respect to $\mathbf{W}$, resulting in maximally independent source signal estimates.

Direct minimization of $\mathrm{I}\left[\mathbf{u}\right]$ is not simple. In (Comon, 1994) a truncated Edgeworth expansion of the involved p.d.f.'s together with minimization of all pairwise mutual information terms $\mathrm{I}\left[u_i; u_j\right]$ is investigated, and shown to lead to good performance. Others, e.g. (Amari et al., 1996), propose to use a truncated Gram-Charlie expansion to evaluate the Kullback-Leibler divergence. A different approach is to conveniently rewrite (3.53) as

$$\begin{aligned}
\mathrm{I}\left[\mathbf{u}\right] &= \left\langle \log \frac{f(\mathbf{u})}{\prod_{i=1}^{N} f(u_i)} \right\rangle_{f(\mathbf{u})} && (3.54)\\
&= \left\langle \log f(\mathbf{u}) \right\rangle_{f(\mathbf{u})} - \sum_{i=1}^{N} \left\langle \log f(u_i) \right\rangle_{f(\mathbf{u})} && (3.55)\\
&= \sum_{i=1}^{N} \mathrm{H}\left[u_i\right] - \mathrm{H}\left[\mathbf{u}\right] \quad , && (3.56)
\end{aligned}$$

which shows that mutual information can be decreased either by decreasing some or all of the individual entropies $\mathrm{H}\left[u_i\right]$ or by increasing the joint entropy $\mathrm{H}\left[\mathbf{u}\right]$. Both terms diverge to infinity for an arbitrarily large unmixing matrix $\mathbf{W}$. To avoid this (Bell and Sejnowski, 1995b) introduces squashing functions $y_i = g_i(u_i)$ that limit the estimated source signals. Further, they conjecture that the "interference" from the sum of individual entropies is small when the slopes of the nonlinearities $g_i(\cdot)$ match the p.d.f.'s of the source signals, in which case ICA can be performed by maximizing the joint entropy of the nonlinearly transformed source signal estimates—so, despite the speculative nature of the above, it leads to the same result as maximum likelihood estimation under a constraint similar to (3.46). The duality between the information maximization and maximum likelihood approaches is also discussed in (MacKay, 1996) and (Cardoso, 1997).

3.3.4 Robustness

When the model source densities are correct,

$$\mathrm{p}(\tilde{\mathbf{s}}) = f(\mathbf{s}) \quad , \tag{3.57}$$

the ML estimator (3.44) of $\mathbf{W}$ is $\mathrm{ML}(\mathbf{W}) = -\mathrm{K}\left[\mathbf{WAs}; \mathbf{s}\right]$, which is minimized at $\mathbf{C}^\star = \mathbf{W}^\star\mathbf{A} = \mathbf{I}_q$ where $\mathrm{ML}(\mathbf{W}^\star) = 0$. Thus the source signals can be completely recovered[6],

[6] Assuming that we are actually able to find a global maximum $\mathbf{W}^\star$. The answer to that question lies in the properties of the procedure we employ to locate $\mathbf{W}^\star$—an issue that will be addressed shortly. (Remember that many global optima exist due to the high symmetry of $\mathcal{W}$.)

given that the model densities are correct. More interestingly, however, is the situation of incorrect source models so that (3.57) no longer holds. Suffice it here to summarize the analysis in (Cardoso, 1997) which argues that "small errors" in the source model specification lead to solutions with $\mathbf{C} = \operatorname{diag}[\lambda_i, \dots, \lambda_S] \neq \mathbf{I}$. The source signals are recovered up to a set of scaling factors so the solution is still satisfactory. Large density mismatches, on the other hand, may make the stationary point $\mathbf{C} = \operatorname{diag}[\lambda_i, \dots, \lambda_S]$ *unstable*—a situation that has been reported by many researchers on a wide range of problems. We shall investigate the effects of incorrect source density models in section 3.4.

3.3.5 Iterative entropy maximization

To find a (potentially local) maximum, $\mathbf{W}^\circ$, of the joint entropy of the nonlinearly transformed source signal estimates,

$$\mathrm{H}[\mathbf{y}] = \mathrm{H}[\mathbf{g}(\mathbf{u})] = \mathrm{H}[\mathbf{g}(\mathbf{W}\mathbf{x})] \quad , \tag{3.58}$$

one approach is to iteratively update the time-t estimate of the unmixing matrix, $\mathbf{W}_t$, along the direction of the gradient of $\mathrm{H}[\mathbf{y}]$ with respect to $\mathbf{W}$, evaluated in $\mathbf{W} = \mathbf{W}_t$. In other words, we employ gradient ascent

$$\mathbf{W}_{t+1} = \mathbf{W}_t + \eta \left. \frac{\partial \mathrm{H}[\mathbf{y}]}{\partial \mathbf{W}} \right|_{\mathbf{W}=\mathbf{W}_t} \tag{3.59}$$

for some initial estimate $\mathbf{W}_0$. However, (Amari et al., 1996) shows that (3.59) is suboptimal; optimization is better performed using what they call the *natural gradient*. This is equivalent to the *relative gradient* derived in (Cardoso and Laheld, 1996), and amounts to multiplying the absolute gradient (3.59) by $\mathbf{W}^\top\mathbf{W}$ resulting in natural gradient ascent

$$\mathbf{W}_{t+1} = \mathbf{W}_t + \eta \left. \left[\frac{\partial \mathrm{H}[\mathbf{y}]}{\partial \mathbf{W}} \mathbf{W}^\top \mathbf{W} \right] \right|_{\mathbf{W}=\mathbf{W}_t} \quad . \tag{3.60}$$

The most immediate advantage of the natural gradient over the absolute gradient is increased convergence speed, often by several orders of magnitude (Bell and Sejnowski, 1996; Cardoso and Laheld, 1996; Amari et al., 1996).

To evaluate (3.60) we first recall that with all partial derivatives arranged in the *Jacobian*

$$\mathbf{J} = \det \begin{bmatrix} \frac{\partial y_1}{\partial x_1} & \cdots & \frac{\partial y_1}{\partial x_N} \\ \vdots & & \vdots \\ \frac{\partial y_N}{\partial x_1} & \cdots & \frac{\partial y_N}{\partial x_N} \end{bmatrix} = (\det \mathbf{W}) \prod_{i=1}^{N} \frac{\partial y_i}{\partial u_i} \tag{3.61}$$

the relation between the p.d.f.'s for $\mathbf{x}$ and $\mathbf{y}$ is (Papoulis, 1991)

$$f_y(\mathbf{y}) = \frac{f_x(\mathbf{x})}{|\mathbf{J}|} \quad . \tag{3.62}$$

It is now straightforward to derive the absolute gradient

$$\begin{aligned}
\frac{\partial \mathrm{H}[\mathbf{y}]}{\partial \mathbf{W}} &= -\frac{\partial}{\partial \mathbf{W}} \langle \log f_y(\mathbf{y}) \rangle & (3.63)\\
&= -\frac{\partial}{\partial \mathbf{W}} \left\langle \log \frac{f_x(\mathbf{x})}{|\mathbf{J}|} \right\rangle & (3.64)\\
&= \left\langle \frac{\partial}{\partial \mathbf{W}} \log |\mathbf{J}| \right\rangle & (3.65)\\
&= \frac{\partial}{\partial \mathbf{W}} \log |\det \mathbf{W}| + \frac{\partial}{\partial \mathbf{W}} \log \prod_{i=1}^{N} \left| \frac{\partial y_i}{\partial u_i} \right| & (3.66)\\
&= \left[\mathbf{W}^{\mathsf{T}}\right]^{-1} + \frac{\partial}{\partial \mathbf{W}} \log \prod_{i=1}^{N} \left| \frac{\partial y_i}{\partial u_i} \right| \quad , & (3.67)
\end{aligned}$$

where the third equality (3.65) is the result of $f_x(\mathbf{x})$ being independent of $\mathbf{W}$, and the last due to one of many not-so-obvious matrix gradients, see e.g. (Scharf, 1991, page 276). In (3.67) the $\partial y_i/\partial u_i$ terms depend on $g_i(u_i)$, which in turn should match the c.d.f. of the true source distribution, $f(\mathbf{s})$. This follows from (3.46), (3.57), and the comments above about robustness, and means that source density models must be identified before the natural gradient ascent rule (3.60) can be implemented.

3.3.6 Source density models

The *asymmetric generalized logistic function*

$$\frac{\partial y}{\partial u} = y^p (1-y)^r \tag{3.68}$$

proposed in (Bell and Sejnowski, 1995b) provides source density models that via two parameters can be tweaked to become very flat ($p, r < 1$) or very peaked ($p, r > 1$), as well as symmetric ($p = r$) and asymmetric ($p \neq r$). For $p = r = 1$ we get

$$p = r = 1 \quad \Leftrightarrow \quad y = g(u) = \frac{1}{1+e^{-u}} \quad , \tag{3.69}$$

i.e. the ordinary sigmoid function.

Recall that the "peaky-ness" of a random variable s with corresponding p.d.f. $f(s)$ is quantified by the *kurtosis*

$$\mathrm{k}[s] = \frac{\langle (s - \langle s \rangle)^4 \rangle}{\langle (s - \langle s \rangle)^2 \rangle^2} - 3, \tag{3.70}$$

where 3 is subtracted in order to make the kurtosis of a random variable with Gaussian density zero. Densities more sharply peaked than a Gaussian are called *super-Gaussian* and have $\mathrm{k}[s] > 0$, whereas less sharply peaked, more flat distributions are called *sub-Gaussian* and have $\mathrm{k}[s] < 0$. Figure 3.2 shows the cumulative density ($y = g(u)$) and probability density ($\partial y/\partial u$) functions for a few combinations of p and r (solid lines)[7]. For comparison a Gaussian distribution with zero mean and unit variance is also plotted

[7] The densities were computed by numeric integration of (3.68). The resulting densities were sampled to yield empirical kurtosis estimates.

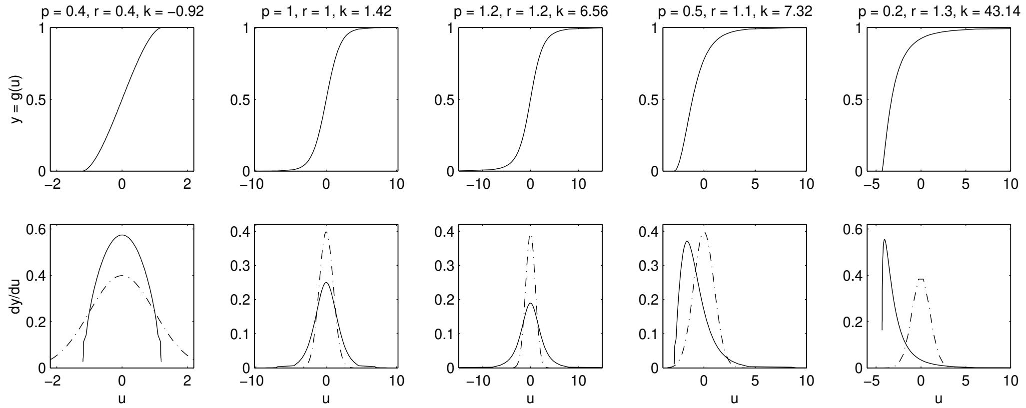

Figure 3.2: The generalized logistic cumulative density ($y = g(u)$) and probability density ($\partial y/\partial u$) functions for a few combinations of p and r (solid lines) with increasing kurtosis. The dash-dotted lines depict a Gaussian distribution with zero mean and unit variance. *Columns 1–3*: Symmetric densities ($p = r$). *Columns 4 and 5*: Asymmetric densities ($p \neq r$).

(dash-dotted lines). In the first three columns the densities are symmetric, and we see how kurtosis increases with increasing values of $p = r$. The last two columns are asymmetric ($p \neq r$) and provide a better fit for some source signals, as we shall see shortly.

To use the generalized logistic function we insert (3.68) into (3.67) and obtain

$$\frac{\partial \mathrm{H}\left[\mathbf{y}\right]}{\partial \mathbf{W}} = \left[\mathbf{W}^{\mathsf{T}}\right]^{-1} + \left(\mathrm{p}(1-\mathbf{y}) - r\mathbf{y}\right)\mathbf{x}^{\mathsf{T}} \quad , \tag{3.71}$$

which yields the natural gradient

$$\begin{aligned} \frac{\partial \mathrm{H}\left[\mathbf{y}\right]}{\partial \mathbf{W}}\mathbf{W}^{\mathsf{T}}\mathbf{W} &= \left[\left(\mathbf{W}^{\mathsf{T}}\right)^{-1} + \left(\mathrm{p}(1-\mathbf{y}) - r\mathbf{y}\right)\mathbf{x}^{\mathsf{T}}\right]\mathbf{W}^{\mathsf{T}}\mathbf{W} && (3.72)\\ &= \mathbf{W} + \left(\mathrm{p}(1-\mathbf{y}) - r\mathbf{y}\right)\mathbf{x}^{\mathsf{T}}\mathbf{W}^{\mathsf{T}}\mathbf{W} && (3.73)\\ &= \left[\mathbf{I} + \left(\mathrm{p}(1-\mathbf{y}) - r\mathbf{y}\right)\mathbf{u}^{\mathsf{T}}\right]\mathbf{W} \quad . && (3.74) \end{aligned}$$

Apart from the increased converge speed (3.74) also has the advantage of avoiding the matrix inversion in (3.71).

3.4 Examples

Before assessing the two reviewed coordinate transformations on real-world functional neuro imaging data, we will, in order to gain insight into the respective methods and their properties, pause to investigate a couple of relatively simple example datasets; one consisting of real-world sound samples, and one consisting of constructed 2D brain-like images.

3.4.1 A sound dataset

The sound dataset[8], $\mathbf{S}_s = [\mathbf{s}_1\ \mathbf{s}_2\ \mathbf{s}_3\ \mathbf{s}_4]$, consists of the four sound signals depicted in figure 3.3, each with 4200 samples. They were chosen randomly to be the sounds of a bell,

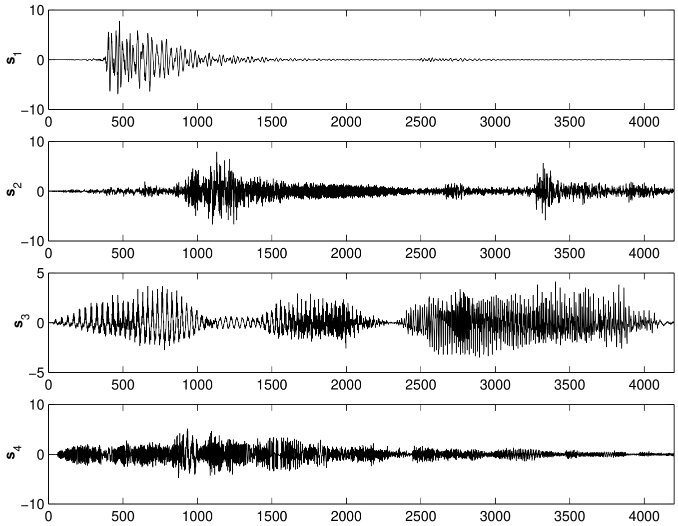

Figure 3.3: The sound dataset source signals. They are the sounds of a bell, a cash register, the phrase "easy left", and running water. All four sources are 4200 samples long.

a cash register, the phrase "easy left", and running water, respectively. The corresponding source signal histograms are shown in figure 3.4. All four sources appear to have symmetric distributions more sharply peaked than a Gaussian, which is confirmed by the source signal kurtosises: $\mathrm{k}[\mathbf{s}_1] = 15.25$, $\mathrm{k}[\mathbf{s}_2] = 7.74$, $\mathrm{k}[\mathbf{s}_3] = 1.14$, $\mathrm{k}[\mathbf{s}_4] = 2.28$. The symmetric, super-Gaussian nature is a characteristic feature of most sound signals.

We proceed to generate a random, full-rank mixing matrix $\mathbf{A}$ with values uniformly distributed between -1 and 1, and use it to generate a data matrix of four linear mixture signals, $\mathbf{X}_s^{\mathsf{T}} = \mathbf{A}_s\mathbf{S}_s^{\mathsf{T}} = [\mathbf{x}_1\ \mathbf{x}_2\ \mathbf{x}_3\ \mathbf{x}_4]^{\mathsf{T}}$, which, from figure 3.5, appear qualitatively similar. Listening to the mixtures confirms this; at best it is extremely hard for the human ear to single out the sources from the mixtures.

3.4.1.1 Principal component analysis

Before further analysis all four mixtures are normalized to unit variance. Singular value decomposition of the centered data matrix yields the principal axes as the left singular vectors, $\mathbf{E} = \mathbf{U}$. Via (3.27) and (3.15) the corresponding singular values provide the variances of the principal components. The variances relative to the total variance are found to $\tilde{l_1} = 57.04\%$, $\tilde{l_2} = 35.33\%$,, and $\tilde{l_3} = 7.63\%$ respectively. The first $N-1=3$ principal axes span signal space so projection onto them provides a loss-free dimensionality reduction as described in section 3.1.2. However, one could hope for something more, namely

[8]The notation $\mathbf{S}_s$ is used to distinguish the source data matrix from the sample covariance matrix defined in (3.11). Note that the 4200 vectors of source signals are the *rows* of $\mathbf{S}_s$, unlike in section 3.3 where source and mixture vectors are *column* vectors.

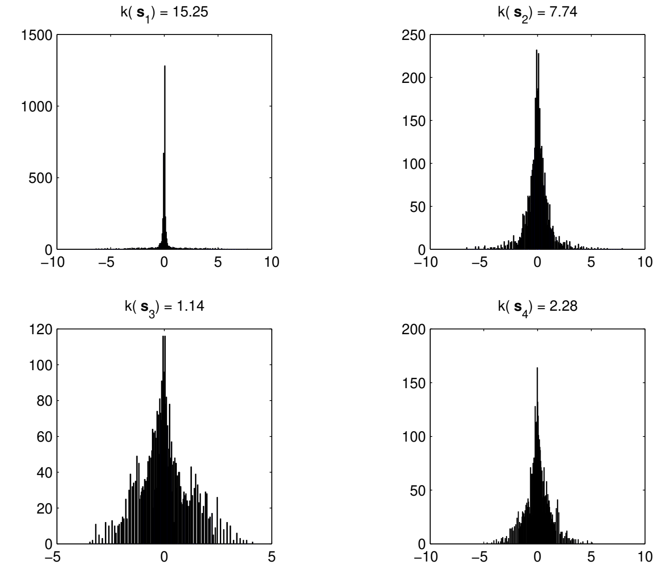

Figure 3.4: The sound dataset source signal histograms. All source signals are symmetric and have positive kurtosis, meaning that they are super-Gaussian—a characteristic feature of sound signals.

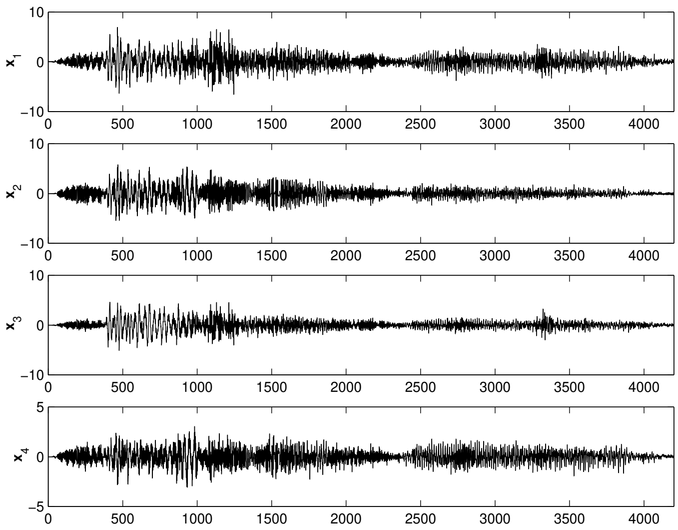

Figure 3.5: Linear mixtures of the four sound sources. All mixtures appear qualitatively similar—something that is verified by listening to them.

informative basis vectors which, in this context, translates to the recovery of the original source signals. From figure 3.6 the first principal axes looks similar to the fourth source signal. In fact, correlation of the columns of the mixing matrix, $\mathbf{a}_i$, with the principal components, $\mathbf{z}_i$, yields $\rho_{\mathbf{z}_1,\mathbf{a}_4} = -0.9942$, so we identify the principal axis that accounts for most variance as an estimate of the fourth source signal. The two remaining axes do

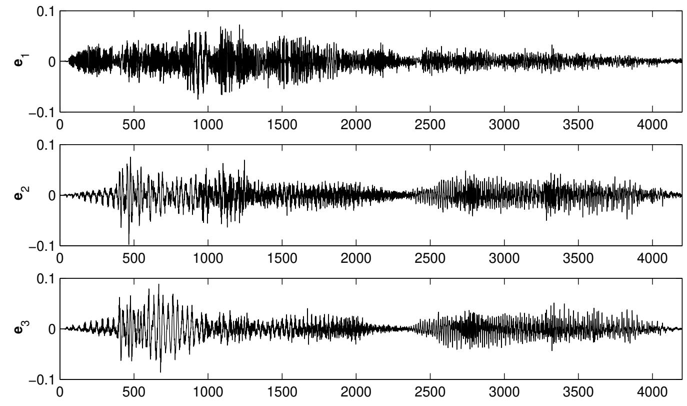

Figure 3.6: The principal axes of the set of linearly mixed sound signals. While they form a basis that provides loss-free dimensionality reduction by projection, they are not very informative. Specifically, they do not recover any of the original source signals.

not, however, recover any of the source signals. Based on this relatively simple example it seems that the principal basis provides only one informative basis vector, reducing it to little more than a loss-free dimensionality reduction—something we could achieve simply by projecting the mixtures onto themselves. Instead we look at the independent basis vectors provided by ICA.

3.4.1.2 Independent component analysis

The application of ICA involves selection of a set of source density models. From figure 3.4 and the computed kurtosises it seems that symmetric, sharply peaked models should be used. We therefore choose[9] a set of symmetric, generalized logistic density models with relatively large parameter values $p = r = [1\ 1\ 1.22\ 1.4]$. These values are found experimentally to yield density models that match the empirical distributions, at least kurtosis-wise.

The independence assumption (3.32) is key to ICA as derived in section 3.3. To test the assumption pairwise plots of the two-dimensional joint densities and marginal product densities appear in figure 3.7. For most two-source combinations there is good agreement between the two two-dimensional densities, qualitatively validating the independence assumption. However, the correlation matrix $\mathbf{P_s} = (\rho_{\mathbf{s}_i,\mathbf{s}_j})$ have several non-zero off-diagonal elements, as can be seen in figure 3.8. Accordingly, the generalized correlation coefficient $|\mathbf{P_s}|$ evaluates to $0.9762 < 1$. This means that the sources are slightly correlated and thus not completely independent. Whether or not the slight discrepancies between the two rows in figure 3.7 can be contributed solely to the small correlations or stem from higher order dependencies the violation of the independence assumption (3.32) for this dataset remains a fact; however, the sources are real-world sound signals, so dependencies are not unlikely. In any event, ICA will attempt to provide maximally independent source signal estimates.

The performance of the natural gradient ascent approach to entropy maximization is assessed by monitoring the relative transformed joint output entropy $\mathrm{H}[\mathbf{g}(\mathbf{Wx})]$ as well

[9] As reported by (Bell and Sejnowski, 1995b) and others the exact shape of the density models is not critical; good convergence is achieved for a wide range of shapes.

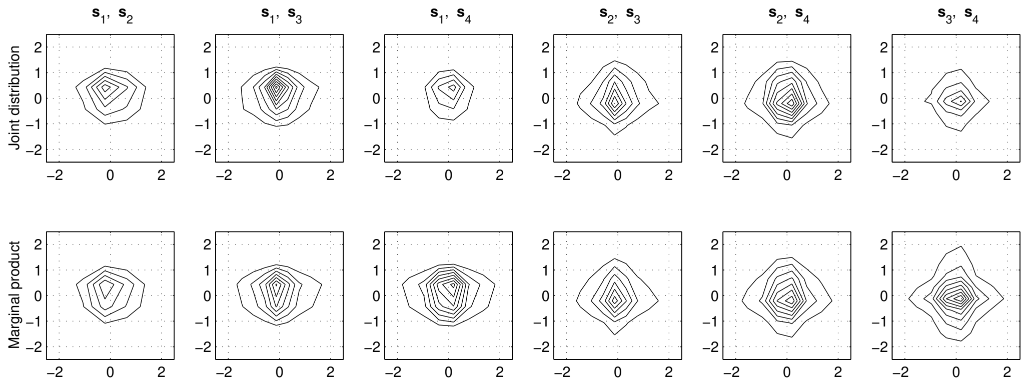

Figure 3.7: Assessing independence of the source signal elements of the sound dataset. For most two-source combinations there is good agreement between the joint density (upper row) and the product of the marginal densities (lower row).

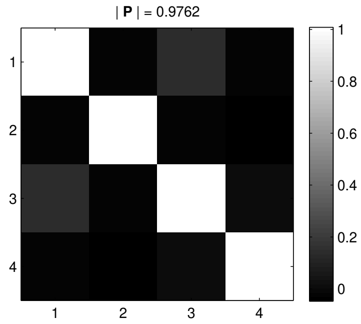

Figure 3.8: The source signal correlation matrix $\mathbf{P_s} = (\rho_{\mathbf{s}_i,\mathbf{s}_j})$ for the sound dataset. There are several non-zero off-diagonal elements, meaning that some of the sources are slightly correlated, and thus not completely independent. The generalized correlation coefficient is $|\mathbf{P_s}| = 0.9762 < 1$.

as measures based on the system matrix $\mathbf{C} = \mathbf{WA}$. The evolution of the entropy, plotted in figure 3.9, displays a sharp early increase. After relatively few iterations convergence is achieved as the curve flattens. To determine if the algorithm has successfully recovered some or all of the source signals we inspect figure 3.10, which displays the evolution of the individual elements of the system matrix, as well as the matrix norm $\|\mathbf{C}-\mathbf{I}\|_2$. As iteration progresses the system matrix evolves into an approximate quasi-identity matrix[10] $\mathbf{I}_q$: Four matrix elements converge towards -1 or $+1$ and the rest converge towards zero (upper panel). This is also reflected by the matrix norm of $\mathbf{C} - \mathbf{I}$ (lower panel)[11]. A possible explanation as to why a few system matrix elements seem to converge to small, non-zero

[10] Recall that the source signals are transformed to have zero mean and unit variance. By also scaling the source signal estimates $\mathbf{u} = \hat{\mathbf{s}}$ to unit variance the unknown scaling of $\mathbf{C}$ is removed. Note that $\mathbf{u}$ is rescaled only for the calculation of $\mathbf{C}$; the scaling does not affect $\mathbf{W}$.

[11] Since signal ordering is arbitrary the matrix norm is in fact based on $\tilde{\mathbf{C}}$, which is the column permutation of $\mathbf{C}$ that makes it as diagonal as possible.

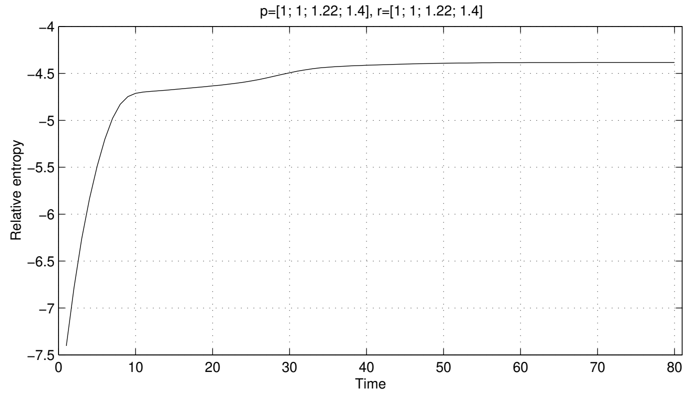

Figure 3.9: Evolution of the relative transformed joint output entropy for the sound dataset. After a sharp early increase convergence is achieved.

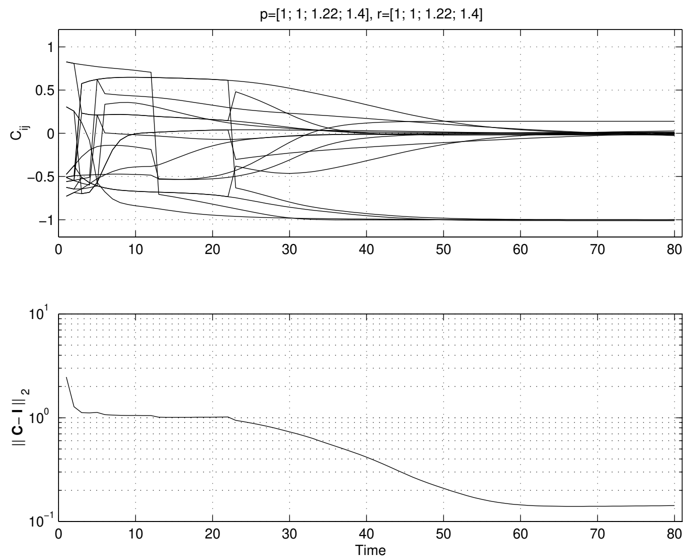

Figure 3.10: Evolution of the system matrix $\mathbf{C} = \mathbf{WA}$ for the sound dataset. *Upper panel*: The individual elements C_{ij} of the system matrix. *Lower panel*: The matrix norm $\|\mathbf{C} - \mathbf{I}\|_2$. As iteration progresses the system matrix evolves into an approximate quasi-identity matrix $\mathbf{I}_q$.

values is that the algorithm has found an unmixing matrix that results in source estimates that are more independent that the source signals themselves. The generalized correlation coefficient of the source estimates is $|\mathbf{P_u}| = 0.9995$, hinting that this is in fact the case. The non-zero elements of the system matrix are not large, though, and convergence is confirmed by looking at the independent axes, i.e. the source signal estimates, in figure 3.11. Up to a set of scaling factors, it appears that all source signals have been recovered, which is confirmed by the fact that the absolute correlation with the proper mixing vector exceeds 0.99 for all independent components.

It is interesting to investigate how the source density models affect performance. Independent component analysis is therefore applied to another set of symmetric, generalized

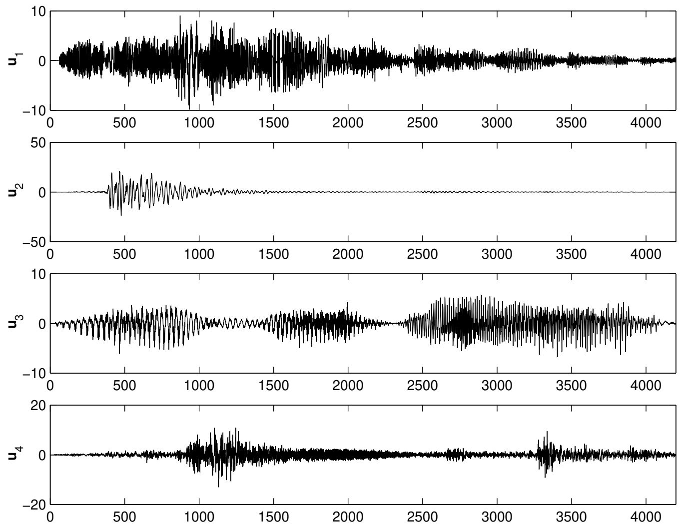

Figure 3.11: Independent axes of the set of linearly mixed sound signals. Up to a set of scaling factors, all sources have been correctly estimated.

logistic source density models, this time identical with $p = r = [1\ 1\ 1\ 1]$. Figures 3.12 and 3.13 reveal that the form of the density models is noncritical: just as for the more carefully selected density models convergence is fast, resulting in an approximate quasi-identity system matrix.

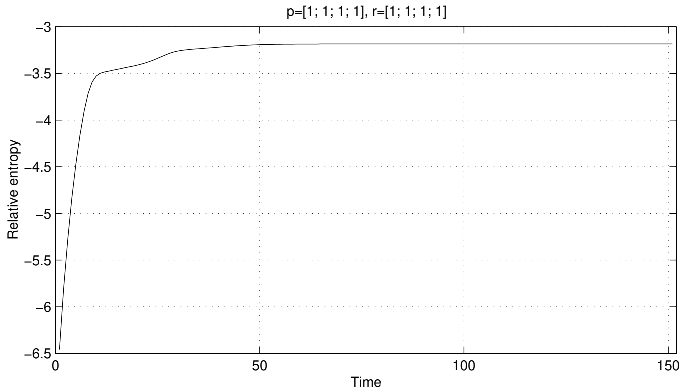

Figure 3.12: Evolution of the relative joint output entropy for the sound dataset, based on identical source density models. Convergence is just as fast as before.

We conclude that ICA performs well when applied to linear mixtures of real-world sound signals. The symmetric, super-Gaussian nature of most sound sources is however, unlikely to be mirrored in spatially distributed patterns of neuronal activity. This issue is investigated in the next section, which also addresses the more fundamental problem of assuming neuronal activity patterns to be generated as linear mixtures of a number of independent source patterns.

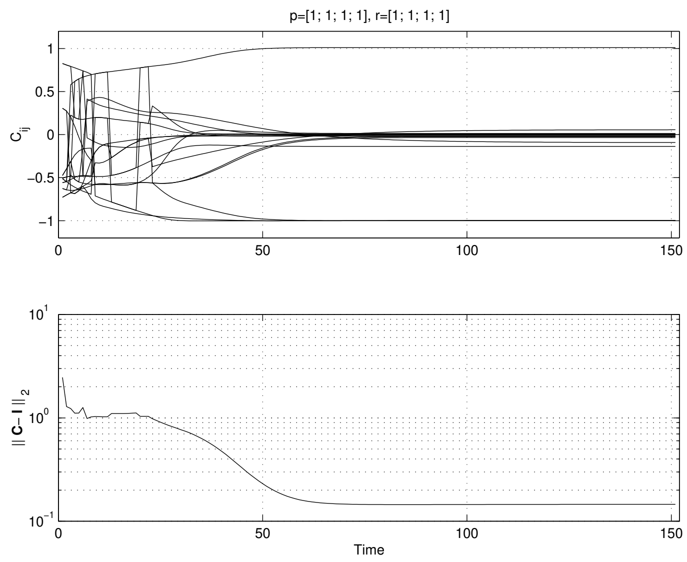

Figure 3.13: Evolution of the system matrix $\mathbf{C} = \mathbf{WA}$ for the sound dataset, based on identical source density models. *Upper panel*: The individual elements C_{ij} of the system matrix. *Lower panel*: The matrix norm $\|\mathbf{C} - \mathbf{I}\|_2$. Like for estimation based on more carefully selected density models the system matrix evolves into a quasi-identity matrix $\mathbf{I}_q$ as iteration progresses.

3.4.2 A two-dimensional brain-like dataset

The activation of a particular neuro-physiological system manifest itself as a spatial pattern of neuronal activity. This is the basic fact that facilitates functional neuro imaging. At any one time many neuro-physiological systems are involved in the complex tasks performed by a living human. The massive number of interconnections in the brain indicates heavy interaction between cognitive modules, meaning that the pattern of neuronal activity observable at a given time is a mixture of the activity of many neuro-physiological systems. Independent component analysis aims to identify such spatial activity patterns by assuming them independent, see also (McKeown et al., 1998).

The application of ICA on sets of microscopic variables with the aim of obtaining images of all involved neuro-physiological systems is bound to fail, however, since it implicitly assumes that the linear mixture model (3.33) holds for human brain function. We can not expect interactions to occur purely in a linear fashion—in fact, many *feedback loops* can be identified anatomically, which clearly violates (3.33). However, this does not necessarily mean that ICA is useless in the context of functional neuro imaging. Rather, we recall that it is one of several dimensionality reducing coordinate transformations. With this in mind, the use of ICA basis vectors over those yielded by other methods should be warranted by the existence of a subset that provides "informative" projections, as discussed in sections 3.1.2 and 3.1.3. Before assessing if such a subset can be identified from a set of real-world microscopic observations, we shall investigate coordinate transformations of an artificial set of 2D brain-like images with realistic source distributions.

The set of six images depicted in figure 3.14 was constructed for that purpose. Each image contains $40 \times 30 = 1200$ pixels, resulting in the 1200×6 data matrix $\mathbf{S}_b = [\mathbf{s}_1\ \mathbf{s}_2\ \cdots\ \mathbf{s}_6]$. The first source contains super-Gaussian noise within an elliptic shape, chosen to make it

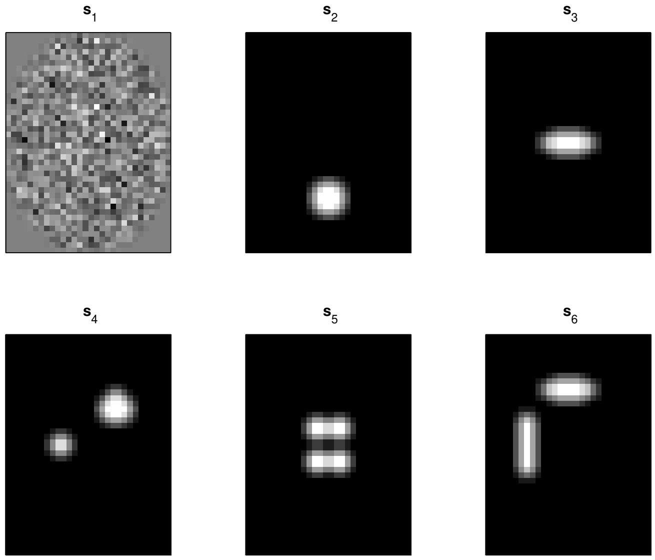

Figure 3.14: The two-dimensional brain-like source signals. The first is super-Gaussian noise within a head-like shape, while the remaining five contain spatially localized patterns with only a few "active" pixels.

resemble the shape of a head sliced horizontally. The remaining five source images contain spatially localized patterns with only a few "active" pixels. These images are intended to represent possible patterns of activity for different neuro-physiological systems. When designing the images the aim was not so much to derive realistic spatial patterns, as to construct realistic source *distributions*. The source signal histograms plotted in figure 3.15 are both skewed and extremely peaked. Only the distribution of the noise image seem close to densities that can be modeled with the generalized logistic function (3.68).

Linear mixtures of the source images (figure 3.16) are generated as $\mathbf{X}_b^\top = \mathbf{A}_b\mathbf{S}_b^\top = [\mathbf{x}_1\ \mathbf{x}_2\ \cdots\ \mathbf{x}_6]^\top$, with the mixing matrix $\mathbf{A}_b$ being constructed with relatively large values in the first column compared to the rest. Therefore the head-shaped noise is relatively dominant in most mixture images. The second column contains the vector $[0\ 1\ 0\ 1\ 0\ 1]^\top$, so that the dot in the second source image appears to be systematically off and on in the mixture images. This is meant to simulate variation induced experimentally by a categorical paradigm.

3.4.2.1 Principal component analysis

After variance normalization the principal axes and corresponding eigenvalues are computed via an SVD of $\mathbf{X}_b$. The relative variances are from (3.18) computed to be $\tilde{l}_1 = 73.47\%$, $\tilde{l}_2 = 11.37\%$, $\tilde{l}_3 = 8.17\%$, $\tilde{l}_4 = 5.04\%$, $\tilde{l}_5 = 1.95\%$. The principal images are depicted in figure 3.17. The head-shaped noise appears to have been somewhat captured by the first principal axis, $\rho_{\mathbf{z}_1,\mathbf{a}_1} = 0.9910$, even though some portions of the second and fourth source signals are visible with the naked eye. However, all the remaining principal axes contain elements of most of the five source "activity" patterns; only the noise component is approximately removed.

The basis vectors provided by PCA are not very informative since the source signals

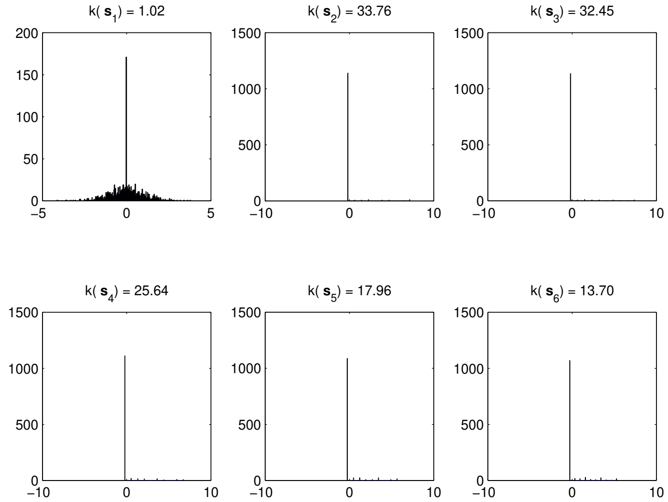

Figure 3.15: The empirical source distributions of the two-dimensional brain-like dataset. All but that of the noise image are both skewed and extremely peaked.

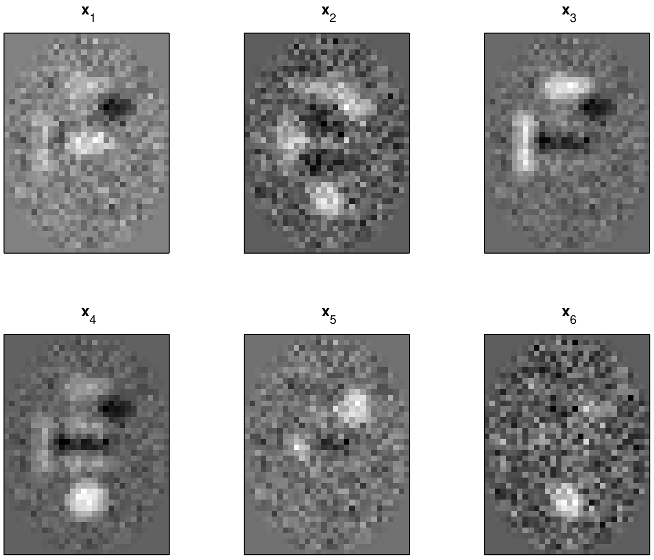

Figure 3.16: Linear mixtures of the images in the two-dimensional brain-like dataset. The (absolute) values of the elements in the first column of the mixing matrix are larger than those in the other columns, meaning that all mixtures have a relatively large component of the head-shaped noise, as is clearly seen.

remain mostly unrecovered. Recall that the second source image is systematically on and off in the mixtures. To see if the categorical paradigm has been captured by any of the principal axes the data matrix is projected onto the PCA basis. Observing the principal components, $\mathbf{z}_i$, as displayed in figure 3.18, it is clear that the systematically induced variation remains undetected. The results indicate that only effects with large variations in

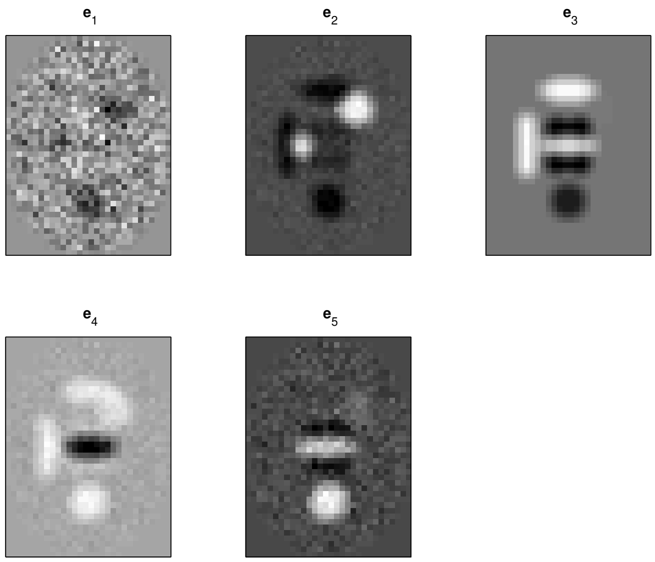

Figure 3.17: Principal axes of the linear mixtures of two-dimensional brain-like sources. The head-shaped noise signal seems to be somewhat captured by the first principal axis, but the remaining axes do not recover any of the sources.

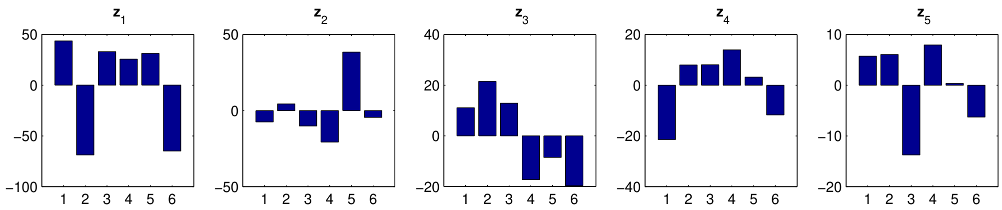

Figure 3.18: The projection of the linear mixtures onto the principal axes for the two-dimensional brain-like dataset. The on-off paradigm is not recovered.

the level of activity are recovered in individual principal axis, in this case the head-shaped noise. Still, the variation accounted for by the last principal axes is so small that they may be ignored all together. In doing so we assume that the information needed to successfully model relevant relationships between the microscopic and macroscopic variables exists in model space as spanned by the first few principal axes; the principal axes may not recover the source signals but do provide a reasonably-sized model space.

3.4.2.2 Independent component analysis

Generalized logistic functions are employed as source density models before application of ICA. As for the sound dataset the model parameters were originally chosen so that the model densities match the empirical distributions: one model with $p = r = 0.7$, which resembles the distribution of the head-shaped noise, and five models with $p = 0.3$ and $r = 1.2$, which are both skewed and very peaked.

Natural gradient ascent progresses as depicted in figure 3.19. Convergence is achieved

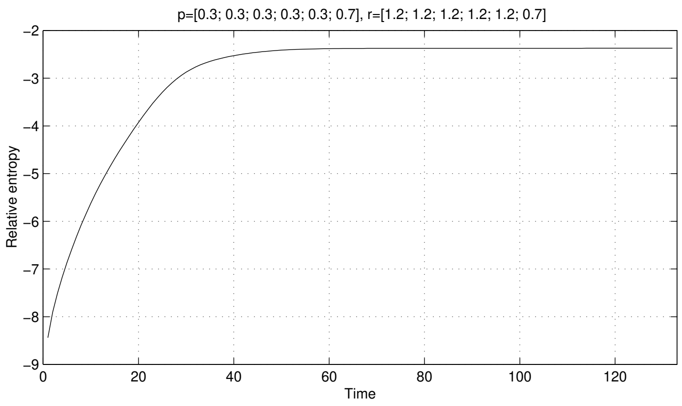

Figure 3.19: Evolution of the relative transformed joint output entropy for the two-dimensional brain-like dataset. Convergence is rapid.

rapidly. The elements of the system matrix in the upper panel of figure 3.20 stops evolving around the same time. Despite a few elements with values different from 0, -1 or $+1$

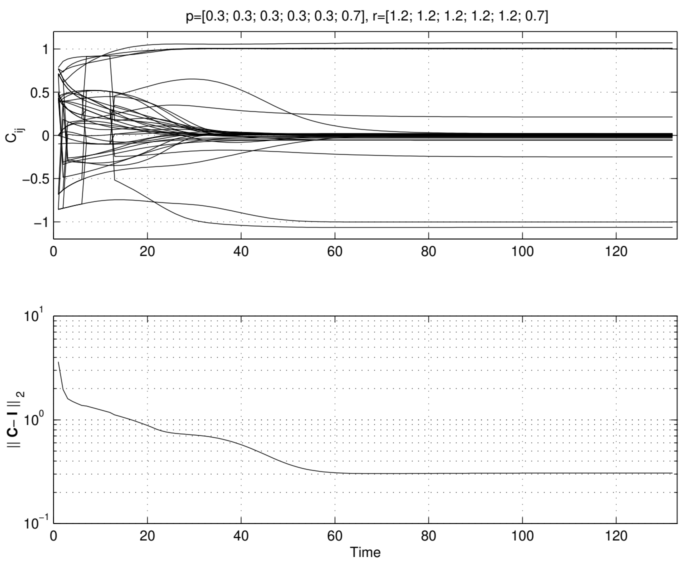

Figure 3.20: Evolution of the system matrix $\mathbf{C} = \mathbf{WA}$ for the two-dimensional brain-like dataset. *Upper panel*: The individual elements C_{ij} of the system matrix. *Lower panel*: The matrix norm $\|\mathbf{C} - \mathbf{I}\|_2$.

the resulting system matrix is almost a quasi-identify matrix, as confirmed by the lower panel of the figure. The degree to which the brain-like image sources have independent distributions has only been quantified by the generalized correlation coefficient $|\mathbf{P_s}| = 0.8186$, but it seems likely that the small non-zero elements of the system matrix can be credited to small dependences between the source images[12]. Still, the independent axes shown in figure 3.21 are very close to the source images. In particular, the fourth estimated

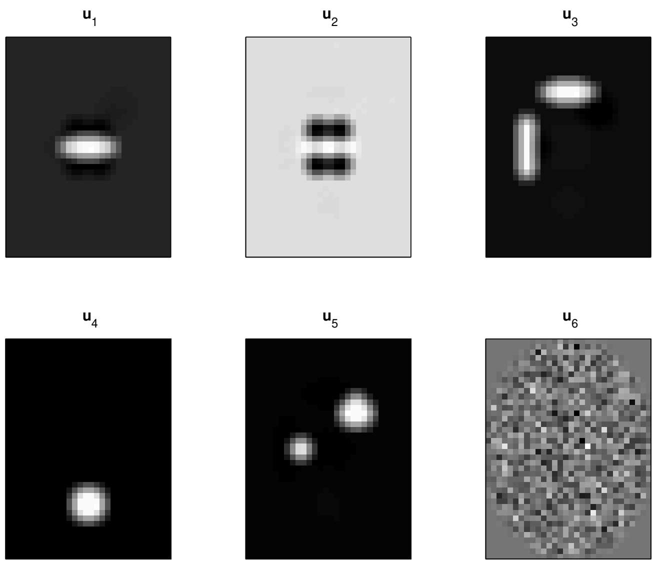

Figure 3.21: Independent axes of the linear mixtures of two-dimensional brain-like sources. Up to a set of scaling factors, all sources have been correctly estimated.

source image greatly resembles the pattern of the categorical paradigm. The projection of the data matrix onto the independent components in figure 3.22 verifies this, as the fourth component correctly discriminates between the mixtures in which the spot is on and those in which it is off. The absolute correlation with the proper mixing vector exceeds 0.97 for

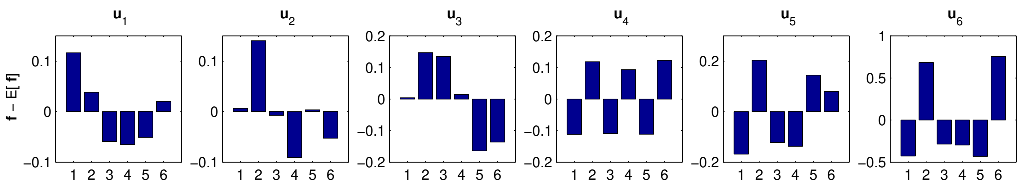

Figure 3.22: The projection of the linear mixtures onto the independent axes for the two-dimensional brain-like dataset. The on-off paradigm is recovered as the fourth independent component.

all independent components.

[12]The source signal estimates are significantly more independent. The generalized correlation coefficient evaluates to $|\mathbf{P_b}| = 0.9901$.

Finally, figures 3.23 and 3.24 reveal that the shape of the source density models is noncritical. Choosing six identical models, all with $p = r = 1$, still results in fast convergence towards the true source images.

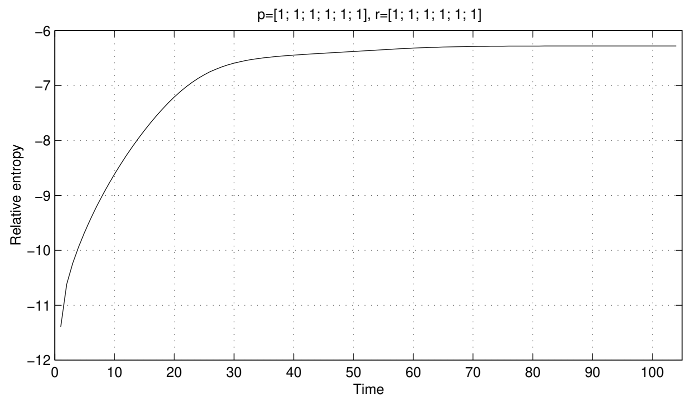

Figure 3.23: Evolution of the relative transformed joint output entropy for the two-dimensional brain-like dataset based on poor source density models. Convergence is still good so a wide range of density models appear to be applicable.

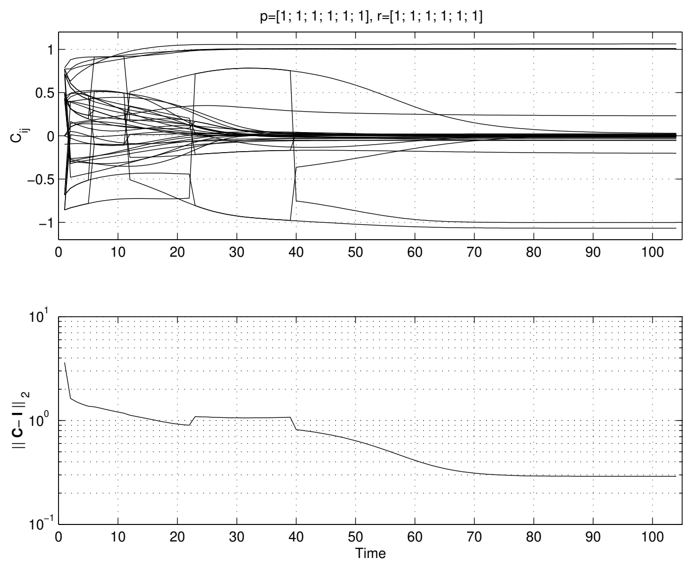

Figure 3.24: Evolution of the system matrix $\mathbf{C} = \mathbf{WA}$ for the two-dimensional brain-like dataset based on poor source density models. The shape of the models is clearly noncritical.

We conclude that both principal and independent component analysis are tools well suited for dimensionality reduction of ill-posed datasets. The variance ranking of PCA allows identification of model space as a subspace of signal space in which most significant information is likely to be present. A similar ranking of the independent basis vectors is not directly available, but we shall investigate other model space identification procedures in the next chapter. However, the individual independent axes seem more informative than their principal counterparts—in fact, they have been demonstrated to closely approximate the original signals when working on linear mixtures of close-to independent sources. When considering the relative success of ICA on the artificial brain-like dataset we must keep in mind, though, the infeasibility of a linear mixture model for human brain function.

3.5 Application to functional neuro imaging data

To demonstrate the reviewed coordinate transformations as well as the modeling and interpretation techniques to be described in the chapters to come, we investigate the PET CPH/SAC dataset described in section A.1.

After standard preprocessing and the application of the common mask to all scans the microscopic data matrix consists of 64 scans, each with 35701 intra-cerebral voxels. We have just seen how projection of the data matrix onto a basis that spans signal space yields loss-free dimensionality reduction. Now we briefly discuss the application of PCA and ICA to the CPH/SAC dataset in order to obtain an efficient representation. Further, the difficulties connected with model space identification based solely on the transformed microscopic variables will become apparent.

3.5.1 Principal component analysis

Singular value decomposition of the centered data matrix yields a set of eigenvalues and corresponding principal axes, as in (3.27). Figure 3.25 reproduces the relative variances accounted for, $\tilde{l}_i$, by all principal axes[13]. The space spanned by the last axes accounts

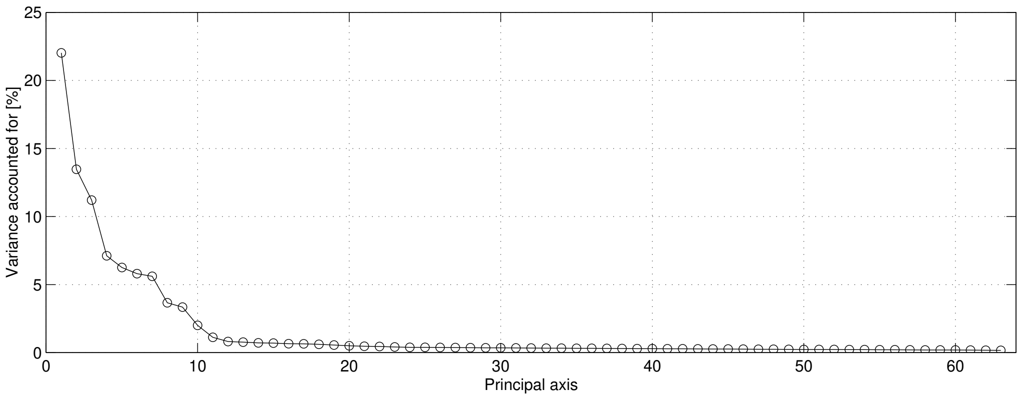

Figure 3.25: Principal component analysis of the CPH/SAC dataset. The plot shows the relative variance accounted for by each of the principal axes.

for very little variance. If the *interesting* variance in the set of microscopic variables, which in this context is the experimentally induced variation, is relatively large compared to variance induced by other sources, it is reasonable to ignore the last principal axes. This effectively identifies model space as the space spanned by the first few principal axes. However, it is not evident that the experimentally induced variance indeed *is* large. By ignoring the principal axes accounting for relatively little variance we therefore risk to

[13] Since we cannot rank the principal axes according to their significance relative to the induced variance, we defer depiction of (selected) principal axes to later.

discard information that is significant when it comes to modeling aspects of the joint micro- and macroscopic probability density.

The model space identification problem relates to the fact that we have no exact knowledge about the true source signals that underly the observed dataset, as we did for the artificially mixed datasets investigated earlier. We are, in other words, unable to identify model space based solely on the microscopic variables. Instead we must utilize information about the experimental design as provided by the macroscopic variables. A simple-minded approach towards this end is to investigate the basis vector projections, i.e. the principal components. These are plotted for the first 12 principal axes in figure 3.26. The dotted vertical lines group the projection values into eight groups, each with scans

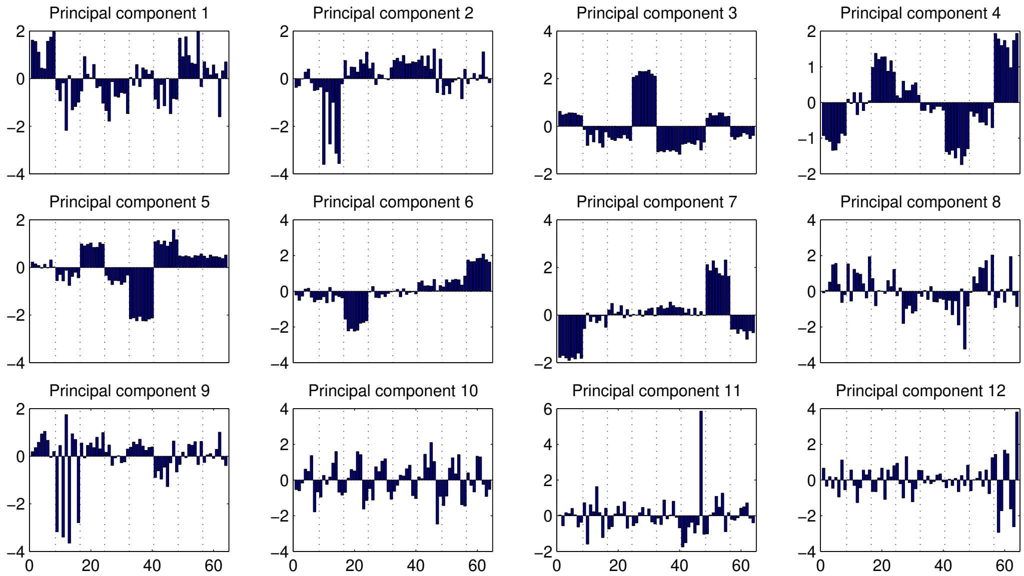

Figure 3.26: The principal components of the first 12 principal axes of the CPH/SAC dataset. The dotted vertical lines group the projection values into eight groups, each with scans from a single subject.

from a single subject. The first seven components appear different from the other five in the figure; strong differences between subjects are evident. These differences will be quantified in section 4.1.2.

3.5.2 Independent component analysis

We recall from section 3.4.2 that the linear mixture model (3.33) cannot be assumed to hold for human brain function. Still, ICA may provide "informative" basis vectors leading to the identification of model space in such a way that variance related to the experimental design is retained.

With reference to the noncritical selection of source density models in the application of ICA to the artificially mixed datasets above we employ identical, generalized logistic density models with $p = r = 1$. Figure 3.27 depicts the evolution of the relative transformed joint output entropy. The issues concerning model space identification from a set of *principal* axes apply to model space identification from a set of *independent* axes as well; we cannot tell "informative" axes from "non-informative" ones without utilizing informa-

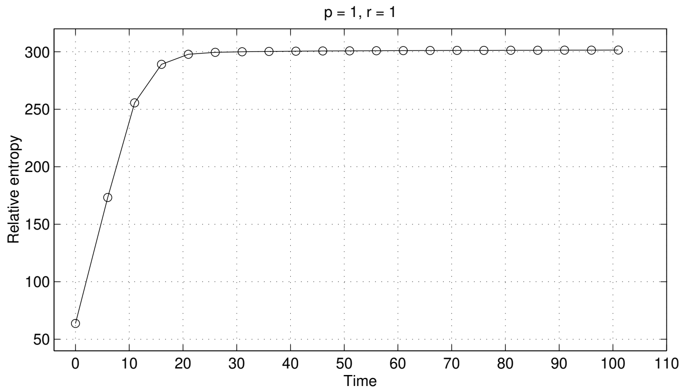

Figure 3.27: Application of ICA to the CPH/SAC dataset. The relative transformed joint output entropy was only evaluated every fifth iteration, resulting in the coarse appearance of the plot. Nevertheless, convergence is achieved after relatively few iterations.

tion about the experimental design. A selection of independent basis vector projections (which we will call independent projections) are depicted in figure 3.28. Structure similar to that of the principal components is not apparent in the limited set shown. However, no equivalent to the ranking of the principal axes based on relative variance exists for ICA, making model space identification even more difficult. Quantitative model space identification together with a more general approach towards modeling of the joint micro- and macroscopic distribution will be presented in the next chapter.

3.6 Summary

In typical functional datasets the dimensionality of each microscopic observation exceeds the number of observations by orders of magnitude. This ill-posed nature holds two major implications for further analysis and modeling; primarily, efficiency can be increased using a basis that spans the same space as the set of all microscopic observations (signal space). Secondly, modeling can benefit from the identification of an even smaller subspace, designed to minimize loss of model-relevant information.

Two basis selection procedures have been reviewed; principal component analysis and independent component analysis. The latter seems to outperform the first when it comes to the identification of informative basis vectors for sets of linearly mixed, independent signals. The extent to which this conclusion holds for sets of microscopic observations of real-world functional datasets is, however, unclear; the unrealistic assumption of a linear mixture model for brain function, coupled with the difficulties related to measuring model-relevance of individual basis vectors based solely on microscopic information, necessitates the inclusion of macroscopic information.

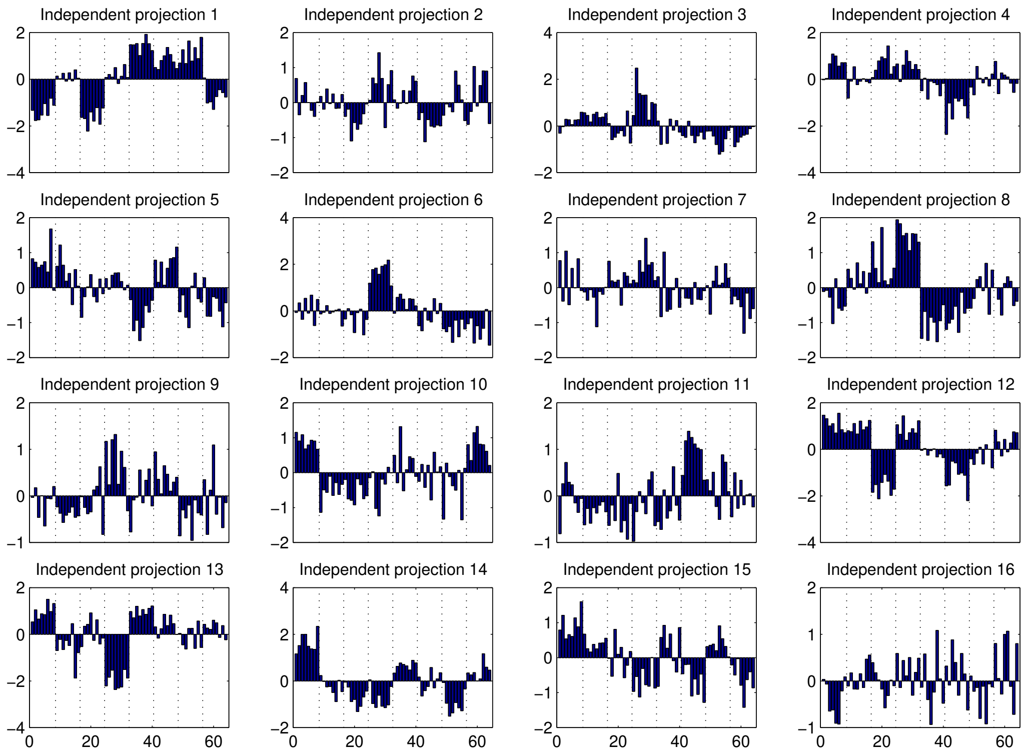

Figure 3.28: The independent projections of 16 independent axes of the CPH/SAC dataset. The dotted vertical lines group the projection values into eight groups, each with scans from a single subject.

Chapter 4

Modeling from signal space

By employing macroscopic information we derive a modeling framework based on generalization theory. The framework helps to highlight an important relation between model performance on one side, and the number of available observations and model flexibility on the other.

4.1 Model space identification

As we saw in the previous chapter a significant dimensionality reduction of the microscopic data representation can be achieved by a coordinate transformation. The reduction is loss-free when the selected basis spans signal space, yielding variables of dimension N instead of d. Such a coordinate transformation does not, however, change the fundamental fact of ill-posed datasets: the observations provide an extremely sparse population of signal space. This sparse sampling of the microscopic probability distribution presents a major problem for techniques based on models of distributional properties of signal space, which is exactly what the analysis and modeling methods we employ are.

In an attempt to remedy the poor sampling of signal space we seek to identify a subspace of signal space which we call *model space* and denote $\mathcal{M}$. Model space must be small (low-dimensional) enough to significantly reduce the sparseness of the microscopic sampling, but at the same time big enough to retain the important aspects of the microscopic probability distribution. The first part of this chapter addresses ways to meet these two conflicting goals.

4.1.1 Model space identification from principal axes

Concentrating first on the basis provided by PCA we recall from (3.17) that it is a set of $N-1$ orthogonal basis vectors ranked according to the amount of variance they account for. If the last $N-k-1$ eigenvalues are small it means that the space spanned by the corresponding eigenvectors accounts for a relatively small portion of the total variance. Ignoring that part of signal space only affects the microscopic sample distribution slightly, so it seems reasonable to identify model space as the space spanned by the first k eigenvectors.

To quantitatively address the issue consider the case of deciding whether or not the $(k+1)$'th principal axis contributes significantly to the variance accounted for by model space. If the variances accounted for by *all* the last $N-k-1$ axes, $\mathbf{e}_{k+1}, \mathbf{e}_{k+2}, \dots, \mathbf{e}_{N-1}$, are identical then the decision to include the $(k+1)$'th axis should lead to the inclusion of all the remaining $N - k - 1$ axes; they all account for an equal amount of variance. This is a case of *isotropic* variance and implies that the space spanned by the first k principal axes, where k is the minimum number for which the remaining $N - k - 1$ principal axes account for an equal amount of variance, is a model space candidate. If the eigenvalue spread of the sample covariance matrix is small, however, the resulting model space may still account for a relatively small portion of the total variance, so the approach is not guaranteed to identify model space in a useful manner.

By assuming the microscopic variables to be multivariate Gaussians, $\mathbf{x}_n \sim N(\mu, \Sigma)$, we find the principal components (transformed variables) to be likewise from (3.19). A likelihood ratio test (LRT) for the hypothesis of the last $\dim(\mathcal{S}) - k$ eigenvalues of the covariance matrix being equal,

$$H_0 : \quad l_{k+1} = l_{k+2} = \cdots = l_{\dim(\mathcal{S})} \quad , \tag{4.1}$$

can then be derived (Mardia et al., 1979, page 235)

$$N\left(\dim(\mathcal{S}) - k\right) \log\left(\frac{a_0}{g_0}\right) \sim \chi^2_{(\dim(\mathcal{S})-k+2)(\dim(\mathcal{S})-k-1)/2} \quad , \tag{4.2}$$

where

$$a_0 = \frac{\sum_{q=k+1}^{\dim(\mathcal{S})} l_q}{\dim(\mathcal{S}) - k} \tag{4.3}$$

$$g_0 = \left[\prod_{q=k+1}^{\dim(\mathcal{S})} l_q\right]^{1/(\dim(\mathcal{S})-k)} \quad . \tag{4.4}$$

Since we are dealing with an LRT the distribution of the test statistic in (4.2) holds asymptotically for $N \to \infty$. The ill-posed nature of functional datasets taken together with the assumption of Gaussianity means we should be careful when attempting to identify model space by employing (4.2).

Figure 4.1 reproduces the eigenvalue spectrum of the CPH/SAC dataset in the top panel, now on a log scale. The lower panel depicts the maximum level for which we can accept the hypothesis (4.1), plotted as a function of k. At a significance level of 5% we cannot reject the hypothesis for $k \leq 5$, meaning that the variance in the space spanned by the last $N - k - 1 = 58$ principal axes can be regarded as isotropic. It follows that the sixth principal axis should not be included in model space without including all remaining axes as well; the space spanned by the first five principal axes

$$\mathcal{M}_{\mathrm{LRT}} = \mathrm{span}\left(\mathbf{e}_i \mid i = 1, \ldots, 5\right) \tag{4.5}$$

candidates for model space. The fraction of variance accounted for by $\mathcal{M}_{\mathrm{LRT}}$ is not very large, however. We find

$$\mathrm{V}\left[\mathcal{M}_{\mathrm{LRT}}\right] = \sum_{i=1}^{5} \tilde{l}_i = 60.1\% \quad , \tag{4.6}$$

which means that while we achieve better sampling of the microscopic probability distribution we ignore some 40% of the total variance, potentially including information that is important in order to satisfactory model the joint micro- and macroscopic density. In

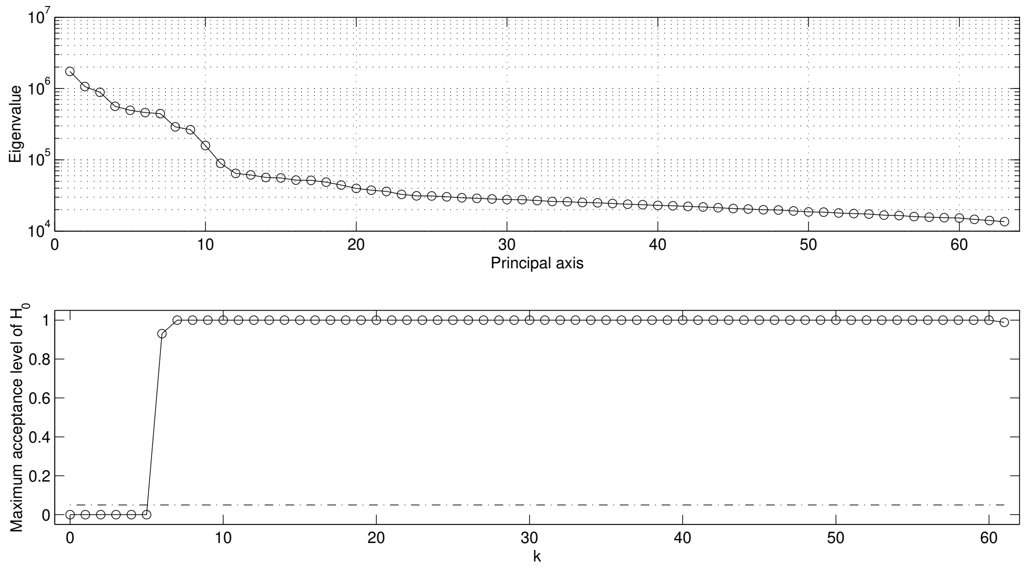

Figure 4.1: Model space identification from the principal axes for the CPH/SAC dataset. *Upper panel*: The eigenvalue spectrum. *Lower panel*: Maximum acceptance level of H_0 : $l_{k+1} = l_{k+2} = \cdots = l_{\dim(\mathcal{S})}$ as a function of k.

fact, it is likely that the experimentally induced saccadic variance is small compared to functional (and potentially remaining anatomical) inter-subject variance.

Generally, model space identification based on eigenvalue LRT's is hampered by the fact that it relies solely on microscopic information. The utilization of information from selected macroscopic variables facilitates a more refined model space identification, as we shall see next.

4.1.2 Analysis of variance

The factors contributing variance to the microscopic variables can be assessed by *variance partitioning*, i.e. the decomposition of the variance into a sum of several factors (Strother et al., 1995a; Lautrup et al., 1994). This is also known as analysis of variance (ANOVA). If we index subjects by t and scans by u, and label the total number of subjects and scans by T and U, respectively, we can decompose the total variance of the i'th principal component

$$\begin{aligned}(N-1)\mathrm{V}[\mathbf{z}_i] &= \sum_{tu}(\mathbf{z}_{i,tu} - \bar{\mathbf{z}}_{i,\cdot\cdot})^2 \\ &= U\sum_{t}(\bar{\mathbf{z}}_{i,t\cdot} - \bar{\mathbf{z}}_{i,\cdot\cdot})^2 + T\sum_{u}(\bar{\mathbf{z}}_{i,\cdot u} - \bar{\mathbf{z}}_{i,\cdot\cdot})^2 \\ &\quad + \sum_{tu}(\mathbf{z}_{i,tu} - \bar{\mathbf{z}}_{i,t\cdot} - \bar{\mathbf{z}}_{i,\cdot u} + \bar{\mathbf{z}}_{i,\cdot\cdot})^2 \quad ,\end{aligned} \tag{4.7}$$

where dot-notation is used for the means

$$\bar{\mathbf{z}}_{i,\cdot\cdot} = \frac{1}{N}\sum_{tu}\mathbf{z}_{i,tu} \qquad \bar{\mathbf{z}}_{i,t\cdot} = \frac{1}{U}\sum_{u}\mathbf{z}_{i,tu} \qquad \bar{\mathbf{z}}_{i,\cdot u} = \frac{1}{T}\sum_{t}\mathbf{z}_{i,tu} \quad . \tag{4.8}$$

The three terms in (4.7) attribute the total variance of $\mathbf{z}_i$ to subject related, scan related, and residual effects. Correspondingly, we label the terms *inter-subject*, *intra-subject*[1], and residual variance.

For the CPH/SAC dataset the application of two-way ANOVA on all principal components yields figure 4.2. Despite the fact that the sources of variance are unlikely to

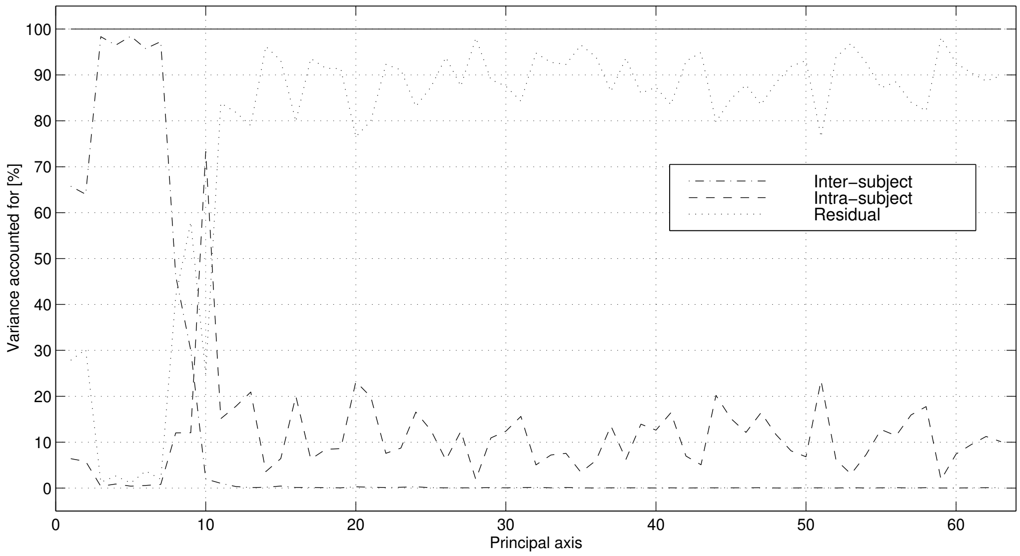

Figure 4.2: Two-way ANOVA of the principal components of the CPH/SAC dataset. The first seven principal axes are completely dominated by inter-subject variance, and only in the tenth component is a significant effect related to saccade frequency variance apparent.

combine in a *linear* fashion, ANOVA may still provide insight into the structure of signal space. In particular, the first seven principal components are almost entirely dominated by inter-subject effects—something that confirms the qualitative impression of the principal components as plotted in figure 3.26. Whether it is due to topographical differences between subjects or shortcomings in the realignment and stereotactic normalization procedures employed as part of the preprocessing, it remains a fact that the most significant saccade-frequency related effect doesn't appear until the tenth component, which accounts for a mere 2% of the total variance. In fact, we are effectively able to identify two orthogonal subspaces: *inter-subject space* spanned by the first seven principal axes which are dominated by inter-subject variance, and *intra-subject space* spanned by the remaining axes[2]. The relatively clear transition that occurs between the $(T-1)$'th and the T'th principal axes turns out to be very characteristic for datasets with T subjects (Strother et al., 1995b)[3].

[1]The term "intra-subject" is used to denote variations *within* subjects. Since the scans of all subjects are consistently organized according to the frequency of the performed saccades, effects related to scan index are identified as saccade-frequency effects.

[2]It is interesting to note that primarily the variance structure of *inter-subject* space changes when advanced nonlinear stereotactic normalization techniques are used; the structure of intra-subject as resolved by ANOVA of the principal components is relatively robust with respect to different stereotactic normalization schemes (Kjems et al., 1997; Kjems, 1998).

[3]Actually, the separation of signal space into spaces dominated by inter- and intra-subject effects is a feature of other orthogonal bases too. In particular, the basis vectors of the scaled subprofile model (SSM),

Figure 4.2 confirms what we suspected all along, namely that the experimentally induced variance of interest, i.e. the variance related to the visual saccades, is small compared to the variance accounting for differences between subjects. This renders model space identification based on eigenvalue LRT's, as described in the previous section, debatable. Instead we may attempt to use two-way ANOVA to identify model space. First, however, let us examine its application to the independent components of the CPH/SAC dataset, as depicted in figure 4.3. In contrast to the principal axes no variance related

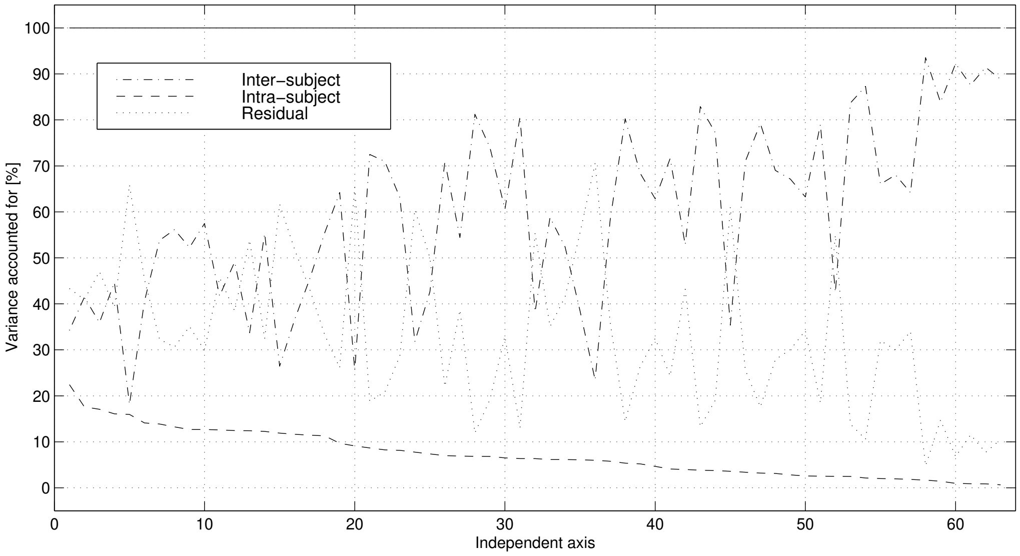

Figure 4.3: Two-way ANOVA of the independent components of the CPH/SAC dataset ranked according to the amount of variance that can be attributed to intra-subject effects. No single component is dominated by intra-subjects effects.

ranking is available for the independent axes. Instead, the components have been ranked according to the amount of variance that can be accounted to intra-subject effects, i.e. saccade frequency variance. While the variance in a number of components appear to be somewhat attributable to this factor no single component isolates the effect as clearly as the tenth principal component in figure 4.2. The immediate interpretation is that ICA provides basis vectors that are less informative than those provided by PCA—however, it may be a bit more involved: the observed microscopic variables measure the combined neuronal activity of many neuro-physiological systems. It is possible, perhaps even likely, that of those systems involved in the task of performing visual saccades activity levels may relate nonlinearly to the frequency with which the saccades are performed. For example, the activity of a neuro-physiological system may exhibit an almost linear relationship with the saccade frequency for small frequencies, but "saturate" when the frequency increases above a certain threshold. This kind of behavior would diminish ANOVA's ability to identify a single intra-subject component related to the saccade frequency, since ANOVA is based on linear decomposition of the variance.

It is hard to determine what causes the differences in the variance decomposition patterns for PCA and ICA, but the very existence of the differences indicates the need

which attempts to account for multiplicative scan effects (Moeller et al., 1987; Moeller and Strother, 1991), often yields an inter-subject space more completely dominated by inter-subject effects than does PCA.

for more sophisticated model space identification methods. In fact, since we by ANOVA attempt to identify one of the basis vectors as the one-dimensional linear subspace that reflects variance related to one particular macroscopic variable, ANOVA constitutes a *model* portraying properties of the joint micro- and macroscopic distribution. As outlined earlier, proper model space identification may indeed depend on the type of model we employ, be it simple as the linear ANOVA model or more complex. We therefore turn to discuss model performance measures.

4.2 Quantifying model performance

Recall from chapter 2 that the system functional neuro imaging aims to investigate is governed by the joint micro- and macroscopic probability distribution $\mathrm{p}(\mathbf{x}, \mathbf{g})$[4]. To estimate or identify the system we employ a model, as in (2.2)

$$\hat{\mathrm{p}}(\mathbf{x}, \mathbf{g}) = \mathrm{p}(\mathbf{x}, \mathbf{g}|\mathbf{w}) \quad , \tag{4.9}$$

where $\mathbf{w}$ is a vector of model parameters. We aim to identify the set of parameters $\mathbf{w}^*$ such that the model density approximates the system

$$\mathrm{p}(\mathbf{x}, \mathbf{g}|\mathbf{w}^*) = \mathrm{p}(\mathbf{x}, \mathbf{g}) \quad . \tag{4.10}$$

We call $\mathbf{w}^*$ the set of true parameters. Next we shall examine quantitative measures of this model-to-system approximation.

While most of what follows is relatively straightforward and results similar to those we derive here exist in the literature, the application of generalization measures is novel in the context of functional neuro modeling. Specifically, the relations between model complexity on model performance have been left largely unaddressed until now.

4.2.1 Maximum a posteriori estimation

For the unknown system $\mathrm{p}(\mathbf{x}, \mathbf{g})$ we are left to identify a proper set of parameters from a dataset of observations of $\mathbf{x}$ and $\mathbf{g}$ drawn simultaneous from the joint distribution $\mathrm{p}(\mathbf{x}, \mathbf{g})$. We call a dataset used in this way a *training set* and label it D

$$\mathsf{D} = \{(\mathbf{x}_n, \mathbf{g}_n) \mid n = 1, \ldots, N\} \quad . \tag{4.11}$$

Consider the joint distribution of model parameters and the training set, which we can rewrite using Bayes theorem (Duda and Hart, 1973)

$$\mathrm{p}(\mathbf{w}, \mathsf{D}) = \mathrm{p}(\mathbf{w}|\mathsf{D})\mathrm{p}(\mathsf{D}) = \mathrm{p}(\mathsf{D}|\mathbf{w})\mathrm{p}(\mathbf{w}) \tag{4.12}$$

$$\Updownarrow$$

$$\mathrm{p}(\mathbf{w}|\mathsf{D}) = \frac{\mathrm{p}(\mathsf{D}|\mathbf{w})\mathrm{p}(\mathbf{w})}{\mathrm{p}(\mathsf{D})} \quad , \tag{4.13}$$

where the normalization factor

$$\mathrm{p}(\mathsf{D}) = \int \mathrm{p}(\mathbf{w})\mathrm{p}(\mathsf{D}|\mathbf{w})\, d\mathbf{w} \tag{4.14}$$

ensures that $\int \mathrm{p}(\mathbf{w}|\mathsf{D})\, d\mathbf{w} = 1$. In (4.13) we identify

[4] The microscopic variables may be represented using any basis that spans signal space, so when we write $\mathbf{x}$ in the following it may be substituted for other representations, such as the principal components $\mathbf{z}$.

The prior $\mathrm{p}(\mathbf{w})$ as the marginal distribution of model parameters that we assume before observing the data in the training set D.

The likelihood $\mathrm{p}(\mathsf{D}|\mathbf{w})$ as the conditional distribution of data for a specific set of model parameters. It quantifies the probability of observing different training sets for a given set of model parameters.

The posterior $\mathrm{p}(\mathbf{w}|\mathsf{D})$ as the conditional distribution of model parameters for the specific training set, i.e. the probability of different sets of model parameters for the observed training set.

Now, a model as in (4.9), $\hat{\mathrm{p}}(\mathbf{x}, \mathbf{g}) = \mathrm{p}(\mathbf{x}, \mathbf{g}|\mathbf{w})$, with parameters estimated from the training set D will depend on D through the parameters. By integration over the parameters we obtain

$$\begin{aligned}
\mathrm{p}(\mathbf{x}, \mathbf{g}|\mathsf{D}) &= \int \mathrm{p}(\mathbf{x}, \mathbf{g}, \mathbf{w}|\mathsf{D})\ \mathrm{d}\mathbf{w} &(4.15)\\
&= \int \mathrm{p}(\mathbf{x}, \mathbf{g}|\mathbf{w}, \mathsf{D})\mathrm{p}(\mathbf{w}|\mathsf{D})\ \mathrm{d}\mathbf{w} &(4.16)\\
&= \int \mathrm{p}(\mathbf{x}, \mathbf{g}|\mathbf{w})\mathrm{p}(\mathbf{w}|\mathsf{D})\ \mathrm{d}\mathbf{w} \quad , &(4.17)
\end{aligned}$$

where the third equality holds because the model density is independent of D once the parameters $\mathbf{w}$ have been set. The approach in (4.17) is called *Bayesian inference* and approximates the joint distribution of $\mathbf{x}$ and $\mathbf{g}$ from the training set as a weighted average over all parameters (Bishop, 1995, chapter 10), (Mardia et al., 1979).

If the posterior distribution is relatively sharply peaked around $\mathbf{w}^*$ then this value will dominate the integral in (4.17), so we get

$$\begin{aligned}
\mathrm{p}(\mathbf{x}, \mathbf{g}|\mathsf{D}) &\simeq \mathrm{p}(\mathbf{x}, \mathbf{g}|\mathbf{w}^*) \int \mathrm{p}(\mathbf{w}|\mathsf{D})\ \mathrm{d}\mathbf{w} &(4.18)\\
&= \mathrm{p}(\mathbf{x}, \mathbf{g}|\mathbf{w}^*) \quad , &(4.19)
\end{aligned}$$

using the normalization condition $\int \mathrm{p}(\mathbf{w}|\mathsf{D})\ \mathrm{d}\mathbf{w} = 1$. We have in effect substituted the weighted posterior average by its maximum value, meaning that the system is modeled by choosing the parameters $\mathbf{w}^*$ that maximize the posterior probability $\mathrm{p}(\mathbf{w}|\mathsf{D})$

$$\mathbf{w}^*_{\mathrm{MAP}} = \arg\max_{\mathbf{w}} \left[\mathrm{p}(\mathbf{w}|\mathsf{D})\right] \quad . \tag{4.20}$$

The principle is known as maximum a posteriori (MAP) estimation.

4.2.2 Maximum likelihood and cost functions

If the training set tuples $(\mathbf{x}_n, \mathbf{g}_n)$ are drawn independently we can factor out the likelihood

$$\mathrm{p}(\mathsf{D}|\mathbf{w}) = \prod_{n=1}^{N} \mathrm{p}(\mathbf{x}_n, \mathbf{g}_n|\mathbf{w}) \quad . \tag{4.21}$$

Similarly, the prior distribution of independent parameters becomes

$$\mathrm{p}(\mathbf{w}) = \prod_{v=1}^{W} \mathrm{p}(w_v) \quad , \tag{4.22}$$

where W is the total number of parameters. By observing that a monotonic (one-to-one) transformation leaves the maximum unchanged we can employ a log-transformation of the posterior distribution

$$\mathbf{w}^*_{\text{MAP}} = \arg\max_{\mathbf{w}} [\mathrm{p}(\mathbf{w}|\mathsf{D})] \tag{4.23}$$

$$= \arg\max_{\mathbf{w}} \left[\frac{1}{N} \frac{\mathrm{p}(\mathsf{D}|\mathbf{w})\mathrm{p}(\mathbf{w})}{\mathrm{p}(\mathsf{D})} \right] \tag{4.24}$$

$$= \arg\max_{\mathbf{w}} \left[\frac{1}{N} \prod_{n=1}^{N} \mathrm{p}(\mathbf{x}_n, \mathbf{g}_n|\mathbf{w}) \prod_{v=1}^{W} \mathrm{p}(w_v) \right] \tag{4.25}$$

$$= \arg\min_{\mathbf{w}} \left[-\frac{1}{N} \sum_{n=1}^{N} \log \mathrm{p}(\mathbf{x}_n, \mathbf{g}_n|\mathbf{w}) - \frac{1}{N} \sum_{v=1}^{W} \log \mathrm{p}(w_v) \right] \tag{4.26}$$

to achieve an additive rather than a multiplicative measure. Note that the posterior has been normalized by N which shall prove beneficial latter.

With no prior knowledge of the parameter distribution we can assume all parameters to be equally likely, i.e. a uniform prior $\mathrm{p}(\mathbf{w})$. Maximum a posteriori estimation then reduces to maximum likelihood (ML) estimation

$$\mathbf{w}^*_{\text{ML}} = \arg\max_{\mathbf{w}} [\mathrm{p}(\mathsf{D}|\mathbf{w})] \tag{4.27}$$

$$= \arg\min_{\mathbf{w}} \left[-\frac{1}{N} \sum_{n=1}^{N} \log \mathrm{p}(\mathbf{x}_n, \mathbf{g}_n|\mathbf{w}) \right] \tag{4.28}$$

$$= \arg\min_{\mathbf{w}} \left[-\frac{1}{N} \sum_{n=1}^{N} e(\mathbf{x}_n, \mathbf{g}_n, \mathbf{w}) \right] \tag{4.29}$$

$$= \arg\min_{\mathbf{w}} [\mathrm{E}(\mathsf{D}, \mathbf{w})] \quad , \tag{4.30}$$

which identifies the model parameters as those that maximize the probability of the observed training set. Maximum likelihood measures of model performance, $\mathrm{E}(\mathsf{D}, \mathbf{w})$, are often called *error functions* or *cost functions*; they measure the sum of errors, i.e. the log-likelihood $e(\mathbf{x}_n, \mathbf{g}_n, \mathbf{w}) = \log \mathrm{p}(\mathbf{x}_n, \mathbf{g}_n|\mathbf{w})$, for independent observations. Next we shall investigate one particular such cost function.

4.2.3 Mean square error

In chapter 2 we discussed the identification of system signals. In particular, we outlined a system in which task-describing macroscopic variables, e.g. the nominal saccade frequency, were considered as inputs and the microscopic patterns of neuronal activity as outputs. However, we could also consider the saccade frequency performance measure as system output[5]. Consequently, we would regard the microscopic variables as system inputs. The renewed signal identification in effect corresponds to a shift of the system boundaries, as discussed in section 2.1.1. In chapter 5 we shall investigate the relationship between the two approaches for linear models.

For now we consider the system that governs the performance of visual saccades. The saccades, in particular the frequency with which they are performed, are the result of the

[5]As described in appendix A the frequency of the performed saccades differs only insignificantly from the frequency of the flashing LED's. Therefore the latter is used instead of the actual saccade frequency.

microscopic patterns of neuronal activity. Therefore, a convenient decomposition of the the joint input-output density is

$$\mathrm{p}(\mathbf{x},\mathbf{g}) = \mathrm{p}(\mathbf{g}|\mathbf{x})\mathrm{p}(\mathbf{x}) \quad , \tag{4.31}$$

in which we identify the conditional density $\mathrm{p}(\mathbf{g}|\mathbf{x})$ as the system that governs the macroscopic behavior $\mathbf{g}$ for a given set of microscopic variables $\mathbf{x}$. The marginal distribution of $\mathbf{x}$ plays an important role in the modeling process, specifically in signal- and model space identification as we saw in the previous chapter. However, for the purpose of relating saccade frequency to the patterns of neuronal activity it is the conditional density $\mathrm{p}(\mathbf{g}|\mathbf{x})$ we aim to model. To this end we employ $\mathbf{y}(\mathbf{x},\mathbf{w}) = \hat{\mathrm{p}}(\mathbf{g}|\mathbf{x}) = \mathrm{p}(\mathbf{g}|\mathbf{x},\mathbf{w})$ in line with (4.9).

Assume that the K elements of the macroscopic vector are independent

$$\mathrm{p}(\mathbf{g}|\mathbf{x}) = \prod_{k=1}^{K} \mathrm{p}(g_k|\mathbf{x}) \quad . \tag{4.32}$$

Further, assume that the macroscopic variables g_k are given by a deterministic function of $\mathbf{x}$ with added Gaussian noise

$$g_k = h_k(\mathbf{x}) + e_k \quad , \qquad e_k \sim N(0,\sigma^2) \quad , \tag{4.33}$$

meaning that the macroscopic variables themselves are Gaussians. Since we employ $y_k(\mathbf{x},\mathbf{w})$ as our model of $h_k(\mathbf{x})$ assumption (4.33) yields

$$y_k(\mathbf{x},\mathbf{w}) \simeq h_k(\mathbf{x}) = g_k - e_k \quad . \tag{4.34}$$

It follows that $\mathrm{p}(g_k|\mathbf{x},\mathbf{w}) = y_k(\mathbf{x},\mathbf{w}) \sim N(\langle h_k(\mathbf{x})\rangle, \mathrm{V}\left[h_k(\mathbf{x})\right]) = N(g_k,\sigma^2)$, i.e.

$$\mathrm{p}(g_k|\mathbf{x},\mathbf{w}) = \frac{1}{\sqrt{(2\pi\sigma^2)}} \exp\left[-\frac{(y_k(\mathbf{x},\mathbf{w}) - g_k)^2}{2\sigma^2}\right] \quad , \tag{4.35}$$

which, by using (4.28) and (4.32), leads to the ML estimator

$$\begin{aligned}
\mathbf{w}^*_{\mathrm{ML}} &= \arg\min_{\mathbf{w}} \left[-\frac{1}{N}\sum_{n=1}^{N} \log \mathrm{p}(\mathbf{g}_n|\mathbf{x}_n,\mathbf{w})\right] && (4.36)\\
&= \arg\min_{\mathbf{w}} \left[-\frac{1}{N}\sum_{n=1}^{N}\sum_{k=1}^{K} \log \mathrm{p}(g_{n,k}|\mathbf{x}_n,\mathbf{w})\right] && (4.37)\\
&= \arg\min_{\mathbf{w}} \left[\frac{1}{2N\sigma^2}\sum_{n=1}^{N}\sum_{k=1}^{K} (y_k(\mathbf{x}_n,\mathbf{w}) - g_{n,k})^2 + K\log\sigma + \frac{K}{2}\log(2\pi)\right] && (4.38)\\
&= \arg\min_{\mathbf{w}} \left[\frac{1}{2N}\sum_{n=1}^{N}\sum_{k=1}^{K} (y_k(\mathbf{x}_n,\mathbf{w}) - g_{n,k})^2\right] \quad , && (4.39)
\end{aligned}$$

where the second and third terms in the third line vanish because they are independent of $\mathbf{w}$. The same goes for the overall factor $1/\sigma^2$, resulting in the so-called *mean square error* (MSE) (Duda and Hart, 1973; Mardia et al., 1979). To reiterate, (4.39) is the result of ML estimation of the model parameters when assuming the macroscopic variables to be Gaussian. The MSE cost function is often used, even when the Gaussian assumption is unwarranted.

For the CPH/SAC dataset the saccade frequency is the macroscopic variable of interest, so we set $K = 1$. This yields the MSE cost function

$$\mathrm{E}_{\mathrm{MSE}}\left(\mathsf{D}, \mathbf{w}\right) = \frac{1}{2N} \sum_{n=1}^{N} \left(y(\mathbf{x}_n, \mathbf{w}) - g_n\right)^2 \quad , \tag{4.40}$$

where the individual error (log-likelihood) terms are

$$e(\mathbf{x}_n, g_n, \mathbf{w}) = \log \mathrm{p}(g_n|\mathbf{x}_n, \mathbf{w}) = \frac{1}{2}\left(y(\mathbf{x}_n, \mathbf{w}) - g_n\right)^2 \quad . \tag{4.41}$$

4.2.4 Gaussian prior

To constrain model complexity and ensure numerical stability when estimating model parameters we can often benefit from assuming a Gaussian parameter prior. The reason for this become clear in section 4.4.1. So, assume that the parameter distribution is Gaussian $\mathrm{p}(w_v) \sim N(0, \rho_v)$, meaning that the individual parameters are independent with p.d.f.

$$\mathrm{p}(w_v) = \frac{1}{\sqrt{(2\pi\rho_v^2)}} \exp\left[-\frac{w_v^2}{2\rho_v^2}\right] \quad . \tag{4.42}$$

Inserting this into the MAP parameter estimate of (4.26) we find

$$\begin{aligned}
\mathbf{w}^*_{\mathrm{MAP}} &= \arg\min_{\mathbf{w}} \left[-\frac{1}{N}\sum_{n=1}^{N} \log \mathrm{p}(\mathbf{x}_n, \mathbf{g}_n|\mathbf{w}) - \frac{1}{N}\sum_{v=1}^{W} \log \mathrm{p}(w_v)\right] && (4.43)\\
&= \arg\min_{\mathbf{w}} \left[\mathrm{E}(\mathsf{D}, \mathbf{w}) + \frac{1}{N}\sum_{v=1}^{W} \frac{w_v^2}{2\rho_v^2}\right] \quad . && (4.44)
\end{aligned}$$

The augmented cost function is called the *regularized cost function*

$$\begin{aligned}
\mathsf{C}(\mathsf{D}, \mathbf{w}) &= \mathrm{E}(\mathsf{D}, \mathbf{w}) + \frac{1}{N}\sum_{v=1}^{W} \frac{w_v^2}{2\rho_v^2} && (4.45)\\
&= \mathrm{E}(\mathsf{D}, \mathbf{w}) + \frac{1}{N}\mathrm{R}(\mathbf{w}) \quad , && (4.46)
\end{aligned}$$

where the regularization term is

$$\mathrm{R}(\mathbf{w}) = \sum_{v=1}^{W} \frac{w_v^2}{2\rho_v^2} = \sum_{v=1}^{W} \frac{\alpha_v}{2} w_v^2 = \frac{1}{2}\mathbf{w}^{\mathsf{T}}\mathbf{R}\mathbf{w} \quad . \tag{4.47}$$

The α_v's in (4.47) are inversely proportional to the prior variances, and are as such non-negative. Arranging them in a diagonal matrix $\mathbf{R} = \mathrm{diag}\left[(\alpha_v)\right]$ the regularization term is conveniently expressed as a quadratic form. As we shall see in section 5.2.1 the regularized cost function is closely linked to Ridge-regression (Hoerl and Kennard, 1970).

4.3 Generalization

In the limit in which the number of training set observations goes to infinity the summation in $\mathrm{E}(\mathsf{D}, \mathbf{w})$ can be replaced with an integral over the joint micro- and macroscopic distribution

$$\begin{aligned} \lim_{N\to\infty} \mathrm{E}(\mathsf{D}, \mathbf{w}) &= \lim_{N\to\infty} \frac{1}{N}\sum_{n=1}^{N} e(\mathbf{x}_n, g_n, \mathbf{w}) && (4.48)\\ &= \iint e(\mathbf{x}, g, \mathbf{w})\mathrm{p}(g, \mathbf{x})\ \mathrm{d}g\mathrm{d}\mathbf{x} && (4.49)\\ &= \iint e(\mathbf{x}, g, \mathbf{w})\mathrm{p}(g|\mathbf{x})\mathrm{p}(\mathbf{x})\ \mathrm{d}g\mathrm{d}\mathbf{x} \quad . && (4.50)\end{aligned}$$

We define the *generalization error* G as the cost function value in this limit (Larsen, 1994)

$$\mathrm{G}(\mathsf{D}, \mathbf{w}) \stackrel{\text{def}}{=} \lim_{N\to\infty} \mathrm{E}(\mathsf{D}, \mathbf{w}) = \iint e(\mathbf{x}, g, \mathbf{w})\mathrm{p}(g|\mathbf{x})\mathrm{p}(\mathbf{x})\ \mathrm{d}g\mathrm{d}\mathbf{x} \quad , \tag{4.51}$$

i.e. the expectation of the cost function with respect to the joint distribution of inputs and outputs. Generalization error is, in other words, the average error over the true joint input-output distribution as measured by the chosen cost function. We note how generalization error depends on the model parameters and through them also on the training set[6]. The definition of generalization error applies to all cost functions, even though we focus on the MSE cost function here.

Ideally, model performance should be assessed by measuring generalization error. In practice, however, generalization error cannot be computed, being the result of a limiting process. It is the integration over the true, but unknown density $\mathrm{p}(g, \mathbf{x})$ that causes the problems. To assess model performance we must substitute $\mathrm{p}(g, \mathbf{x})$ by an empirical estimate. This is in fact the opposite of the limiting process in (4.48), and we have already seen that the empirical training set density

$$\mathrm{p}_{\mathsf{D}}(g, \mathbf{x}) = \frac{1}{N}\sum_{n=1}^{N} \delta(g - g_n, \mathbf{x} - \mathbf{x}_n) \quad , \qquad (g_n, \mathbf{x}_n) \in \mathsf{D} \tag{4.52}$$

results in the summation in (4.40) in the case of mean square error. For this reason the mean error of the training set is denoted *training error*. Before addressing the issue of generalization error estimates further we will turn our attention to the training set dependency of the generalization error as defined in (4.51).

4.3.1 Expected generalization error

Generalization error measures model performance. More specifically it measures performance of the model with the specific set of parameters $\mathbf{w}$. When these have been estimated based on training error, generalization error implicitly depends on the training set. To eliminate this dependency we define *expected generalization error* as the average generalization error over training sets

$$\bar{\mathrm{G}} \stackrel{\text{def}}{=} \langle \mathrm{G}(\mathsf{D}, \mathbf{w}) \rangle_{\mathrm{p}(\mathsf{D}(N))} = \int \mathrm{G}(\mathsf{D}, \mathbf{w})\mathrm{p}(\mathsf{D}(N))\ \mathrm{d}\mathsf{D}(N) \quad , \tag{4.53}$$

[6] Actually, generalization error depends on the training set only because the parameters $\mathbf{w}$ in (4.51) are *estimates* of the set of "true" parameters $\mathbf{w}^*$ which correctly model $h(\mathbf{x})$. More rigorously, we would write $\hat{\mathbf{w}} = \mathbf{w}(\mathsf{D})$, causing generalization error based on such models to be labelled $\mathrm{G}(\mathbf{w}(\mathsf{D}))$.

where $\mathsf{D}(N)$ denotes training sets of size N.

4.3.2 Empirical estimates

Generalization error can be estimated using an empirical density estimate. The empirical training set density in (4.52) yields training error, as we have already seen. Training error is, however, a *biased* estimate of generalization error, since the latter depends on the training set via the estimated parameters $\mathbf{w}$. Analytical properties of the training error bias are investigated in detail in the next section. However, to obtain an unbiased *empirical* estimate we can use a *test set* with observations independent of those in the training set (but still drawn from the true distribution $\mathrm{p}(g, \mathbf{x})$) (Larsen, 1994; Larsen and Hansen, 1995a)

$$\mathsf{T} = \{(\mathbf{x}_n, \mathbf{g}_n) \mid n = 1, \ldots, N_\mathsf{T}\} \tag{4.54}$$

with the empirical density

$$\mathrm{p}_\mathsf{T}(g, \mathbf{x}) = \frac{1}{N_\mathsf{T}} \sum_{n=1}^{N_\mathsf{T}} \delta(g - g_n, \mathbf{x} - \mathbf{x}_n) \quad , \qquad (g_n, \mathbf{x}_n) \in \mathsf{T} \quad , \tag{4.55}$$

resulting in the so-called *test error*

$$\hat{\mathrm{G}}_{\mathsf{T}(\mathrm{N}_\mathsf{T})}(\mathsf{D}, \mathbf{w}) = \frac{1}{N_\mathsf{T}} \sum_{n=1}^{N_\mathsf{T}} e(\mathbf{x}_n, g_n, \mathbf{w}) \quad , \qquad (g_n, \mathbf{x}_n) \in \mathsf{T} \quad . \tag{4.56}$$

The subscript $\mathsf{T}(N_\mathsf{T})$ indicates that the estimate is based on a test set with N_T observations. An empirical estimate of expected generalization error is obtained as the average test error for training sets of size N.

4.3.2.1 Cross-validation

In practical applications the total number of available observations, label it N_{tot}, is limited. This presents a trade-off, since the N_T observations used for evaluating the test error reduces the size of the training set, $N = N_{\text{tot}} - N_\mathsf{T}$. Now, test error corresponds to generalization error in the limit of an infinitely large test set

$$\lim_{N_\mathsf{T} \to \infty} \hat{\mathrm{G}}_{\mathsf{T}(\mathrm{N}_\mathsf{T})}(\mathsf{D}, \mathbf{w}) = \mathrm{G}(\mathsf{D}, \mathbf{w}) \quad , \tag{4.57}$$

so we should opt for a test set as large as possible. However, increasing N_T reduces N resulting in models with poorer generalization abilities[7], so we face a dilemma.

One way to limit the number of observations "lost" to the test set is to employ leave-N-out *cross-validation*[8] (CV), see e.g. (Stone, 1974; Toussaint, 1974; Efron, 1983; Larsen and Hansen, 1995b; Hansen and Larsen, 1996). The idea is to successively leave N_T samples of the training set out for test error evaluation, using the remaining $N = N_{\text{tot}} - N_\mathsf{T}$ for training. For $N_\mathsf{T} = 1$, which is called leave-one-out CV, the training- and test sets can be chosen in N different ways, each time yielding a test error estimate $\hat{\mathrm{G}}(\mathsf{D}, \mathbf{w}_{(j)})$, $j =$

[7] As we shall see in section 4.3.6 generalization error increases with decreasing N.

[8] With the notation used here it is, in fact, more appropriate to label the technique leave-N_T-out cross-validation.

$1, \ldots, N$. Here $\mathbf{w}_{(j)}$ denotes the parameters estimated from the training set segment with the j'th observation left out. The average CV generalization error

$$\hat{\mathsf{G}}_{\text{CV}}(\mathsf{D}) = \frac{1}{N} \sum_{j=1}^{N} \hat{\mathsf{G}}(\mathsf{D}, \mathbf{w}_{(j)}) \tag{4.58}$$

then provides an unbiased estimate of expected generalization error.

4.3.2.2 Bootstrap methods

For CV all observations in each training set segment are unique, which means that training sets no larger than $N = N_{\text{tot}} - N_{\mathsf{T}}$ can be generated. An alternative approach is to sample training sets *with replacement*, so that each observation may occur more than once in the same training set segment. This technique is known as bootstrapping (BS) and has one important property; individual generalization error estimates $\hat{\mathsf{G}}(\mathsf{D}, \mathbf{w}^{(j)})$ based on BS parameter estimates $\mathbf{w}^{(j)}$, $j = 1, \ldots, M$ facilitates BS estimates, such as the average

$$\hat{\mathsf{G}}_{\text{BS}}(\mathsf{D}) = \frac{1}{M} \sum_{j=1}^{M} \hat{\mathsf{G}}(\mathsf{D}, \mathbf{w}^{(j)}) \quad , \tag{4.59}$$

that are *asymptotically central* (Efron, 1982; Efron and Tibshirani, 1993; Young, 1994). Sampling with replacement does not, in other words, introduce bias when estimating properties of the distribution of measures based on the samples. This can be use to generate a large number M of bootstrapped training set samples which in turn facilitates estimates of things like the average and standard deviation of the test error. This will come in handy for the practical applications in chapters 5 and 6.

4.3.3 Algebraic estimates

The quality of the parameter estimates depends on the number of observations in the training set; more observations provide better estimates, so we should avoid reducing the training set by holding observations out for a test set. While the situation can be somewhat remedied using CV or BS this involves the estimation of a large number of models. An alternative approach is to eliminate the observation-consuming test set all together, reverting to algebraic generalization error estimates based on model complexity considerations.

An algebraic generalization error estimate is exactly what we derive in appendix C. While some of the steps may seem somewhat involved the result is well worth the effort. For a general discussion see (Ljung, 1987). Under a few assumptions, namely

- The set of true parameters $\mathbf{w}^*$ falls within the set of relationships that the parameterized model can implement (see also the discussion in section 2.2).

- Noise is additive and independent between observations, and has zero mean.

- The number of observations is large. (This is not the case for typical functional datasets.)

and writing D for $\mathsf{D}(N)$ and $\langle \mathrm{G}\rangle_{\mathsf{D}}$ for $\langle \mathrm{G}(\mathsf{D},\mathbf{w})\rangle_{\mathrm{p}(\mathsf{D}(N))}$, equation (C.36) approximates the expected generalization error $\langle \mathrm{G}\rangle_{\mathsf{D}}$ from the expected training error $\langle \mathrm{E}\rangle_{\mathsf{D}}$ as (in line with (Akaike, 1969; Murata et al., 1994))

$$\langle \mathrm{G}\rangle_{\mathsf{D}} = \langle \mathrm{E}\rangle_{\mathsf{D}} + \frac{1}{N}\mathrm{tr}\left[\mathbf{J}^{-1}\mathbf{Q}\right] \quad . \tag{4.60}$$

In (4.60) $\mathbf{Q}$ is *Fisher's information matrix* (Mardia et al., 1979, page 98)

$$\mathbf{Q} = \langle \nabla e(\mathbf{x}, g, \mathbf{w}^*)\nabla^{\mathsf{T}} e(\mathbf{x}, g, \mathbf{w}^*)\rangle_{\mathsf{D}} \quad , \tag{4.61}$$

i.e. the second order moment of the error (log-likelihood) gradient of independent observations, evaluated at the true parameters $\mathbf{w}^*$. Further, $\mathbf{J}$ is the *Hessian matrix* of second order derivatives of the regularized training error (4.46)

$$\mathbf{J} = \frac{\partial^2 \mathrm{C}(\mathbf{w}^*)}{\partial\mathbf{w}\partial\mathbf{w}^{\mathsf{T}}} = \frac{\partial^2 \mathrm{E}(\mathbf{w}^*)}{\partial\mathbf{w}\partial\mathbf{w}^{\mathsf{T}}} + \frac{1}{N}\frac{\partial^2 \mathrm{R}(\mathbf{w}^*)}{\partial\mathbf{w}\partial\mathbf{w}^{\mathsf{T}}} = \mathbf{H} + \frac{1}{N}\mathbf{R} \quad , \tag{4.62}$$

conveniently expressed as the sum of the unregularized Hessian $\mathbf{H}$ and the second order derivative of the regularization term.

The very interesting relationship (4.60) expresses that the expected training error is a *biased* estimator of expected generalization error, something that we already argued in section 4.3.2. The derived generalization error estimate quantifies the bias, enabling us to estimate expected generalization performance without setting aside observations in a test set. We must keep in mind, however, the assumption of N being large. Later we shall investigate the extent to which empirical and algebraic generalization error estimates agree.

For a single training error estimate equation (4.60) yields the equivalent non-averaged estimate

$$\hat{\mathrm{G}}(\mathsf{D},\mathbf{w}) = \hat{\mathrm{E}}(\mathsf{D},\mathbf{w}) + \frac{1}{N}\mathrm{tr}\left[\mathbf{J}^{-1}\mathbf{Q}\right] \quad . \tag{4.63}$$

Since training error is the normalized negative log-likelihood

$$\mathrm{E}(\mathsf{D},\mathbf{w}) = -\frac{1}{N}\log \mathrm{p}(\mathsf{D}|\mathbf{w}) \tag{4.64}$$

we identify (writing p_{D} instead of $\mathrm{p}(\mathsf{D}|\mathbf{w}^*)$ for simplicity) the expected second order derivative of the training error as Fisher's information matrix

$$\frac{\partial^2}{\partial \mathbf{w} \partial \mathbf{w}^\top} \langle \mathrm{E}(\mathsf{D}, \mathbf{w}) \rangle_{\mathsf{D}} \tag{4.65}$$

$$= \frac{\partial^2}{\partial \mathbf{w} \partial \mathbf{w}^\top} \int \mathrm{p}_{\mathsf{D}} \left[-\frac{1}{N} \log \mathrm{p}_{\mathsf{D}} \right] \, \mathrm{d}\mathsf{D} \tag{4.66}$$

$$= \frac{1}{N} \int \mathrm{p}_{\mathsf{D}} \left[-\frac{\partial}{\partial \mathbf{w}} \left(\frac{1}{\mathrm{p}_{\mathsf{D}}} \frac{\partial \mathrm{p}_{\mathsf{D}}}{\partial \mathbf{w}^\top} \right) \right] \, \mathrm{d}\mathsf{D} \tag{4.67}$$

$$= \frac{1}{N} \int \mathrm{p}_{\mathsf{D}} \left[\frac{1}{\mathrm{p}_{\mathsf{D}}^2} \frac{\partial \mathrm{p}_{\mathsf{D}}}{\partial \mathbf{w}} \frac{\partial \mathrm{p}_{\mathsf{D}}}{\partial \mathbf{w}^\top} - \frac{1}{\mathrm{p}_{\mathsf{D}}} \frac{\partial^2 \mathrm{p}_{\mathsf{D}}}{\partial \mathbf{w} \partial \mathbf{w}^\top} \right] \, \mathrm{d}\mathsf{D} \tag{4.68}$$

$$= \frac{1}{N} \int \mathrm{p}_{\mathsf{D}} \left[\frac{\partial \log \mathrm{p}_{\mathsf{D}}}{\partial \mathbf{w}} \frac{\partial \log \mathrm{p}_{\mathsf{D}}}{\partial \mathbf{w}^\top} - \frac{1}{\mathrm{p}_{\mathsf{D}}} \frac{\partial^2 \mathrm{p}_{\mathsf{D}}}{\partial \mathbf{w} \partial \mathbf{w}^\top} \right] \, \mathrm{d}\mathsf{D} \tag{4.69}$$

$$= N \langle \nabla \mathrm{E}(\mathbf{w}^*) \nabla^\top \mathrm{E}(\mathbf{w}^*) \rangle_{\mathsf{D}} - \frac{1}{N} \int \mathrm{p}_{\mathsf{D}} \frac{1}{\mathrm{p}_{\mathsf{D}}} \frac{\partial^2 \mathrm{p}_{\mathsf{D}}}{\partial \mathbf{w} \partial \mathbf{w}^\top} \, \mathrm{d}\mathsf{D} \tag{4.70}$$

$$\simeq \mathbf{Q} - \frac{1}{N} \frac{\partial^2}{\partial \mathbf{w} \partial \mathbf{w}^\top} \int \mathrm{p}_{\mathsf{D}} \, \mathrm{d}\mathsf{D} \tag{4.71}$$

$$= \mathbf{Q} \quad , \tag{4.72}$$

where we in (4.71) use (C.18). So, for a single training set we have $\mathbf{Q} = \mathbf{H}$, and the generalization error estimate (4.63) becomes

$$\hat{\mathrm{G}}(\mathsf{D}, \mathbf{w}) = \hat{\mathrm{E}}(\mathsf{D}, \mathbf{w}) + \frac{1}{N} \mathrm{tr} \left[\mathbf{J}^{-1} \mathbf{H} \right] \quad . \tag{4.73}$$

4.3.3.1 Effective number of parameters

With no prior knowledge of the parameters we assume them all to be equally likely, making $\mathrm{p}(\mathbf{w})$ uniform. In this case $\alpha_v = 0$ for all v, and the regularized cost function equals the unregularized one

$$\mathrm{C}(\mathsf{D}, \mathbf{w}) = \mathrm{E}(\mathsf{D}, \mathbf{w}) \quad . \tag{4.74}$$

If follows that $\mathbf{J} = \mathbf{H}$, so (4.73) reduces to

$$\hat{\mathrm{G}}(\mathsf{D}, \mathbf{w}) = \hat{\mathrm{E}}(\mathsf{D}, \mathbf{w}) + \frac{W}{N} \quad , \tag{4.75}$$

where W is the number of model parameters. This means that we by using training error as an estimate of generalization error introduce a bias of W/N, i.e. a term that measures model complexity relative to the number of training set observations. In the general case where $\mathbf{J} \neq \mathbf{H}$ the numerator of the bias term in (4.73) still describes model complexity; we label it the *effective number of parameters* (Moody, 1992; Larsen, 1994; Svarer et al., 1993) and find

$$\hat{\mathrm{G}}(\mathsf{D}, \mathbf{w}) = \hat{\mathrm{E}}(\mathsf{D}, \mathbf{w}) + \frac{W_{\mathrm{eff}}}{N} \quad , \quad W_{\mathrm{eff}} = \mathrm{tr} \left[\mathbf{J}^{-1} \mathbf{H} \right] \quad . \tag{4.76}$$

4.3.4 Model output interpretation

Having observed the value of the expected generalization error as a performance measure we investigate it further. In the MSE case its definition in (4.51) provides for an interesting decomposition, leading to an intuitive understanding of the model output $y(\mathbf{x}, \mathbf{w})$. By labeling

$$\begin{aligned} \langle g|\mathbf{x}\rangle &= \int g\,\mathrm{p}(g|\mathbf{x})\ \mathrm{d}g & (4.77)\\ \langle g^2|\mathbf{x}\rangle &= \int g^2\mathrm{p}(g|\mathbf{x})\ \mathrm{d}g & (4.78)\end{aligned}$$

we can rewrite the integration over $\mathrm{p}(g|\mathbf{x})$ as

$$\begin{aligned} &\int (y(\mathbf{x}, \mathbf{w}) - g)^2\, \mathrm{p}(g|\mathbf{x})\ \mathrm{d}g & (4.79)\\ &= \int (y(\mathbf{x}, \mathbf{w}) - \langle g|\mathbf{x}\rangle + \langle g|\mathbf{x}\rangle - g)^2\, \mathrm{p}(g|\mathbf{x})\ \mathrm{d}g & (4.80)\\ &= \int \Big\{ (y(\mathbf{x}, \mathbf{w}) - \langle g|\mathbf{x}\rangle)^2 + (\langle g|\mathbf{x}\rangle - g)^2 \\ &\qquad +2\,(y(\mathbf{x}, \mathbf{w}) - \langle g|\mathbf{x}\rangle)\,(\langle g|\mathbf{x}\rangle - g)\Big\}\mathrm{p}(g|\mathbf{x})\ \mathrm{d}g & (4.81)\\ &= \int \Big\{ (y(\mathbf{x}, \mathbf{w}) - \langle g|\mathbf{x}\rangle)^2 + (\langle g|\mathbf{x}\rangle - g)^2 \Big\}\, \mathrm{p}(g|\mathbf{x})\ \mathrm{d}g \quad . & (4.82)\end{aligned}$$

Inserting this back into (4.51) we obtain a decomposition of MSE generalization error

$$\begin{aligned} \mathrm{G}_{\mathrm{MSE}}(\mathsf{D}, \mathbf{w}) &= \frac{1}{2}\int (y(\mathbf{x}, \mathbf{w}) - \langle g|\mathbf{x}\rangle)^2\, \mathrm{p}(\mathbf{x})\ \mathrm{d}\mathbf{x} \\ &\quad +\frac{1}{2}\int \left(\langle g^2|\mathbf{x}\rangle - \langle g|\mathbf{x}\rangle^2\right) \mathrm{p}(\mathbf{x})\ \mathrm{d}\mathbf{x} \quad . & (4.83)\end{aligned}$$

The second term is independent of $\mathbf{w}$ and expresses the variance of the system output. The first term shall be decomposed further in a minute, but before doing so we note that generalization error is minimum when the first term in (4.83) is zero, leading to

$$y(\mathbf{x}, \hat{\mathbf{w}}^*) = \langle g|\mathbf{x}\rangle \quad , \tag{4.84}$$

where $\hat{\mathbf{w}}^*$ are the optimal model parameters. This observation is of some importance; it states that the optimal model equals the conditional average of the system output. This is also called the *regression* of g on $\mathbf{x}$. Intuitively, it seems reasonable that the estimated system output is the average of the true conditional system output distribution, at least when modeling the saccade frequency. The situation is sketched in the left panel of figure 4.4. In categorical designs in which the conditional output density is likely to be multi-modal, however, other cost functions may be more appropriate. In that case the averaging of a MSE regressor makes less sense, as outlined in the right panel of figure 4.4; rather, we should attempt to model the modes relative to each other, effectively employing a classification model. Cost functions for classification are discussed in more detail in e.g. (Bridle, 1990), (Bishop, 1995, chapter 6) and (Hintz-Madsen et al., 1995).

4.3.5 Bias and variance

Looking again at the MSE generalization error decomposition in (4.83) we recall that the second term is independent of the model parameters; it is merely the system output variance, or system noise, which we shall denote $\sigma^2_{g|\mathbf{x}}$. In the following we focus on the first term, investigating its properties when averaged over training sets.

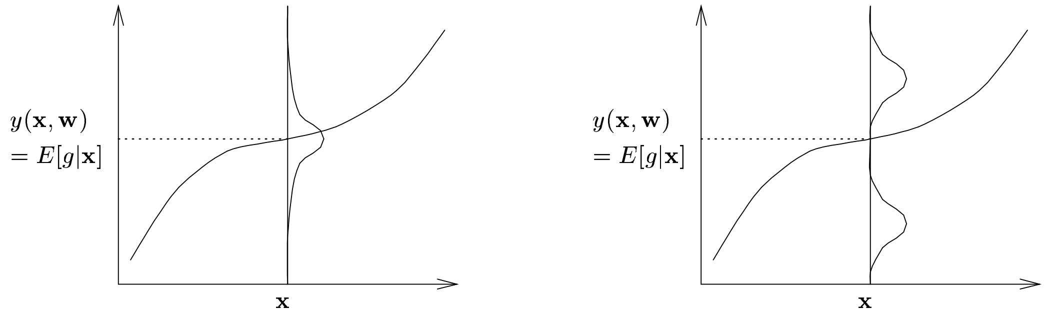

Figure 4.4: Modeling using the mean square error (MSE) cost function. *Left panel*: For regression problems the average of the output conditioned on the input is a reasonable model. *Right panel*: For classification problems the conditional distribution may be multi-modal in which case the average MSE cost function is inappropriate.

The specific training set used to estimate the model parameters enters the generalization error in (4.51). Now, consider the expected generalization error written using the decomposition (4.83)

$$\bar{\mathrm{G}}_{\mathrm{MSE}} = \frac{1}{2}\int \left\langle (y(\mathbf{x},\mathbf{w}) - \langle g|\mathbf{x}\rangle)^2 \right\rangle_{\mathsf{D}} \mathrm{p}(\mathbf{x})\ \mathrm{d}\mathbf{x} + \sigma^2_{g|\mathbf{x}} \quad , \tag{4.85}$$

where we have simplified the expectation notation by substituting D for $\mathrm{p}(\mathsf{D}(N))$. We can further decompose the average operand in the integrand as (Geman et al., 1992; Mørch et al., 1996a)

$$\begin{aligned}
&(y(\mathbf{x},\mathbf{w}) - \langle g|\mathbf{x}\rangle)^2 \\
&= (y(\mathbf{x},\mathbf{w}) - \langle y(\mathbf{x},\mathbf{w})\rangle_{\mathsf{D}})^2 + (\langle y(\mathbf{x},\mathbf{w})\rangle_{\mathsf{D}} - \langle g|\mathbf{x}\rangle)^2 \\
&\quad +2\,(y(\mathbf{x},\mathbf{w}) - \langle y(\mathbf{x},\mathbf{w})\rangle_{\mathsf{D}})\,(\langle y(\mathbf{x},\mathbf{w})\rangle_{\mathsf{D}} - \langle g|\mathbf{x}\rangle) \\
&= (y(\mathbf{x},\mathbf{w}) - \langle y(\mathbf{x},\mathbf{w})\rangle_{\mathsf{D}})^2 + (\langle y(\mathbf{x},\mathbf{w})\rangle_{\mathsf{D}} - \langle g|\mathbf{x}\rangle)^2 \quad ,
\end{aligned} \tag{4.86}$$

to find

$$\bar{\mathrm{G}}_{\mathrm{MSE}} = \sigma^2_{g|\mathbf{x}} \tag{4.87}$$

$$+\frac{1}{2}\int \left\langle (y(\mathbf{x},\mathbf{w}) - \langle y(\mathbf{x},\mathbf{w})\rangle_{\mathsf{D}})^2 \right\rangle_{\mathsf{D}} \mathrm{p}(\mathbf{x})\ \mathrm{d}\mathbf{x} \tag{4.88}$$

$$+\frac{1}{2}\int (\langle y(\mathbf{x},\mathbf{w})\rangle_{\mathsf{D}} - \langle g|\mathbf{x}\rangle)^2\, \mathrm{p}(\mathbf{x})\ \mathrm{d}\mathbf{x} \tag{4.89}$$

$$= \sigma^2_{g|\mathbf{x}} + \text{“variance”} + \text{“bias”} \quad . \tag{4.90}$$

We have achieved a decomposition of the expected generalization error into a system noise term (4.87), a *variance* term (4.88), and a *bias* term[9] (4.89). A closer look at the two last terms justifies the labelling. The variance term expresses the variance of the model $y(\mathbf{x},\mathbf{w})$ over training sets, i.e. the extent to which the model is sensitive to the choice of training set. Conversely, the bias is the squared difference between the average model and the correct regression $\langle g|\mathbf{x}\rangle_{\mathsf{D}}$.

[9]Sometimes the bias is defined only as the difference $\langle y(\mathbf{x},\mathbf{w})\rangle_{\mathsf{D}} - \langle g|\mathbf{x}\rangle$. In that case (4.89) becomes a squared bias term, (“bias”)^2.

Two imaginary models of a system $g = h(\mathbf{x}) + e$ (as in (4.33)) help to illuminate the meaning of bias and variance. Firstly, consider a model that is completely independent of the training set. This can be achieved by fixing the parameters $\mathbf{w}(\mathsf{D}) = \mathbf{w}_0$. Unless we employ some form of prior knowledge the fixed parameter estimate is likely to be poor, resulting in high bias; the model approximates the system rather poorly on average. However, the variance term vanishes since the parameters and thus the model is fixed. Secondly, consider a model that fits the observations in the training set perfectly. Given a sufficiently flexible model this will always be possible[10]. For such a model the expected model output will equal the true regression

$$\langle y(\mathbf{x}, \mathbf{w}) \rangle_{\mathsf{D}} = \langle g|\mathbf{x} \rangle = h(\mathbf{x}) \tag{4.91}$$

for the observations in the training set. If $h(\mathbf{x})$ is smooth it follows that bias will be small in the neighborhood of the training set observations. The variance, on the other hand, becomes large since

$$\left\langle (y(\mathbf{x}, \mathbf{w}) - \langle y(\mathbf{x}, \mathbf{w}) \rangle_{\mathsf{D}})^2 \right\rangle_{\mathsf{D}} = \left\langle (y(\mathbf{x}, \mathbf{w}) - h(\mathbf{w}))^2 \right\rangle_{\mathsf{D}} = \langle e^2 \rangle_{\mathsf{D}} \quad , \tag{4.92}$$

i.e. the variance of the stochastic noise e. It is obvious that a trade-off between bias and variance exists; for training sets of a given size we can reduce bias by employing a relatively complex model. This will, however, increase the variance of the model output due to the model's increased sensitivity to the training set observations. To reduce variance the model needs to be constrained. Yet, a constrained model approximates the system less accurately, meaning that the bias is increased. Hence, we face the so-called *bias-variance trade-off*, as further illustrated in figure 4.5; as functions of model complexity, e.g. the

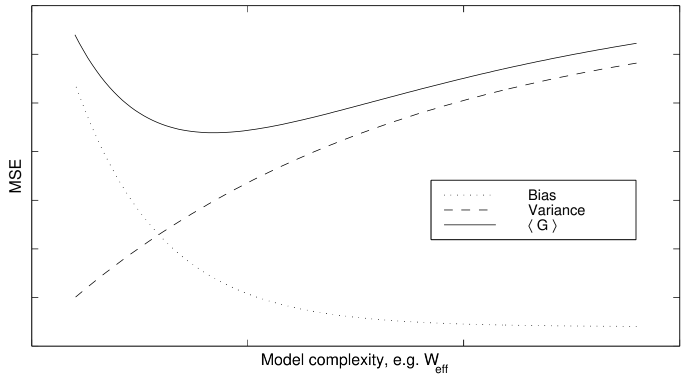

Figure 4.5: Illustration of the bias-variance trade-off. For a given training set size and within the set of possible models as governed by one or more complexity controlling parameters, e.g. W_{eff}, the trade-off between bias and variance may result in an optimal model complexity.

effective number of parameters W_{eff}, bias decreases and variance increases. The result may be that an optimal model complexity exists. Clearly, such a model is optimal for a specific training set size, and only within the set of possible models governed by the complexity controlling parameters.

[10]An unregularized polynomial model of at least the same degree as the number N of training set observations will have zero training error.

To improve model performance for a *specific* model with fixed complexity we must decrease bias and variance *simultaneously*, or at least reduce one of the terms without affecting the other. This can be achieved by increasing the number of training set observations, as we discuss next.

4.3.6 Learning curves

For a given model bias can be decreased by relaxing model constraints, e.g. by including second order polynomial terms in a linear model, or by employing a nonlinear model instead of a linear one. If, at the same time, the number N of training set observations is increased the potential increase in variance can be avoided; model sensitivity to individual observations decreases as their number grows. To illustrate this point we introduce the notion of a *learning curve* (Hertz et al., 1994). A learning curve depicts how expected generalization error evolves with increasing N; it quantifies the improved performance of models based on increasingly large training sets. For an example of learning curves in the context of functional neuro modeling, see (Mørch et al., 1997).

Figure 4.6 sketches the learning curves for two different models. For both, the expected

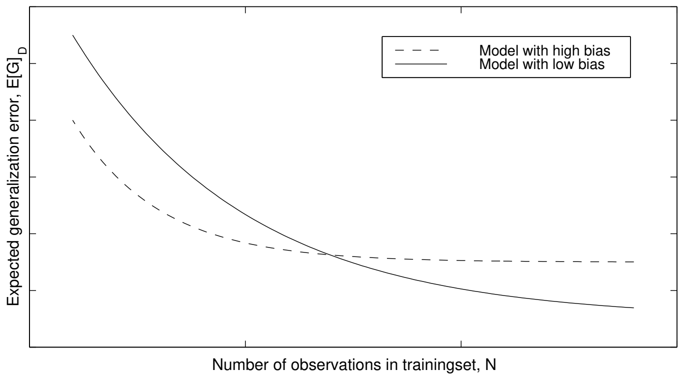

Figure 4.6: Learning curves for two models with different flexibility. A high-bias model (dashed line) is constrained, leading to a rapid, but small generalization error decrease as the training set grows. The larger flexibility of a low-bias model (solid line) holds the potential of improved performance; however, it takes more training set observations for the improvement to manifest itself.

generalization error decreases with large training set sizes. The model variance term (4.88) in the bias-variance decomposition decreases as more and more observations are added to the training set. Eventually, model performance is governed completely by the bias and system noise terms (4.89) and (4.87).

The dashed line in the figure illustrates the performance evolution of a model with relatively high bias. The bias constrains the model's ability to approximate complex relationships, thereby reducing it's sensitivity to the training set. Consequently, a relatively low number of observations is needed to decrease model variance. This is reflected in the rapid generalization error decrease. However, the high bias means that model performance is limited, even when the model is based on large training sets. The situation is different for the more flexible model depicted by the solid line. The lower bias means that many observations are needed to reduce the model variance; the generalization error decrease is slower than for the more biased model. Nevertheless, for large N the performance will predominantly be controlled by the bias, which in this case is low. The important lesson to be learned is that performance depends on both the size of the training set and the complexity of the model employed. We shall quantitatively investigate this phenomenon in the chapters 5 and 6. Before doing so, however, we discuss techniques aimed at controlling model complexity.

4.4 Complexity control

Generalization performance depends on both training set size and model complexity[11], as we have just seen. While the first is easily manipulated[12] the issue of controlling the latter is more involved.

4.4.1 Parameter priors and regularization

We now show how we, by employing a Gaussian parameter prior as in section 4.2.4, can control model flexibility via the variance of the individual parameters. Reiterating, the regularized cost function is

$$\mathrm{C}(\mathsf{D}, \mathbf{w}) = \mathrm{E}(\mathsf{D}, \mathbf{w}) + \frac{1}{N}\mathrm{R}(\mathbf{w}) \quad , \tag{4.93}$$

with

$$\mathrm{R}(\mathbf{w}) = \sum_{v=1}^{W} \frac{\alpha_v}{2} w_v^2 = \frac{1}{2}\mathbf{w}^\mathsf{T}\mathbf{R}\mathbf{w} \quad . \tag{4.94}$$

as the regularization term. The α_v's in the diagonal of $\mathbf{R}$ are the inverse of the prior variances, meaning that $\mathbf{R}$ is positive semidefinite.

Insight into the flexibility constraining properties of (4.94) is gained by considering a Taylor expansion to second order around $\mathbf{w}_0$ of the unregularized cost function $\mathrm{E}(\mathsf{D}, \mathbf{w}) = \mathrm{E}(\mathbf{w})$

$$\mathrm{E}(\mathbf{w}) = \mathrm{E}(\mathbf{w}_0) + \nabla\mathrm{E}(\mathbf{w}_0)(\mathbf{w} - \mathbf{w}_0) + \frac{1}{2}(\mathbf{w} - \mathbf{w}_0)^\mathsf{T}\mathbf{H}(\mathbf{w} - \mathbf{w}_0) \quad , \tag{4.95}$$

where $\mathbf{H}$ is the unregularized Hessian matrix, as in (4.62). For the unregularized model with parameters $\hat{\mathbf{w}}^* = \mathbf{w}_\mathrm{E}$ estimated by minimizing $\mathrm{E}(\mathbf{w})$ we find

$$\nabla\mathrm{E}(\mathbf{w}_\mathrm{E}) = \nabla\mathrm{E}(\mathbf{w}_0) + \mathbf{H}(\mathbf{w}_\mathrm{E} - \mathbf{w}_0) = 0 \quad , \tag{4.96}$$

where we have used (4.95), ignoring terms of higher order than two. Similarly, the cost function gradient for the estimated regularized model parameters $\mathbf{w}_\mathrm{C}$ is

$$\nabla\mathrm{C}(\mathbf{w}_\mathrm{C}) = \nabla\mathrm{E}(\mathbf{w}_0) + \mathbf{H}(\mathbf{w}_\mathrm{C} - \mathbf{w}_0) + \alpha(\mathbf{w}_\mathrm{C} - \mathbf{w}_0) = 0 \quad . \tag{4.97}$$

To simplify the following we have assumed all regularization parameters to be identical $\alpha_w = \alpha$. Combining (4.96) and (4.97) and translating the center of the coordinate system to $\mathbf{w}_0$ we find

$$\mathbf{H}\mathbf{w}_\mathrm{E} = \mathbf{H}\mathbf{w}_\mathrm{C} + \alpha\mathbf{w}_\mathrm{C} \quad . \tag{4.98}$$

[11] In the following we shall use the terms "complexity" and "flexibility" interchangeably.

[12] The limited number of observations in functional datasets constitutes an upper limit on N. The situation is further aggravated if some observations are reserved for empirical estimation of the generalization error, i.e. held out in a test set.

To relate the two sets of parameters further we apply the spectral decomposition of the Hessian

$$\mathbf{H}\mathbf{e}_i = l_i \mathbf{e}_i \quad , \tag{4.99}$$

which provides a set of orthogonal basis vectors, $\mathbf{e}_i$, for parameter space. The representation of $\mathbf{w}_{\mathrm{E}}$ and $\mathbf{w}_{\mathrm{C}}$ using this basis

$$\mathbf{w}_{\mathrm{E}} = \sum_i w_{\mathrm{E},\,i} \mathbf{e}_i \quad , \qquad \mathbf{w}_{\mathrm{C}} = \sum_i w_{\mathrm{C},\,i} \mathbf{e}_i \tag{4.100}$$

inserted into (4.98) yields

$$\mathbf{H} \sum_i w_{\mathrm{E},\,i} \mathbf{e}_i = \mathbf{H} \sum_i w_{\mathrm{C},\,i} \mathbf{e}_i + \alpha \sum_i w_{\mathrm{C},\,i} \mathbf{e}_i \quad . \tag{4.101}$$

Since the $\mathbf{e}_i$'s are orthogonal we obtain an interesting relation between the two sets of transformed parameters

$$l_i w_{\mathrm{E},\,i} \mathbf{e}_i = l_i w_{\mathrm{C},\,i} \mathbf{e}_i + \alpha w_{\mathrm{E},\,i} \mathbf{e}_i \tag{4.102}$$

$$\Updownarrow \tag{4.103}$$

$$w_{\mathrm{C},\,i} = \frac{l_i}{l_i + \alpha} w_{\mathrm{E},\,i} \quad . \tag{4.104}$$

So, in the transformed coordinate system the magnitude of the regularized parameters are reduced compared to the unregularized ones. The reduction depends on the cost function curvature along the axes of the transformed coordinate system, as determined by the l_i's. This is illustrated in figure 4.7, where the ellipse represent a contour of constant unregularized error[13]. The parameters are effectively forced towards $\mathbf{0}$ by the regularization term;

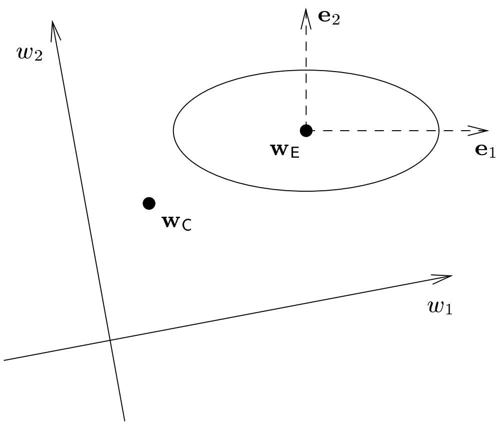

Figure 4.7: The relation between regularized, $\mathbf{w}_{\mathrm{C}}$, and unregularized, $\mathbf{w}_{\mathrm{E}}$, parameter estimates. The regularization reduces the magnitude of the parameters according to the cost function curvature.

this corresponds nicely to the intuitive interpretation of the application of a zero-mean Gaussian parameter prior. The result is a model more constrained than before. It is in this sense that regularization controls model flexibility.

[13] To aid understanding the center of the transformed coordinate system has been translated to $\mathbf{w}_{\mathrm{E}}$ in the figure.

4.4.2 Optimizing the parameter configuration

While regularization facilitates some control over model complexity, a more direct approach is to somehow select which parameters to include in the model. For nonlinear models this field was pioneered by Le Cun et al., see e.g. (Le Cun, Y. et al., 1990; Le Cun, Y. et al., 1989). The approach is, however, hampered by the fact that the number of possible model parameters is infinite[14]. We therefore need a systematic way of exploring the space of parameter configurations. The next section addresses that problem, while the issue of assessing parameter importance is the topic of section 4.4.2.2.

4.4.2.1 Exploring the space of parameter configurations

Several approaches can be taken when exploring the space of parameter configurations. A useful first step is to restrict the class of possible models in such a way that the exploration is limited to a configuration space of finite size. Having done this, three different exploration approaches are:

Exhaustive search where all possible parameter configurations are examined. The obvious problem with this approach is the number of models that require examination; if configuration space is spanned by, say, polynomial models up to order M the number, W, of possible model parameters scales as $[\dim(\mathbf{x})]^M$. Consequently, even for small M the number of possible parameter configurations is huge, rendering the exhaustive search approach impractical.

Model growing which starts from a model with a limited number of parameters. Configuration space is explored by successively adding parameters. In the context of linear models an example of this approach is *forward selection* where parameters are added based on tests of significance (Kendall and Stuart, 1967). Another example is the cascade-correlation learning architecture used for growing nonlinear models (Fahlman and Lebiere, 1990).

Model pruning, conversely, starts with a relatively complex model with many parameters. In this case, and in contrast to model growing, configuration space is explored by removing (or *pruning*) parameters, successively reducing the size of the model (Ripley, 1996). For linear models the approach is known as *backwards elimination* when based on tests of significance, as for forward selection mentioned above[15] (Kendall and Stuart, 1967).

Here we shall focus on model pruning approaches. After choosing a sufficiently complex initial parameter configuration, the exploration of configuration space basically proceeds as follows

1. Estimate the model parameters by minimizing the training error
2. Rank parameters by relative importance
3. Remove (prune) the least important parameters, and return to step 1.

The process continues until only a single parameter remains.

[14] In the context of a polynomial model we can keep adding terms of higher and higher order, each time introduce more model parameters.

[15] For linear models backwards elimination and forward selection may even be combined, in which case we talk of *stepwise regression*. A similar approach can, of course, be employed for other model types.

4.4.2.2 Estimating parameter importance

When exploring configuration space the importance of individual parameters must be evaluated and compared. We shall denote measures of parameter importance as *saliency* measures. The following lists a few candidates. Most attention will be directed to the second of these in the chapters to come.

Magnitude based measures The magnitude $|w_v|$ of parameter v to some extent measures it's saliency; it at least *seems* that the effect of removing very small parameters will be limited. Recall, however, from section 4.4.1 that regularization reduces model complexity by forcing parameters towards zero, *based on the cost function curvature*. So, while a magnitude based saliency measure may seem valid from an *ad hoc* point of view, an approach that explicitly takes the cost function curvature into account is better justified. The next section describes such a measure.

Optimal brain surgeon As a quantitative measure of the effect of removing a parameter (Hassibi and Stork, 1992) proposes the resulting increase in training error[16]. The pruning scheme is called *optimal brain surgeon* (OBS) since it not only estimates parameter saliency, but also provides an estimated location of the cost function minimum after the removal of the least salient parameter. In line with (Hansen and Pedersen, 1994) we extend the approach to deal with regularization. We will assume that the model parameters have been optimized based on the regularized cost function (4.93), so that $\nabla \mathrm{C}(\mathbf{w}_{\mathrm{C}}) = 0$. From the Taylor expansion to second order of the unregularized cost function around $\mathbf{w}_{\mathrm{C}}$ we find the increase due to a small parameter change $\delta \mathbf{w}$

$$\delta \mathrm{E}(\mathbf{w}) = \delta \mathbf{w}^{\mathsf{T}} \nabla \mathrm{E}(\mathbf{w}_{\mathrm{C}}) + \frac{1}{2} \delta \mathbf{w}^{\mathsf{T}} \mathbf{H} \delta \mathbf{w} \quad . \tag{4.105}$$

Now, the parameters are optimized based on the regularized cost function, which means

$$\nabla \mathrm{C}(\mathbf{w}_{\mathrm{C}}) = \nabla \mathrm{E}(\mathbf{w}_{\mathrm{C}}) + \frac{1}{N} \mathbf{R} \mathbf{w}_{\mathrm{C}} = 0 \tag{4.106}$$

$$\Updownarrow$$

$$\nabla \mathrm{E}(\mathbf{w}_{\mathrm{C}}) = -\frac{1}{N} \mathbf{R} \mathbf{w}_{\mathrm{C}} \quad , \tag{4.107}$$

and the training error increase becomes

$$\delta \mathrm{E}(\mathbf{w}) = -\frac{1}{N} \delta \mathbf{w}^{\mathsf{T}} \mathbf{R} \mathbf{w}_{\mathrm{C}} + \frac{1}{2} \delta \mathbf{w}^{\mathsf{T}} \mathbf{H} \delta \mathbf{w} \quad . \tag{4.108}$$

The removal of the v'th parameter corresponds to the parameter change

$$\delta w_v = -w_v \tag{4.109}$$

$$\Updownarrow$$

$$\mathbf{e}_v^{\mathsf{T}} \delta \mathbf{w} + w_v = 0 \quad . \tag{4.110}$$

[16] Parameter saliency may also defined as the increase in estimated *generalization* error; see (Pedersen et al., 1995) for a discussion of saliency measures based on an algebraic generalization error estimate, and (Larsen et al., 1996) for an example of empirical saliency assessment based on an independent set of data.

To find the parameter change that minimizes the cost function increase we employ a Lagrange multiplier of the constraint (4.110)

$$\begin{aligned} \mathrm{S}(\mathbf{w}) &= \mathrm{C}(\mathbf{w}) + \lambda\,(\mathbf{e}_v^\top \delta\mathbf{w} + w_v) && (4.111)\\ &= \frac{1}{2}\delta\mathbf{w}^\top \mathbf{J}\delta\mathbf{w} + \lambda\,(\mathbf{e}_v^\top \delta\mathbf{w} + w_v) \quad , && (4.112) \end{aligned}$$

and minimize it to find the corresponding parameter change

$$\frac{\partial \mathrm{S}(\mathbf{w})}{\partial \delta\mathbf{w}} = \mathbf{J}\delta\mathbf{w} + \lambda\mathbf{e}_v = 0 \qquad (4.113)$$

$$\Updownarrow$$

$$\delta\mathbf{w} = -\lambda\mathbf{J}^{-1}\mathbf{e}_v \quad . \qquad (4.114)$$

Inserted into (4.110) this yields the Lagrange multiplier

$$\lambda = \frac{w_v}{(\mathbf{J}^{-1})_{vv}} \quad , \qquad (4.115)$$

which in turn yields the the optimal parameter change

$$\begin{aligned} \delta\mathbf{w} &= -\lambda\mathbf{J}^{-1}\mathbf{e}_v && (4.116)\\ &= -\frac{w_v}{(\mathbf{J}^{-1})_{vv}}\mathbf{J}^{-1}\mathbf{e}_v \quad . && (4.117) \end{aligned}$$

Finally, we find the saliency for parameter v as the estimated training error increase by inserting (4.117) into (4.108)

$$\begin{aligned} \delta\mathrm{E}_v(\mathbf{w})_{\mathrm{OBS}} &= -\frac{1}{N}\delta\mathbf{w}^\top \mathbf{R}\mathbf{w}_{\mathrm{C}} + \frac{1}{2}\delta\mathbf{w}^\top \mathbf{H}\delta\mathbf{w} && (4.118)\\ &= \frac{1}{N}\frac{w_v}{(\mathbf{J}^{-1})_{vv}}\mathbf{e}_v^\top \mathbf{J}^{-1}\mathbf{R}\mathbf{w}_{\mathrm{C}} + \frac{1}{2}\left(\frac{w_v}{(\mathbf{J}^{-1})_{vv}}\right)^2 \mathbf{e}_v^\top \mathbf{J}^{-1}\mathbf{H}\mathbf{J}^{-1}\mathbf{e}_v && (4.119)\\ &= \frac{1}{N}\frac{w_v}{(\mathbf{J}^{-1})_{vv}}\mathbf{e}_v^\top \mathbf{J}^{-1}\mathbf{R}\mathbf{w}_{\mathrm{C}} + \frac{1}{2}\left(\frac{w_v}{(\mathbf{J}^{-1})_{vv}}\right)^2 \left(\mathbf{J}^{-1}\mathbf{H}\mathbf{J}^{-1}\right)_{vv} \quad . && (4.120) \end{aligned}$$

We observe that with no regularization, $\alpha_v = 0 \;\Rightarrow\; \mathbf{R} = \mathbf{0}$, $\mathbf{J} = \mathbf{H}$, the saliency reduces to

$$\begin{aligned} \delta\mathrm{E}_v(\mathbf{w})_{\mathrm{OBS}}\big|_{\alpha_v=0} &= \frac{1}{2}\left(\frac{w_v}{(\mathbf{J}^{-1})_{vv}}\right)^2 \left(\mathbf{J}^{-1}\mathbf{J}\mathbf{J}^{-1}\right)_{vv} && (4.121)\\ &= \frac{1}{2}\frac{w_v^2}{(\mathbf{J}^{-1})_{vv}} \quad . && (4.122) \end{aligned}$$

Optimal brain damage By employing a diagonal approximation of the Hessian matrices the optimal parameter change becomes

$$\delta\mathbf{w} = -\frac{w_v}{(\mathbf{J}^{-1})_{vv}}\mathbf{J}^{-1}\mathbf{e}_v = -\frac{w_v}{(\mathbf{J}^{-1})_{vv}}(\mathbf{J}^{-1})_{vv}\mathbf{e}_v = -w_v\mathbf{e}_v \quad , \qquad (4.123)$$

which trivially reiterates the removal of the v'th parameter; only trivial information about the parameter change is gained, since we ignore information about second order derivatives with respect to different parameters. For this reason the pruning scheme based on diagonal approximations of the Hessian matrices is called *optimal brain damage*[17] (OBD) (Le Cun, Y. et al., 1990). The OBD saliency estimate correspondingly becomes

$$\begin{aligned}
\delta \mathrm{E}_v(\mathbf{w})_{\mathrm{OBD}} &= \frac{w_v^2}{N}\mathbf{R}_{vv} + \frac{1}{2}\left(\frac{w_v}{(\mathbf{J}^{-1})_{vv}}\right)^2 \left(\mathbf{J}^{-1}\right)_{vv} \mathbf{H}_{vv} \left(\mathbf{J}^{-1}\right)_{vv} && (4.124)\\
&= \frac{w_v^2}{N}\alpha_v + \frac{1}{2}w_v^2\mathbf{H}_{vv} && (4.125)\\
&= \left(\frac{\alpha_v}{N} + \frac{\mathbf{H}_{vv}}{2}\right) w_v^2 \quad . && (4.126)
\end{aligned}$$

While a diagonal approximation of the Hessian may seem crude experience shows their performance to be close to identical. In fact, (Pedersen, 1997) reports cases where OBD performs better than OBS, due to inaccuracies of the quadratic approximation.

4.5 Summary

Analysis of variance reveals that experimentally induced variance of interest constitutes only a tiny fraction of the total microscopic variance; for the analyzed functional dataset only the variance along the tenth principal axis, which accounts for a mere 2% of the total variance, is dominated by effects related to intra-subject differences. The relatively strong correlation with the microscopic observations along one particular principal basis vector is not reflected in the basis provided by independent component analysis. However, the phenomenon is not easily interpretable; it is possible that the activity of involved neuro-physiological systems combine in a nonlinear fashion to produce the observed microscopic patterns.

To quantitatively assess model performance a statistical framework is proposed. The approach is centered around measures of model generalization ability, i.e. performance of models with parameters estimated in the limit of infinitely many observations. While generalization theory is well-studied in many areas, it's application is novel in the context of functional neuro modeling. Specifically, the observation that performance depends on both the number of observations and model complexity is important; it facilitates the determination of the extent to which a given dataset supports the application of complex models over other, more simple ones.

[17] As opposed to optimal brain *surgeon* which potentially provides non-trivial information about the parameter change.

Chapter 5

Linear modeling

In this chapter we exemplify the generalization theoretical framework proposed in the previous chapter in the context of linear models of the conditional input-output distribution, and apply the techniques to the CPH/SAC dataset.

5.1 Linear microscopic regression

Recall from chapter 2 how Bayes theorem provides decompositions of $\mathrm{p}(\mathbf{x}, \mathbf{g})$. As in the previous chapter we focus first on the conditional macroscopic distribution $\mathrm{p}(\mathbf{g}|\mathbf{x})$, i.e. the conditional distribution of the macroscopic behavior on the microscopic variables, like in (4.31).

We investigate one specific example of the ML estimation approach of section 4.2.3; in line with (4.33) we assume the elements of the macroscopic vector to be governed by a deterministic, but *linear*, function of $\mathbf{x}$ with added Gaussian noise; for the k'th macroscopic element it reads

$$\begin{aligned} g_k &= h(\mathbf{x}) + e_k && (5.1)\\ &= \sum_{i=1}^{d} w_{k,i} x_i + e_k && (5.2)\\ &= \mathbf{w}_k^{\mathsf{T}} \mathbf{x} + e_k \quad , && (5.3) \end{aligned}$$

where i indices the elements of the microscopic vector[1]. Since g_k is linearly expressed in terms of the microscopic vector $\mathbf{x}$ we talk of *linear microscopic regression.* By arranging the model parameters in a matrix we can simultaneously express all K elements

$$\begin{aligned} \mathbf{g} &= [\mathbf{w}_1\ \mathbf{w}_2 \cdots \mathbf{w}_K]^{\mathsf{T}}\, \mathbf{x} + \mathbf{e} && (5.4)\\ &= \mathbf{W}^{\mathsf{T}} \mathbf{x} + \mathbf{e} \quad . && (5.5) \end{aligned}$$

Note that parameter vectors for individual macroscopic elements are *columns* of $\mathbf{W}$. The noise is assumed to be Gaussian with zero mean and covariance structure Σ_e, i.e. $\mathbf{e} \sim N(\mathbf{0}, \Sigma_e)$.

[1] Again, the microscopic variables may be represented using any basis that spans signal space, so when we write $\mathbf{x}$ (which is d-dimensional) in the following it may be substituted by other representations, such as the vector of principal components $\mathbf{z}$ (which is $(N-1)$-dimensional).

5.1.1 Parameter estimation

Analogous to section 4.2.3 the Gaussian noise assumption yields the conditional distribution

$$\mathrm{p}(\mathbf{g}|\mathbf{x},\mathbf{W}) = \frac{1}{\sqrt{|2\pi\Sigma_e|}} \exp\left[-\frac{1}{2}(\mathbf{g}-\mathbf{W}^\top\mathbf{x})^\top\Sigma_e^{-1}(\mathbf{g}-\mathbf{W}^\top\mathbf{x})\right] \quad ; \tag{5.6}$$

likewise multivariate normal. Under the assumption of independent observations the unregularized MSE cost function correspondingly becomes[2]

$$\begin{aligned} \mathrm{E}(\mathsf{D},\mathbf{W}) &= -\frac{1}{N}\sum_{n=1}^{N}\log \mathrm{p}(\mathbf{g}_n|\mathbf{x}_n,\mathbf{W}) && (5.7)\\ &= \frac{1}{2N}\sum_{n=1}^{N}(\mathbf{g}_n-\mathbf{W}^\top\mathbf{x}_n)^\top\Sigma_e^{-1}(\mathbf{g}_n-\mathbf{W}^\top\mathbf{x}_n) && (5.8)\\ &= \frac{1}{2N}\mathrm{tr}\left[(\mathbf{G}-\mathbf{W}^\top\mathbf{X})^\top\Sigma_e^{-1}(\mathbf{G}-\mathbf{W}^\top\mathbf{X})\right] \quad , && (5.9) \end{aligned}$$

where terms that are independent of $\mathbf{W}$ have been ignored. In (5.9) the summation over observations is expressed as a matrix trace by employing the micro- and macroscopic *data matrices* defined in chapter 1, both with individual observations arranged in columns[3].

Employing ML estimation the optimal parameters are those that minimize $\mathrm{E}(\mathsf{D},\mathbf{W})$, so we find the cost function derivative

$$\frac{\partial \mathrm{E}(\mathsf{D},\mathbf{W})}{\partial \mathbf{W}} = -\frac{1}{N}\mathbf{X}\Sigma_e^{-1}(\mathbf{G}-\mathbf{W}^\top\mathbf{X})^\top \tag{5.10}$$

and set it equal to zero to yield

$$\begin{aligned} \frac{\partial \mathrm{E}(\mathsf{D},\hat{\mathbf{W}})}{\partial \mathbf{W}} &= 0 \\ &\Updownarrow \\ \mathbf{X}\Sigma_e^{-1}\mathbf{G}^\top &= \mathbf{X}\Sigma_e^{-1}\mathbf{X}^\top\hat{\mathbf{W}} \\ &\Updownarrow \\ \hat{\mathbf{W}} &= \left(\mathbf{X}\Sigma_e^{-1}\mathbf{X}^\top\right)^{-1}\mathbf{X}\Sigma_e^{-1}\mathbf{G}^\top \end{aligned} \tag{5.11}$$

as the ML estimate of the model parameters. If the true covariance matrix is unknown we assume it diagonal, e.g. $\Sigma_e = \sigma^2\mathbf{I}$, which leads to the parameter estimate

$$\hat{\mathbf{W}} = (\mathbf{X}\mathbf{X}^\top)^{-1}\mathbf{X}\mathbf{G}^\top \quad . \tag{5.12}$$

In the specific case of the CPH/SAC dataset we have $K = 1$; with the row vector $\mathbf{g} = [g_{1,1}\ g_{1,2}\ \cdots\ g_{1,N}]$ denoting the vector of scalar macroscopic observations (saccade frequency observations) we obtain

$$\hat{\mathbf{w}} = (\mathbf{X}\mathbf{X}^\top)^{-1}\mathbf{X}\mathbf{g}^\top \quad . \tag{5.13}$$

[2] In this chapter we consider only MSE cost functions; for simplicity the subscript used in chapter 4 is dropped.

[3] This is in contrast to traditional notation as in e.g. (Mardia et al., 1979), where data matrices are composed of observations in *rows*. As a consequence care has to be taken when comparing expressions herein, e.g. parameter estimates, with derivations elsewhere.

5.2 Complexity control

To facilitate complexity control of the linear microscopic regression model we evaluate the two approaches, regularization and parameter configuration optimization, proposed in section 4.4.

5.2.1 Gaussian prior and ridge regression

Consider the case of univariate macroscopic observations, as above. If we assume a Gaussian parameter prior of the form discussed in section 4.2.4

$$\mathrm{R}(\mathbf{w}) = \frac{1}{2}\mathbf{w}^\top \mathbf{R}\mathbf{w} \quad , \tag{5.14}$$

the regularized cost function becomes

$$\mathrm{C}(\mathsf{D}, \mathbf{w}) = \frac{1}{2N}(\mathbf{g} - \mathbf{w}^\top \mathbf{X})^\top (\mathbf{g} - \mathbf{w}^\top \mathbf{X}) + \frac{1}{2N}\mathbf{w}^\top \mathbf{R}\mathbf{w} \quad . \tag{5.15}$$

Differentiation with respect to $\mathbf{w}$ yields

$$\frac{\partial \mathrm{C}(\mathsf{D}, \mathbf{w})}{\partial \mathbf{w}} = \frac{1}{N}\left[-\mathbf{X}(\mathbf{g} - \mathbf{w}^\top \mathbf{X})^\top + \mathbf{R}\mathbf{w}\right] \tag{5.16}$$

leading to the corresponding regularized parameter estimate

$$\begin{aligned} \mathbf{X}\mathbf{g}^\top &= \mathbf{X}\mathbf{X}^\top \hat{\mathbf{w}} + \mathbf{R}\hat{\mathbf{w}} \\ &\Updownarrow \\ \hat{\mathbf{w}} &= \left(\mathbf{X}\mathbf{X}^\top + \mathbf{R}\right)^{-1} \mathbf{X}\mathbf{g}^\top \quad . \end{aligned} \tag{5.17}$$

If the regularizer is of simple diagonal form, $\mathbf{R} = \alpha \mathbf{I}$, equation (5.17) is the *Ridge* estimate of $\mathbf{w}$ (Hoerl and Kennard, 1970). Thus, from the discussion in section 4.4.1 we recognize Ridge regression is a linear model, of which the flexibility can be controlled via the regularization parameter α.

5.2.2 Parameter pruning

With reference to section 4.4.2 we recall how model pruning is one particular way to explore the space of parameter configurations, the aim of course being to control model complexity more directly than what is possible using regularization. In particular, OBS was derived as an estimate of parameter importance. Next we compute the Hessian matrices for the linear microscopic regression model. Further, we briefly discuss pruning techniques based on tests of parameter significance.

5.2.2.1 Optimal brain surgeon

The regularized Hessian matrix of second order derivatives is easily found from (5.16)

$$\mathbf{J} = \frac{\partial^2 \mathrm{C}(\mathsf{D}, \mathbf{w})}{\partial \mathbf{w} \partial \mathbf{w}^\top} = \frac{1}{N}\mathbf{X}\mathbf{X}^\top + \frac{1}{N}\mathbf{R} = \mathbf{H} + \frac{1}{N}\mathbf{R} \quad . \tag{5.18}$$

Insertion into (4.117) yields the minimum parameter change caused by the removal of a single parameter. The cost function increase is correspondingly estimated by inserting $\mathbf{J}$ and $\mathbf{H}$ into the OBS saliency expression (4.120).

It is evident from (5.18) that the second order Taylor expansion of the cost function that underlies the OBS parameter saliency estimate is exact in the linear case.

5.2.2.2 Testing parameter significance

For the linear model it is possible to devise a parametric test of the hypothesis that the squared error of the model based on one subset of microscopic variables differs significantly from the squared error of that based on a different set. When applied successively to an initially large model, each time removing the least significant parameter, this approach is known as backwards elimination (Kendall and Stuart, 1967).

The technique closely resembles parameter pruning based on OBS or OBD in that it facilitates the elimination of parameters based on their contribution to the squared model error. For a further discussion of backwards elimination and illustrations of it's application in functional neuro modeling see e.g. (Mørch and Thomsen, 1994; Lundsager and Kristensen, 1996).

5.3 Application to the CPH/SAC dataset

We now turn to investigate the effects of model complexity and training set size on the generalization performance of the linear microscopic regression model when applied to the CPH/SAC dataset. Efficient microscopic representations in signal space are provided by the principal components and independent projections as computed in section 3.5.

5.3.1 Complexity control

To empirically estimate generalization error a test set of $N_{\mathsf{T}} = 16$ observations was randomly selected. The remaining observations were bootstrapped (sampled with replacement) to yield $M = 20$ training sets, all of size $N = 48$. Varying both the regularization and the parameter configuration as described below, the parameters of the linear microscopic regression model was subsequently estimated from each of the training sets to yield sets of models $\hat{h}^{(m)}(\mathbf{x})$, $m = 1, \ldots , M$ of the macroscopic variable (saccade frequency). These in turn facilitated the average test error as an estimate of the expected generalization error.

While bootstrap sampling apparently facilitates an estimate of the expected generalization error, sample statistics for which bootstrapping fails do exist. One such example is the expected sample maximum. Attempting to estimate this by bootstrapping an observation pool of finite size yields no additional information since the bootstrap sample maximum is limited by the sample maximum of the original observation pool. At present, however, we have no reason to believe that the sample test error belongs to the class of sample statistics for which bootstrapping fails, but this issue definitely deserves further investigation.

In the following we empirically investigate how model complexity affects the estimated generalization ability of the linear microscopic regression model.

5.3.1.1 Regularization

We begin by controlling model complexity via a single regularization parameter α as in section 5.2.1, effectively employing Ridge regression models. Ideally, we should evaluate generalization performance for all combinations of model complexity and training set size. In practice, however, it is impossible to sample the space of possible combinations in more than a few points; something especially true for the nonlinear models of chapter 6, as we shall see. Consequently, the results that appear in the current as well as the next chapter serve merely to illustrate the principles of chapter 4; they by no means convey the whole picture. Therefore, models with better performance than those reported here almost certainly exist, even within the restricted class of linear microscopic regression models.

Figure 5.1 shows the result of controlling model complexity via regularization for the microscopic vectors represented using the PCA basis. It appears that generalization error

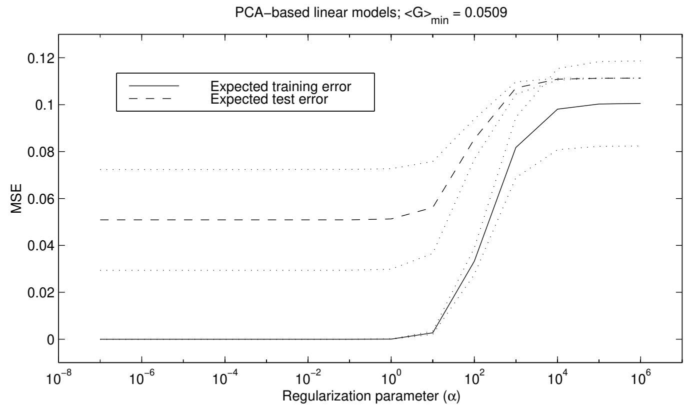

Figure 5.1: The expected generalization error (test error) as function of the regularization parameter α for linear microscopic Ridge regression PCA-based models of the saccade frequency for the CPH/SAC dataset. The estimate indicates that heavily constrained models (large values of α) yield reduced generalization performance. The dotted lines represent one standard deviation of the test error estimate.

increases with large values of α; this is in line with the discussion of the model flexibility constraining properties of regularization in section 4.4.1. However, no well-defined generalization error minimum exists; rather, generalization error is constant for small values of α. Decreasing generalization performance for small values of α would be the result of a model overly sensitive to the training set data. The fact that generalization error remains constant for even very small values of α in figure 5.1 seems to indicate that the bias introduced by the linear nature of the model itself reduces the problem of over-fitting. The inflexible nature of the linear model may mean, however, that performance better than the estimated generalization error of 0.0509 can be obtained with more flexible models. We shall investigate this issue in the next chapter.

For models based on the microscopic observations represented using the ICA basis the situation is as illustrated in figure 5.2. The picture resembles that of the PCA-based models. However, the minimum value of the estimated generalization error is smaller than before. Consequently, for the specific, fully parameterized linear models[4], the ICA basis seems more informative than it's PCA counterpart. Next we investigate if this holds for other parameter configurations as well.

Together, figures 5.1 and 5.2 indicate that, for regularized linear models based on bootstrapped training sets of a given size, model performance can be partially controlled via the regularization parameter α. For large values of α the models are heavily constrained and model performance suffers; performance improves with decreasing regularization and thus model bias, but only down to a certain level.

[4]Meaning that they are based on all elements of the projection vectors.

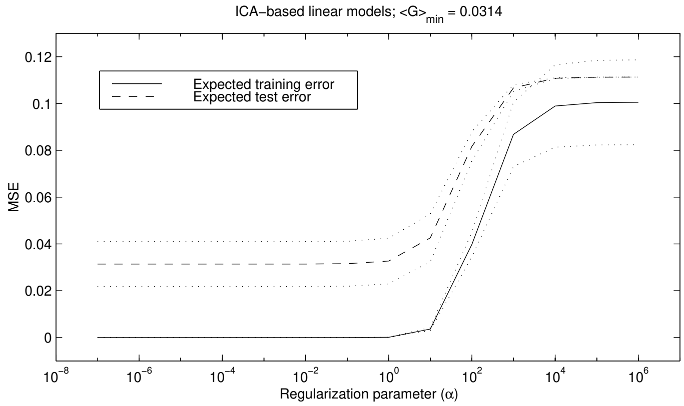

Figure 5.2: The expected generalization error (test error) as function of the regularization parameter α for linear microscopic Ridge regression ICA-based models of the saccade frequency for the CPH/SAC dataset. The dotted lines represent one standard deviation of the test error estimate. The situation resembles that of PCA-based models; however, the minimum estimated expected generalization error value is smaller.

5.3.1.2 Parameter pruning

Using the Hessian matrices in (5.18) model complexity can be controlled by OBS-based parameter pruning. Starting from linear models based on all elements of the projection vectors (be it the vectors of principal components or the vectors of independent projections) we proceed by eliminating a single parameter at a time. The elimination of parameters corresponds to model space reduction; since the j'th parameter quantifies the influence of the j'th basis vector, model space is reduced when the parameter is removed.

Figure 5.3 reproduces the evolution of model performance for the PCA-based models as pruning progresses. The regularization parameter was set to $\alpha = 0.01$, which from figure 5.1 yields close to optimal generalization performance. We observe that both the algebraic and the empirical generalization error estimates predict optimal model performance for models with a relatively low number of parameters. The algebraic estimate, however, suffers from the very low number of independent training set observations. This in turn limits the rank of the Hessian matrices and thus the effective number of parameters. The effect is clearly seen in the figure; the effective number of parameters is never larger than 48, resulting in an almost constant algebraic generalization error estimate for models with some forty parameters or more. In effect, the algebraic generalization error estimate is rendered inadequate by the low number of independent observations; consequently, we shall use only test error estimates to empirically identify optimal model complexity in the following.

The dotted vertical line in figure 5.3 indicates the average test error minimum which occurs for models with two parameters. For better illustration the algebraic estimate is left out in figure 5.4 which reproduces the test error estimate of the expected generalization error, along with error-bars indicating one standard deviation.

While the parameter configuration of the individual optimal models may differ, so that they not all correspond to the same model space, the tenth principal axis is included in all of them. Referring back to the ANOVA plot in figure 4.2 we recognize this axis as the

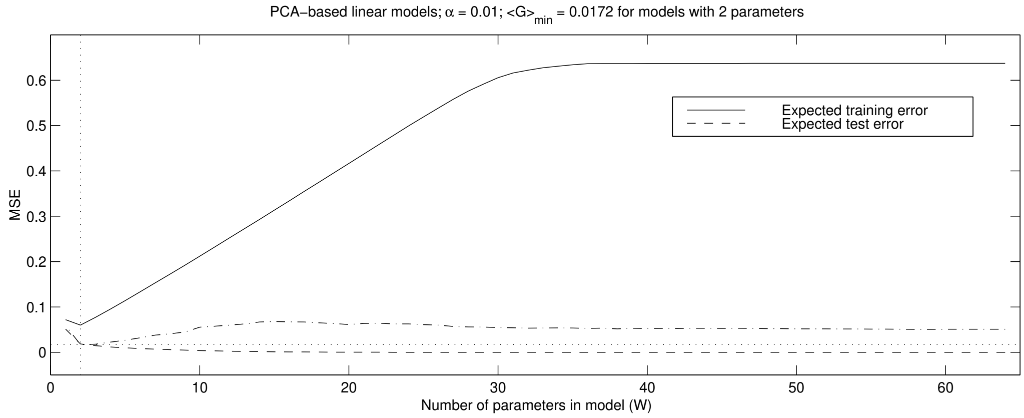

Figure 5.3: The estimated expected generalization error as function of the number of parameters for linear microscopic Ridge regression PCA-based models of the saccade frequency for the CPH/SAC dataset. The regularization parameter is $\alpha = 0.01$. The empirical (dash-dotted line) as well as the algebraic (solid line) estimates predict model performance to be optimal for models with a relatively small number of parameters.

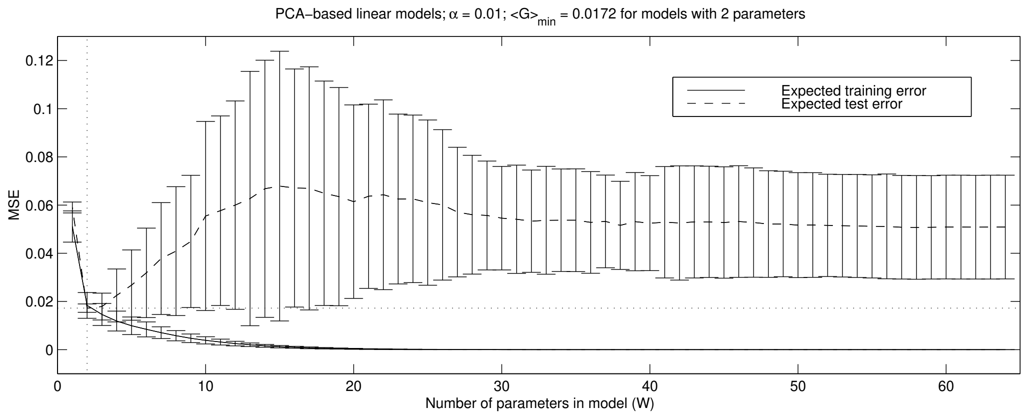

Figure 5.4: The empirically estimated expected generalization error as function of the number of parameters for linear microscopic Ridge regression PCA-based models of the saccade frequency for the CPH/SAC dataset. The regularization parameter is $\alpha = 0.01$. Error-bars represent one standard deviation of the error estimates.

one most dominated by intra-subject effects[5]. The minimum average test error amounts to 0.0172, which is considerably less than the minimum value of 0.0509 achieved when controlling model complexity only by varying regularization as plotted in figure 5.1. The interpretation is straightforward: models that incorporate information from all principal components are overly sensitive to the training set observations. Better generalization performance is achieved by ignoring the information along all but a few principal axis.

Figure 5.5 displays the evolution of model performance for models based on the independent projections. Again optimal performance is estimated for models with a limited

[5]Remember that the variances in figure 4.2 are relative; the true variances are obtained by scaling with the eigenvalues as plotted in the upper panel of figure 4.1.

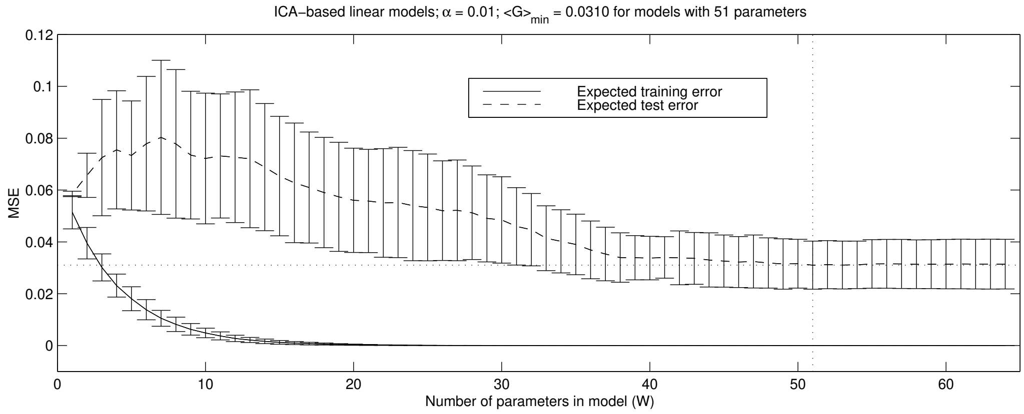

Figure 5.5: The estimated expected generalization error as function of the number of parameters for linear microscopic Ridge regression ICA-based models of the saccade frequency for the CPH/SAC dataset. The regularization parameter is $\alpha = 0.01$. Error-bars represent one standard deviation of the error estimates.

number of parameters. However, the empirical test error estimate indicates that only a few elements of the vector of independent projections should be ignored. The resulting models are relatively large compared to the small models that were predicted to be optimal based on the principal axes; further, they yield slightly worse generalization performance. The independent axes, in other words, provide less informative projections than does their principal equivalents. It seems reasonable to assume that the difference relates to the fact that the variance of the independent projections remain relatively unrelated to intra-subject effects on an individual basis, as we saw in the beginning of chapter 4. Hence, the results argue in disfavor of linear models based on the basis provided by ICA. However, since a linear mixture model for brain function is unrealistic the apparent problems with model space identification based on the ICA representation may be related to the application of linear models rather than to the ICA basis itself. The next chapter will attempt to address this issue.

5.3.2 Learning curves

To investigate the impact of training set size on generalization performance the pool of 48 training observations was bootstrapped to provide sets of increasing size. Twenty sets were generated for each size; sizes ranged from 10 to 100. Model regularization was fixed by employing Ridge regression models with identical α's of 0.01.

The resulting learning curve[6] for models with two parameters based on the PCA basis is displayed in figure 5.6. As the number of training set observations increases generalization ability improves. Eventually the average test error stabilizes at a level close to the minimum value 0.0172 of figure 5.4. Similarly, figure 5.7 depicts the learning curve of ICA-based models with 51 parameters. Again the average test error is high for models based on small training sets. As more observations become available performance approaches and eventually falls below the test error of 0.0310 in figure 5.5. For all the examined training set sizes models based on the ICA basis performs worse than the PCA-based

[6]As before expected generalization error is empirically estimated using the test set.

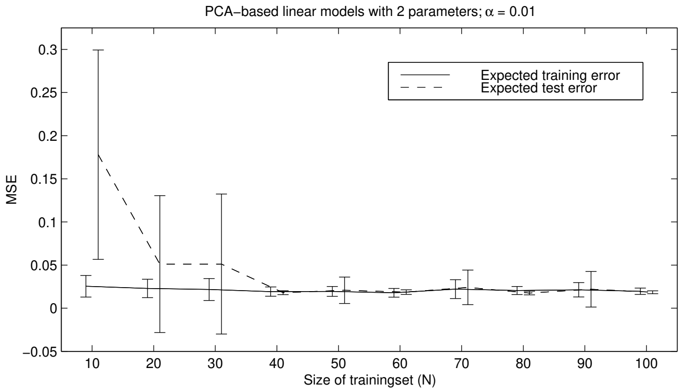

Figure 5.6: Empirical learning curve for two-parameter Ridge regression models with identical regularization parameters $\alpha = 0.01$, based on the PCA basis for the CPH/SAC dataset. The error-bars represent one standard deviation of the error estimates. As the number of training set observations increases generalization performance improves.

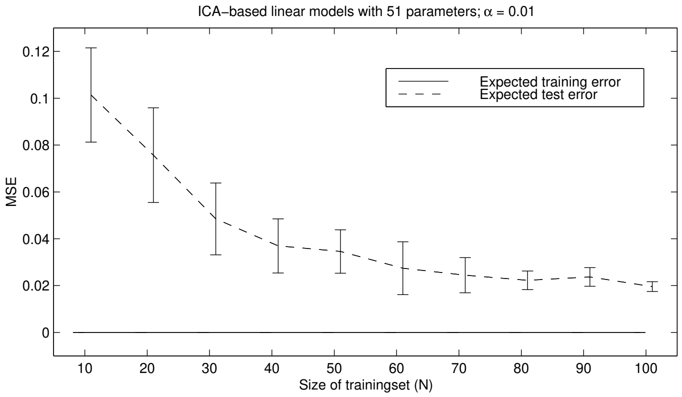

Figure 5.7: Empirical learning curve for 51-parameter Ridge regression models with identical regularization parameters $\alpha = 0.01$, based on the ICA basis for the CPH/SAC dataset. Error-bars represent one standard deviation of the error estimates. As the number of training set observations increases generalization error settles at higher value than for the PCA-based models.

ones. However, the difference for relatively large training sets is less than for smaller training set sizes; this observation is in line with the comments in section 4.3.6 and owes to the fact that ICA-based models with 51 parameters are more flexible than their two-parameter PCA-based counterparts, meaning that more observations are needed to obtain comparable generalization performance.

The estimated learning curves for the linear Ridge regression models verify that model performance is a function of the size of the training set; whether or not a particular model is used to it's full potential depends on the number of observations available to estimate the parameters. An initial performance increase can be achieved by adding more observations to the training set. However, when the training set increases above a certain size very little is gained; this is due to flexibility constraints in the model itself. To improve generalization performance beyond this point a different, more flexible class of models must be employed. We shall investigate one particular such model class in the next chapter. First, however, we address linear models further.

5.4 The general linear model

An alternative to linear microscopic regression can be obtained by decomposing the joint micro- and macroscopic density as $\mathrm{p}(\mathbf{x}, \mathbf{g}) = \mathrm{p}(\mathbf{x}|\mathbf{g})\mathrm{p}(\mathbf{g})$, in which case we can regard the system governed by the conditional distribution $\mathrm{p}(\mathbf{x}|\mathbf{g})$ as the one to model. This effectively corresponds to a shift in system boundaries, as discussed in chapter 2. The widely used so-called statistical parametric mapping (SPM) tools (Friston et al., 1995; Friston et al., 1996) models $\mathrm{p}(\mathbf{x}|\mathbf{g})$ rather than $\mathrm{p}(\mathbf{g}|\mathbf{x})$, and do so by employing the *general linear model* (GLM). In the following we discuss GLM and compare it to linear microscopic regression. For a recent review of linear modeling of functional datasets see (Worsley et al., 1998).

In contrast to linear microscopic regression the GLM assumes the elements of the microscopic vector to be governed by a deterministic, linear function of the macroscopic vector $\mathbf{g}$ with added Gaussian noise. Thus, the i'th microscopic vector element (which is a voxel if the original Euclidean basis is used) is

$$
\begin{aligned}
x_i &= f(\mathbf{g}) + \epsilon_i && (5.19)\\
&= \sum_{k=1}^{K} b_{i,k} g_k + \epsilon_i && (5.20)\\
&= \mathbf{b}_i^\intercal \mathbf{g} + \epsilon_i \quad . && (5.21)
\end{aligned}
$$

As for the linear microscopic regression model we can simultaneously express the relationship for all d elements of all N observations as

$$
\begin{aligned}
\mathbf{X} &= [\mathbf{b}_1 \; \mathbf{b}_2 \; \cdots \; \mathbf{b}_d]^\intercal \, \mathbf{G} + \epsilon && (5.22)\\
&= \mathbf{B}^\intercal \mathbf{G} + \epsilon \quad , && (5.23)
\end{aligned}
$$

where the parameter matrix $\mathbf{B}$ is composed of the individual parameter vectors $\mathbf{b}_i$ in columns. In this context the macroscopic data matrix $\mathbf{G}$ is called the *design matrix*; it is often divided into two distinct parts assumed to hold interesting and uninteresting effects (frequently dubbed *covariates*), respectively[7] (Friston et al., 1996).

By assuming the noise to be Gaussian as before, i.e. $\epsilon \sim N(\mathbf{0}, \Sigma_\epsilon)$, it is straightforward to obtain the ML estimate

$$
\hat{\mathbf{B}} = \left(\mathbf{G}\Sigma_\epsilon^{-1}\mathbf{G}^\intercal\right)^{-1} \mathbf{G}\Sigma_\epsilon^{-1}\mathbf{X}^\intercal \quad , \qquad (5.24)
$$

in analogy to (5.11).

5.4.1 Relationship between linear microscopic regression and GLM

The two linear modeling approaches of linear microscopic regression and the general linear model may seem principally different at first sight. They are, however, analogous; to see

[7] While the design matrix is normally partitioned column-wise as $\mathbf{G} = [\mathbf{G}_0|\mathbf{G}_1]$ the partitioning would be row-wise in our case due to the transposed nature of the data matrices described in chapter 1 as compared to the conventional notation used in e.g. (Mardia et al., 1979; Friston et al., 1996).

this we first use Bayes theorem to rewrite the joint density

$$\begin{aligned} \mathrm{p}(\mathbf{x},\mathbf{g}) &= \mathrm{p}(\mathbf{g}|\mathbf{x})\mathrm{p}(\mathbf{x}) &(5.25)\\ &= \mathrm{p}(\mathbf{x}|\mathbf{g})\mathrm{p}(\mathbf{g}) \quad . &(5.26)\end{aligned}$$

This establishes a link between $\mathrm{p}(\mathbf{x}|\mathbf{g})$ and $\mathrm{p}(\mathbf{g}|\mathbf{x})$ that requires knowledge of the marginal distributions $\mathrm{p}(\mathbf{x})$ and $\mathrm{p}(\mathbf{g})$. If we assume the microscopic variables to be Gaussian, $\mathbf{x} \sim N(\mathbf{0}, \Sigma_x)$, so that

$$\mathrm{p}(\mathbf{x}) = \frac{1}{\sqrt{|2\pi\Sigma_x|}} \exp\left[-\frac{1}{2}\mathbf{x}^\intercal\Sigma_x^{-1}\mathbf{x}\right] \quad , \tag{5.27}$$

and similarly for the macroscopic variables, meaning that $\mathbf{g} \sim N(\mathbf{0}, \Sigma_g)$

$$\mathrm{p}(\mathbf{g}) = \frac{1}{\sqrt{|2\pi\Sigma_g|}} \exp\left[-\frac{1}{2}\mathbf{g}^\intercal\Sigma_g^{-1}\mathbf{g}\right] \quad . \tag{5.28}$$

and recall that the conditional density of the linear microscopic regression model in (5.6) is Gaussian

$$\mathrm{p}(\mathbf{g}|\mathbf{x}) = \frac{1}{\sqrt{|2\pi\Sigma_e|}} \exp\left[-\frac{1}{2}(\mathbf{g}-\mathbf{W}^\intercal\mathbf{x})^\intercal\Sigma_e^{-1}(\mathbf{g}-\mathbf{W}^\intercal\mathbf{x})\right] \quad , \tag{5.29}$$

it follows from (5.25) that $\mathrm{p}(\mathbf{g}|\mathbf{x})\mathrm{p}(\mathbf{x})$ is Gaussian as well;

$$\begin{aligned}&\mathrm{p}(\mathbf{g}|\mathbf{x})\mathrm{p}(\mathbf{x}) = \\ &\quad \frac{1}{\sqrt{|2\pi\Sigma_e|}}\frac{1}{\sqrt{|2\pi\Sigma_x|}} \exp\left[-\frac{1}{2}(\mathbf{g}-\mathbf{W}^\intercal\mathbf{x})^\intercal\Sigma_e^{-1}(\mathbf{g}-\mathbf{W}^\intercal\mathbf{x}) - \frac{1}{2}\mathbf{x}^\intercal\Sigma_x^{-1}\mathbf{x}\right] \quad . \end{aligned}\tag{5.30}$$

We obtain a similar expression for (5.26)

$$\begin{aligned}&\mathrm{p}(\mathbf{x}|\mathbf{g})\mathrm{p}(\mathbf{g}) = \\ &\quad \frac{1}{\sqrt{|2\pi\Sigma_\epsilon|}}\frac{1}{\sqrt{|2\pi\Sigma_g|}} \exp\left[-\frac{1}{2}(\mathbf{x}-\mathbf{B}^\intercal\mathbf{g})^\intercal\Sigma_\epsilon^{-1}(\mathbf{x}-\mathbf{B}^\intercal\mathbf{g}) - \frac{1}{2}\mathbf{g}^\intercal\Sigma_g^{-1}\mathbf{g}\right] \quad . \end{aligned}\tag{5.31}$$

Now, the two distributions (5.30) and (5.31) are identical via Bayes theorem, so we expand and compare their operands. Ignoring the constant factors the operand of (5.30) is

$$\begin{aligned} &(\mathbf{g}-\mathbf{W}^\intercal\mathbf{x})^\intercal\Sigma_e^{-1}(\mathbf{g}-\mathbf{W}^\intercal\mathbf{x}) + \mathbf{x}^\intercal\Sigma_x^{-1}\mathbf{x} &(5.32)\\ &\quad = \mathbf{g}^\intercal\Sigma_e^{-1}\mathbf{g} + (\mathbf{W}^\intercal\mathbf{x})^\intercal\Sigma_e^{-1}\mathbf{W}^\intercal\mathbf{x} - 2\mathbf{g}^\intercal\Sigma_e^{-1}\mathbf{W}^\intercal\mathbf{x} + \mathbf{x}^\intercal\Sigma_x^{-1}\mathbf{x} \quad . &(5.33)\end{aligned}$$

A similar expansion of (5.31) yields

$$\begin{aligned} &(\mathbf{x}-\mathbf{B}^\intercal\mathbf{g})^\intercal\Sigma_\epsilon^{-1}(\mathbf{x}-\mathbf{B}^\intercal\mathbf{g}) + \mathbf{g}^\intercal\Sigma_g^{-1}\mathbf{g} &(5.34)\\ &\quad = \mathbf{x}^\intercal\Sigma_\epsilon^{-1}\mathbf{x} + (\mathbf{B}^\intercal\mathbf{g})^\intercal\Sigma_\epsilon^{-1}\mathbf{B}^\intercal\mathbf{g} - 2\mathbf{g}^\intercal\mathbf{B}\Sigma_\epsilon^{-1}\mathbf{x} + \mathbf{g}^\intercal\Sigma_g^{-1}\mathbf{g} \quad . &(5.35)\end{aligned}$$

Finally, by comparing the terms in $\mathbf{x}$

$$\mathbf{x}^\intercal\Sigma_\epsilon^{-1}\mathbf{x} \quad \leftrightarrow \quad \mathbf{x}^\intercal\mathbf{W}\Sigma_e^{-1}\mathbf{W}^\intercal\mathbf{x} + \mathbf{x}^\intercal\Sigma_x^{-1}\mathbf{x} \quad , \tag{5.36}$$

we are able to identify

$$\Sigma_\epsilon^{-1} = \mathbf{W}\Sigma_e^{-1}\mathbf{W}^\mathsf{T} + \Sigma_x^{-1} \quad . \tag{5.37}$$

Further, the corresponding mixed term relationship

$$-2\mathbf{g}^\mathsf{T}\Sigma_e^{-1}\mathbf{W}^\mathsf{T}\mathbf{x} \quad \leftrightarrow \quad -2\mathbf{g}^\mathsf{T}\mathbf{B}\Sigma_\epsilon^{-1}\mathbf{x} \tag{5.38}$$

yields the identity

$$\mathbf{B} = \Sigma_e^{-1}\mathbf{W}^\mathsf{T}\Sigma_\epsilon \quad . \tag{5.39}$$

This shows that we, by assuming the marginal distributions $\mathrm{p}(\mathbf{x})$ and $\mathrm{p}(\mathbf{g})$ to be Gaussian, can compute the parameters of the GLM from the parameters of the linear microscopic regression model. The inverse relationships are easily found to be

$$\Sigma_e^{-1} = \mathbf{B}\Sigma_\epsilon^{-1}\mathbf{B}^\mathsf{T} + \Sigma_g^{-1} \tag{5.40}$$
$$\mathbf{W} = \Sigma_\epsilon^{-1}\mathbf{B}^\mathsf{T}\Sigma_e \quad . \tag{5.41}$$

5.5 Visualization

In the discussion about possible uses of successful models (which we can now identify as models with small generalization error) in section 2.2.2 we argued that insight into the function of the brain could potentially be gained by the identification of the features emphasized by such models.

With the proposed framework of conditional macroscopic density modeling in place we have a handle that allows us to interpret the somewhat vague notion of "features emphasized by the model"; they are the aspects of the microscopic density that affects the model's ability to approximate the macroscopic density. We must, in other words, identify the parts of input space that are used by the model. While these parts of signal space are always *contained* in model space[8] the two are not, in general, identical. That is to say, the information in some parts of model space may remain unused by the model. The linear microscopic regression models, however, is a special case; the estimated macroscopic variables are simply linear combinations of the microscopic observations as represented by the model space basis. So, for each element of the macroscopic vector we can identify an associated direction in signal space (and thus in input space) that fully identifies the parts of model space emphasized by the linear model. This direction forms a one-dimensional linear subspace of input space, which we shall refer to as the *projection space* of the linear model.

The simple projection space of linear models has the huge advantage of being something we can visualize; it is, in effect, merely a vector in input space. In general, though, projection space may be a nonlinear *manifold* in model space (Bell, 1990). This leads to visualization and interpretation problems, as we shall discover for the nonlinear models discussed in the next chapter.

[8]Recall that model space is defined as the space spanned by the subset of basis vectors (with which the microscopic observations are represented) that remain after non-salient parameters are pruned away during model complexity optimization.

5.5.1 Application to the CPH/SAC dataset

To demonstrate how the simple projection space of linear models facilitates model visualization figure 5.8 depicts the average emphasis of the twenty PCA-based, OBS-pruned two-parameter models with estimated optimal generalization ability (see also figure 5.4). The panels depict a transverse, a coronal and a sagittal slice, respectively. The objective

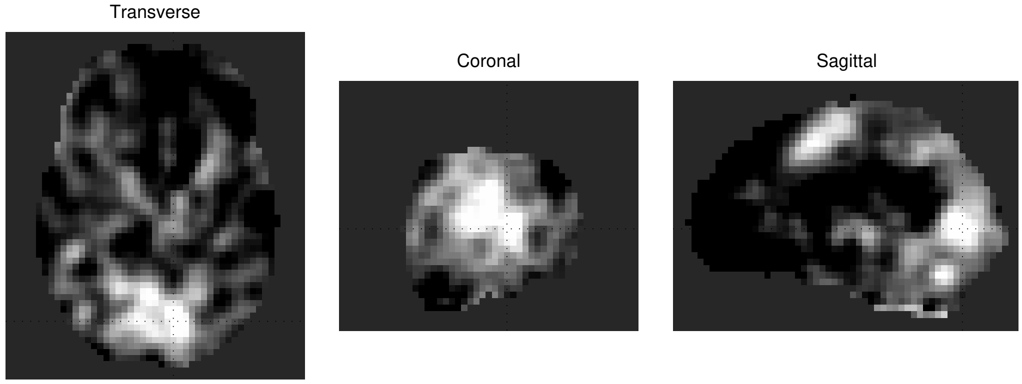

Figure 5.8: A transverse, a coronal, and a sagittal slice of the average linear one-dimensional projection space of two-parameter, OBS-optimized linear microscopic regression models based on the PCA representation of the CPH/SAC dataset. The large emphasized area in the back of the brain is the visual cortex.

of the present work is not so much to provide neuro-physiological insights, but rather to facilitate modeling and visualization based on a generalization theoretical framework. We therefore refrain from detailed interpretation of the current as well as later visualizations. Briefly, however, we note the relatively large emphasized area in the back of the brain. We identify this as the *visual cortex*; an area we clearly expect to be active during the performance of visual saccades. For further interpretation of linear model emphasis for the CPH/SAC dataset see e.g. (Mørch et al., 1996b; Law, 1997; Mørch and Thomsen, 1994; Lundsager and Kristensen, 1996).

5.6 Summary

We have exemplified the proposed generalization theoretical framework by providing estimates of the parameters of linear microscopic regression models which approximates the conditional macroscopic density. Further, the linear microscopic regression model has been proved analogous to the general linear model that approximates the conditional microscopic density; the latter is widely used in existing tools for analysis and modeling of functional datasets.

The dependency of model performance, as quantified by generalization error, on model flexibility and training set size has been demonstrated. For the restricted class of regularized linear models the proposed framework is shown to provide estimates of optimal model complexity; implicitly as the optimal regularization parameter value, and explicitly as the limited model space identified by OBS-based parameter pruning. Moreover, the estimated learning curves of model performance as functions of training set size emphasize the importance of matching model flexibility to the number of available observations;

large datasets may well support the application of relatively complex models.

Finally, linear models have been shown to emphasize a one-dimensional linear subspace of input space. This facilitates straightforward visualization of model emphasis.

Chapter 6

Nonlinear modeling

In the previous chapter we observed how model performance depends on model complexity. The investigated microscopic linear regression model is relatively inflexible by nature; no matter how we tweak the regularization and parameter configuration the model is constrained to linear first order relationships between the micro- and macroscopic variables. In an attempt to increase model performance beyond what is possible with simple linear models we examine a specific class of more flexible, nonlinear models.

6.1 Model basis functions

Consider models of which the output can be expressed as a linear combination of a set of model basis functions $z_j(\mathbf{x}, \mathbf{w})$

$$y(\mathbf{x}, \mathbf{w}) = \sum_{j=0}^{J} \tilde{w}_j z_j(\mathbf{x}, \mathbf{w}) \quad , \tag{6.1}$$

where the so-called *output parameters* $\tilde{w}_j$ are elements of the overall parameter vector $\mathbf{w}$. For convenience we use an index of zero to denote basis functions (and inputs) fixed to a value of one; the corresponding so-called *bias* parameters enable the mean of the model output to be non-zero, even for zero-mean basis functions.

The linear microscopic regression model investigated in chapter 5 is a special case of (6.1); the basis functions are simply the projection of the observations onto the signal space spanning basis. In the following we shall not regard such projections as part of the model. Rather, we use $\mathbf{x}$ to denote the microscopic variables regardless of the basis used to represent them. A fairly simple-minded extension of the linear regression model is to include polynomial terms of second order, i.e.

$$y(\mathbf{x}, \mathbf{w}) = \sum_{i_1=0}^{d} \sum_{i_2=0}^{d} w_{i_1 i_2} x_{i_1} x_{i_2} \quad . \tag{6.2}$$

Written out for the first two elements of the microscopic vector[1] ($d = 2$) equation (6.2)

[1] Or the first two elements of e.g. the vector of principal components, if the PCA basis is used to efficiently represent the microscopic vectors in signal space.

reads

$$
\begin{aligned}
y(\mathbf{x}, \mathbf{w}) &= \sum_{i_1=0}^{2} \sum_{i_2=0}^{2} w_{i_1 i_2} x_{i_1} x_{i_2} &\quad (6.3)\\
&= w_{00} + (w_{01} + w_{10})\, x_1 + (w_{02} + w_{20})\, x_2 &\quad (6.4)\\
&\quad + w_{11} x_1^2 + w_{22} x_2^2 + (w_{12} + w_{21})\, x_1 x_2 &\\
&= \sum_{j=0}^{5} \tilde{w}_j z_j(x_1, x_2) \quad . &\quad (6.5)
\end{aligned}
$$

We identify (6.5) as a special case of (6.1) in which all basis functions are *fixed*, i.e. independent of the model parameters $\mathbf{w}$. The simple linear regression model of the previous chapter includes only the zero and first order order terms of (6.4).

It is clear that model flexibility is increased by including terms of higher order. Consequently, a generalization of the polynomial model (6.2) to high order constitutes a very flexible model with potentially lower bias than—and improved generalization performance over—the simple first order linear microscopic regression model. It is important to note that the increased model flexibility does not complicate parameter estimation; the output is still a linear combination of fixed functions of the inputs so the parameters can be estimated analytically, exactly as in section 5.1.1.

6.1.1 The curse of dimensionality

From the discussion above a polynomial model seems a good choice when looking for increased model flexibility. However, there is no such thing as a free lunch, and the polynomial model *does* have a major drawback related to what is known as the *curse of dimensionality* (Bellman, 1961; Duda and Hart, 1973). The problem is easy to see; for an M'th order polynomial model with input vectors containing d elements the total number of adjustable parameters grows like

$$W \sim d^M \quad . \qquad (6.6)$$

To obtain a reasonably flexible model W must be huge. This is true even for the ill-posed datasets we are considering here; the projection of the microscopic variables onto a signal space spanning basis reduces the dimensionality of the model input vectors from d to N, but a very large number of parameters is still needed for all but simple first order models. To estimate the many parameters correctly, i.e. with low variance, a correspondingly large number of training set observations is required. Since observations are in short supply the poor dimensionality scaling of polynomial models constitutes a problem.

6.1.2 Adaptive basis functions

To obtain a model with flexibility that scales with the dimensionality of the input vectors better than (6.6) we reexamine the model expression (6.1). The polynomial model increases model flexibility by implementing a large number of fixed basis functions $z_j(\mathbf{x})$. This approach is hampered by the difficulties associated with choosing the basis functions in a way suitable for the modeling task at hand. Instead, we may let the basis functions themselves be parameterized, $z_j(\mathbf{x}, \mathbf{w})$, so that they can adapt to the observed data[2].

[2]Note that adaptive basis functions that are linear in $\mathbf{x}$ effectively result in a simple first order linear model, since the basis function parameters can be included in the output parameters $\tilde{w}_j$.

There are many possible ways to implement adaptive basis functions. One particular approach is that of *radial basis functions* (RBF) (Moody and Darken, 1989). Here, however, we shall focus on another adaptive basis function model, namely the multi-layer perceptron.

6.1.2.1 The multi-layer perceptron

The multi-layer perceptron (MLP) implements adaptive basis functions as monotonic (one-to-one), nonlinear transformation $g(\cdot)$ of signal space projections

$$z_j(\mathbf{x}, \mathbf{w}) = g\left(\sum_{i=0}^{d} w_{ij} x_i\right) = g(a_j) \quad . \tag{6.7}$$

The direction of the signal space projections are determined by the adaptive parameters w_{ji} , $i = 1, \ldots, d$. The total number of parameters for models based on (6.7) scales as $W \sim d$, which is better than for polynomial models. However, there is a price to pay; since the activation functions are nonlinear, the model itself is nonlinear in (some of) the parameters. This significantly complicates parameter estimation, as we shall discuss shortly.

Typically, activation functions are sigmoidal. Using the hyperbolic tangent, $g(a_j) = \tanh(a_j)$, the model obtained by combining (6.1) and (6.7) becomes

$$\begin{aligned} y(\mathbf{x}, \mathbf{w}) &= \sum_{j=0}^{J} \tilde{w}_j z_j(\mathbf{x}, \mathbf{w}) \\ &= \sum_{j=0}^{J} \tilde{w}_j \tanh\left(\sum_{i=0}^{d} w_{ij} x_i\right) \quad . \end{aligned} \tag{6.8}$$

This is the model we shall investigate in the remainder of this chapter. It can be regarded as a two-layer model; it consists of two separate layers of *processing units*, each of which computes a function of a linear combination of the output from units in the previous layer. This is visualized in figure 6.1, where the notion of *hidden units* is introduced for the processing units in the first layer. The model may be generalized to more than two layers,

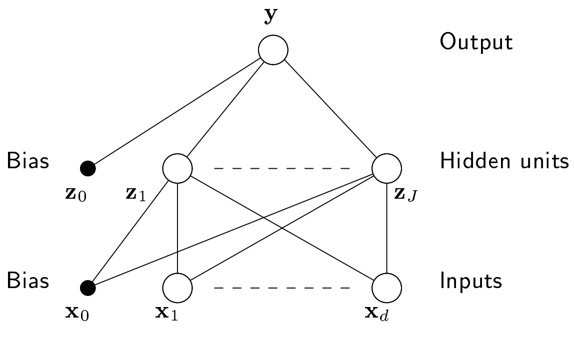

Figure 6.1: An MLP with two layers. The processing units in each layer feeds forward functions of linear combinations of the output from units in the previous layer. Note the fixed bias units which allow arbitrary mean outputs.

as implied by the first part, *multi-layer*, of the MLP name. The second part, *perceptron*, indicates the roots of models of this kind. The origin of the perceptron traces back to (Rosenblatt, 1962) and his work on networks of threshold units. The study of these single-layer networks was motivated by efforts to understand the computational workings of the brain at a neuronal level. This and related works gave rise to the field of *artificial neural networks* (ANN). In 1986 the rediscovery of an iterative parameter estimation scheme originally investigated by (Werbos, 1974), but now dubbed *back-propagation* sparked the field anew (Rumelhart et al., 1986). Since then progress has been great, as has the hype about the brain-like function of ANN models. The ANN field is, however, merely concerned with a certain class of parameter-efficient nonlinear models, and as such plays a role in statistics and mathematical modeling in line with more traditional disciplines.

The approximation capability of MLP's with sigmoidal activation functions has been studied intensely. An important result that can be found in various versions in the literature states that a two-layer perceptron can approximate arbitrarily well any continuous functional mapping from one finite-dimensional space to another, provided the number J of hidden units is sufficiently large (Cybenko, 1990; Hornik et al., 1990). It is our purpose in this chapter to investigate if the above result and the fact that the total number of model parameters scales better for MLP's than for polynomial models justifies the application of complex, nonlinear models in the analysis of functional datasets.

6.2 Parameter estimation

The two-layer perceptron (6.8) is nonlinear in the parameters $\mathbf{w}$, which means that it is impossible to analytically estimate the parameters that minimize the cost function. Instead we turn to iterative estimation procedures where the time-t estimate $\mathbf{w}_t$ is updated along a search direction $\Delta\mathbf{w}_t$

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \eta\Delta\mathbf{w}_t \quad . \tag{6.9}$$

The approach is outlined in figure 6.2, from which the problem of local optima is clear; while we are attempting to locate the global minimum $\mathbf{w}^*$ an iterative parameter estimation procedure may get stuck in the local minimum $\mathbf{w}^\circ$, depending on the initial parameter estimate $\mathbf{w}_0$. We shall not address this difficulty further here; for details on parameter initialization and optimization of MLP's and other ANN models (Bishop, 1995) is an excellent source. In the next sections we review some of the schemes for computing the parameter search direction $\Delta\mathbf{w}_t$ and determining a proper value of η.

6.2.1 First order optimization

The negative cost function gradient

$$\Delta\mathbf{w}_t = -\nabla\mathrm{C}(\mathbf{w}_t) \tag{6.10}$$

leads to the straightforward *gradient descent* optimization approach. A Taylor expansion to first order of the cost function around $\mathbf{w}_t$ evaluated at $\mathbf{w}_{t+1}$ yields

$$\begin{aligned} \mathrm{C}(\mathbf{w}_{t+1}) &= \mathrm{C}(\mathbf{w}_t) + \nabla\mathrm{C}(\mathbf{w}_t)^\mathsf{T}\Delta\mathbf{w}_t & (6.11)\\ &= \mathrm{C}(\mathbf{w}_t) - \eta\nabla\mathrm{C}(\mathbf{w}_t)^\mathsf{T}\nabla\mathrm{C}(\mathbf{w}_t) \quad , & (6.12) \end{aligned}$$

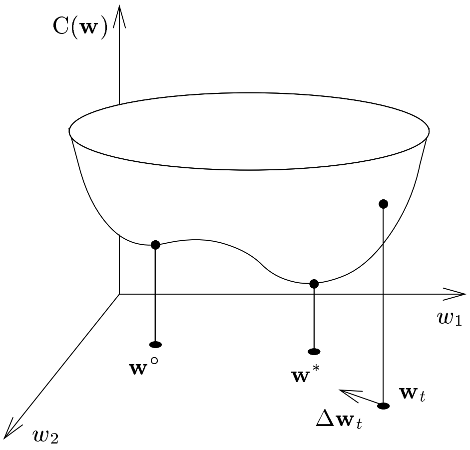

Figure 6.2: Iterative parameter estimation. The cost function can be regarded as a surface in parameter space of which we aim to find the global minimum $\mathbf{w}^*$. Depending on the initial parameter estimate we may end up in a local minimum $\mathbf{w}^\circ$.

which ensures that $\mathrm{C}(\mathbf{w})$ decreases for sufficiently small η. Consequently, $\mathbf{w}_t$ converges towards a local minimum $\mathbf{w}^\circ$. If the cost function is smooth and the parameter estimate is properly initialized, the global minimum $\mathbf{w}^*$ may be found[3].

For the two-layer perceptron with hyperbolic tangent activation functions in (6.8) the gradient of the regularized MSE cost function

$$\begin{aligned}
\nabla\mathrm{C}(\mathbf{w}) &= \nabla\mathrm{E}(\mathbf{w}) + \frac{1}{N}\nabla\mathrm{R}(\mathbf{w}) && (6.13)\\
&= \frac{1}{2N}\sum_{n=1}^{N}\frac{\partial\left(y(\mathbf{x}_n,\mathbf{w}) - g_n\right)^2}{\partial\mathbf{w}} + \frac{1}{N}\mathbf{R}\mathbf{w} && (6.14)\\
&= \frac{1}{N}\sum_{n=1}^{N}\frac{\partial y(\mathbf{x}_n,\mathbf{w})}{\partial\mathbf{w}}\cdot\left(y(\mathbf{x}_n,\mathbf{w}) - g_n\right) + \frac{1}{N}\mathbf{R}\mathbf{w} && (6.15)
\end{aligned}$$

is easily computed from the first layer derivatives

$$\frac{\partial y(\mathbf{x},\mathbf{w})}{\partial w_{ij}} = \tilde{w}_j\left(1 - z_j(\mathbf{x},\mathbf{w})^2\right)x_i \qquad (6.16)$$

and their second layer counterparts

$$\frac{\partial y(\mathbf{x},\mathbf{w})}{\partial \tilde{w}_j} = z_j(\mathbf{x},\mathbf{w}) \quad . \qquad (6.17)$$

6.2.1.1 Gradient computation by error back-propagation

Straightforward gradient computation via (6.16) and (6.17) requires W calculations for each of the first layer derivatives, leading to a total computational effort that scales as W^2. A much more efficient way of calculating the derivatives is obtained by employing *error back-propagation* (Werbos, 1974; Rumelhart et al., 1986).

[3]However, there is no way of distinguishing a local from the global minimum once the iterative procedure has converged.

While back-propagation lessens the computational burden of gradient evaluation and thus parameter estimation it shall not be a topic of further investigation here; the above references as well as ANN textbooks like (Bishop, 1995) discuss it in detail. Suffice it to say that the approach rests on a chain rule decomposition of the cost function derivate, which facilitates the computation of the gradient by storing local error measures for each unit.

6.2.2 Second order optimization

The convergence of gradient descent can be very slow. Consequently, a large number of refinements to the algorithm have been proposed, all aiming to improve convergence speed. Rather than investigating any of these in detail, we shall explore how information of second order derivatives may improve optimization. The Taylor expansion to second order of the regularized cost function around the global minimum $\mathbf{w}^*$ yields[4]

$$\mathrm{C}(\mathbf{w}) = \mathrm{C}(\mathbf{w}^*) + \frac{1}{2}(\mathbf{w}-\mathbf{w}^*)^{\mathsf{T}}\mathbf{J}(\mathbf{w}-\mathbf{w}^*) \quad , \tag{6.18}$$

since the first order derivative is zero. The Hessian matrix $\mathbf{J}$ consists of second order derivatives. The gradient of the expansion

$$\nabla\mathrm{C}(\mathbf{w}) = \mathbf{J}(\mathbf{w}-\mathbf{w}^*) \tag{6.19}$$

leads to *Newton's formula*

$$\mathbf{w}^* = \mathbf{w} - \mathbf{J}^{-1}\nabla\mathrm{C}(\mathbf{w}) \quad . \tag{6.20}$$

Provided that the set of parameters $\mathbf{w}$ is close to the global minimum so that the second order Taylor expansion constitutes a good approximation to the cost function surface, a Newton step of

$$\Delta\mathbf{w}_t = -\left(\mathbf{J}|_{\mathbf{w}=\mathbf{w}_t}\right)^{-1}\nabla\mathrm{C}(\mathbf{w}_t) \quad , \tag{6.21}$$

with $\eta = 1$, moves $\mathbf{w}$ very close to the global minimum. In practice the Taylor expansion is less than perfect so we reduce the step-size η—in fact, a one-dimensional line search along the parameter search direction is often employed in an attempt to find an optimal step-size.

6.2.2.1 Levenberg-Marquardt approximation

It is complicated and computationally expensive to compute the second order derivatives exactly. Even-though it can be done, see e.g. (Buntine and Weigend, 1994), approximations hold a number of advantages as we shall see next.

The cost function consists of two terms; the unregularized cost function and the regularizer. The first and second order derivatives of the latter are straightforward to compute

$$\mathrm{R}(\mathbf{w}) = \frac{1}{2}\mathbf{w}^{\mathsf{T}}\mathbf{R}\mathbf{w} \qquad \Rightarrow \qquad \frac{\partial\mathrm{R}(\mathbf{w})}{\partial\mathbf{w}} = \mathbf{R}\mathbf{w} \quad , \quad \frac{\partial^2\mathrm{R}(\mathbf{w})}{\partial\mathbf{w}\partial\mathbf{w}^{\mathsf{T}}} = \mathbf{R} \quad . \tag{6.22}$$

[4] The following is true for any optimum, meaning that the procedure may identify both minima and maxima, local as well as global. Therefore the parameters must be close to the (global) minimum before the approach is employed.

Concentrating therefore on the Hessian matrix $\mathbf{H}$ of the unregularized cost function $\mathrm{E}(\mathsf{D}, \mathbf{w})$ we apply the chain-rule to yield the gradient

$$\begin{aligned}\frac{\partial \mathrm{E}(\mathsf{D}, \mathbf{w})}{\partial \mathbf{w}} &= \frac{1}{2N} \sum_{n=1}^{N} \frac{\partial (y(\mathbf{x}_n, \mathbf{w}) - g_n)^2}{\partial \mathbf{w}} \qquad (6.23)\\ &= \frac{1}{N} \sum_{n=1}^{N} \frac{\partial y(\mathbf{x}_n, \mathbf{w})}{\partial \mathbf{w}} \cdot (y(\mathbf{x}_n, \mathbf{w}) - g_n) \quad . \qquad (6.24)\end{aligned}$$

The two factors lead to the decomposition of the corresponding Hessian matrix of second order derivatives

$$\begin{aligned}\mathbf{H} &= \frac{\partial^2 \mathrm{E}(\mathsf{D}, \mathbf{w})}{\partial \mathbf{w} \partial \mathbf{w}^\top} \qquad (6.25)\\ &= \frac{1}{N} \sum_{n=1}^{N} \left[\frac{\partial y(\mathbf{x}_n, \mathbf{w})}{\partial \mathbf{w}} \cdot \frac{\partial y(\mathbf{x}_n, \mathbf{w})}{\partial \mathbf{w}^\top} + \frac{\partial^2 y(\mathbf{x}_n, \mathbf{w})}{\partial \mathbf{w} \partial \mathbf{w}^\top} \cdot (y(\mathbf{x}_n, \mathbf{w}) - g_n) \right] \quad . \qquad (6.26)\end{aligned}$$

Consider now the limit of many training set observations, in which training error is defined as generalization error. In (4.83) generalization error was decomposed into two terms, repeated here for convenience

$$\begin{aligned}\mathrm{G}_{\mathrm{MSE}}(\mathsf{D}, \mathbf{w}) &= \frac{1}{2} \int (y(\mathbf{x}, \mathbf{w}) - \langle g|\mathbf{x}\rangle)^2 \, \mathrm{p}(\mathbf{x}) \, \mathrm{d}\mathbf{x} \\ &\quad + \frac{1}{2} \int \left(\langle g^2|\mathbf{x}\rangle - \langle g|\mathbf{x}\rangle^2 \right) \mathrm{p}(\mathbf{x}) \, \mathrm{d}\mathbf{x} \quad . \qquad (6.27)\end{aligned}$$

Only the first term of (6.27) depends on the model parameters, meaning that the expected Hessian matrix can be expressed in terms of the conditional average, $\langle g|\mathbf{x}\rangle$, of the system output

$$\begin{aligned}&\int \frac{\partial^2 \mathrm{E}(\mathsf{D}, \mathbf{w})}{\partial \mathbf{w} \partial \mathbf{w}^\top} \mathrm{p}(\mathbf{x}) \, \mathrm{d}\mathbf{x} \qquad (6.28)\\ &= \int \left[\frac{\partial y(\mathbf{x}, \mathbf{w})}{\partial \mathbf{w}} \cdot \frac{\partial y(\mathbf{x}, \mathbf{w})}{\partial \mathbf{w}^\top} + \frac{\partial^2 y(\mathbf{x}, \mathbf{w})}{\partial \mathbf{w} \partial \mathbf{w}^\top} \cdot (y(\mathbf{x}, \mathbf{w}) - \langle g|\mathbf{x}\rangle) \right] \mathrm{p}(\mathbf{x}) \, \mathrm{d}\mathbf{x} \quad . \qquad (6.29)\end{aligned}$$

If the parameters are close the global minimum, $\mathbf{w} \simeq \mathbf{w}^*$ the model is close to the conditional system output, $y(\mathbf{x}, \mathbf{w}) \simeq \langle g|\mathbf{x}\rangle$, according to (4.84). Consequently, the last term in (6.29) will be small. So, for parameters close to the global minimum, model error is small on average, justifying that the second term of (6.26) be ignored (Hassibi and Stork, 1992). The result is the *Levenberg-Marquardt* (LM) approximation (Marquardt, 1963)

$$\mathbf{H} \simeq \frac{1}{N} \sum_{n=1}^{N} \frac{\partial y(\mathbf{x}_n, \mathbf{w})}{\partial \mathbf{w}} \cdot \frac{\partial y(\mathbf{x}_n, \mathbf{w})}{\partial \mathbf{w}^\top} \quad , \qquad (6.30)$$

also referred to as the outer-product approximation. The LM approximated Hessian holds two major advantages; firstly, it involves only first order derivative information, meaning that it is computationally relatively inexpensive to employ. Secondly, when used in the second order iterative optimization procedure of (6.21) it will always lead to a cost function decrease, since it is guaranteed to be positive semi-definite.

6.2.2.2 Diagonal approximation

A computationally even less expensive Hessian approximation is achieved by ignoring the off-diagonal terms in the LM approximation (Le Cun, Y. et al., 1990). For a diagonal regularizer $\mathbf{R} = \operatorname{diag}[\alpha_i]$ the resulting iterative updating scheme, here written for a single parameter i,

$$\Delta \mathbf{w}_{i,t} = -\left(\mathbf{J}|_{\mathbf{w}=\mathbf{w}_t}\right)^{-1}_{ii} (\nabla \mathrm{C}(\mathbf{w}_t))_i \simeq -N \frac{(\nabla \mathrm{C}(\mathbf{w}_t))_i}{(\nabla y(\mathbf{x}, \mathbf{w}_t))_i^2 + \alpha_i} \quad , \tag{6.31}$$

is called *pseudo Gauss Newton* optimization.

6.2.3 Example

To exemplify the iterative approach above figure 6.3 depicts the evolution of training error during parameter estimation for one particular nonlinear model of the saccade frequency for the CPH/SAC dataset. An MLP with three hidden units was used, resulting in a

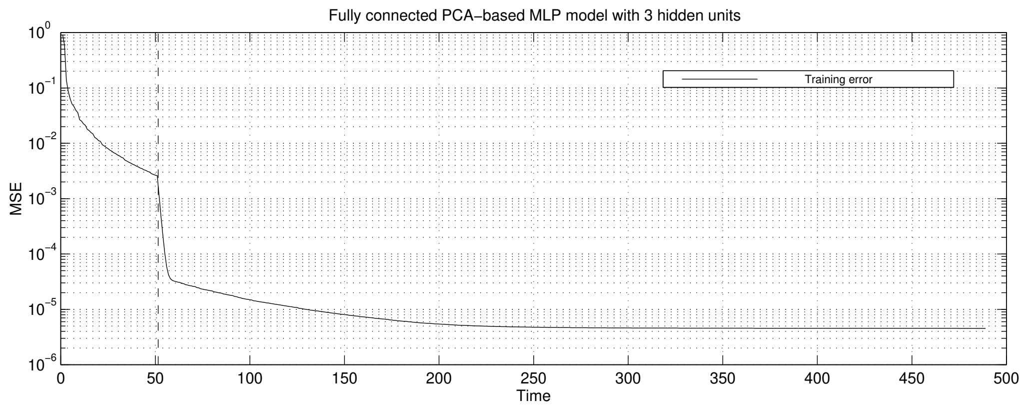

Figure 6.3: Evolution of the MSE training error during iterative parameter estimation for one particular MLP model on the CPH/SAC dataset. After 51 first order iterations (dashed line), a second Newton method speeds up convergence.

total of $W = 196$ parameters. The first few iterations (to the left of the dashed vertical line) are performed using gradient descent, for which convergence quickly becomes very slow. After 51 iterations a second order Newton method is employed based on the LM approximation of the Hessian. A sharp increase in convergence speed results. After some 500 iterations convergence is again slow, and iteration is stopped; the resulting parameters are subsequently used as an estimate of $\mathbf{w}^*$. For details on issues such as parameter initialization and iteration stopping criteria, see e.g. (Bishop, 1995).

6.3 Complexity control

As discussed in chapter 4, and investigated for the linear models in the previous chapter, flexibility and training set size affect model generalization performance. So too for nonlinear MLP models. Recalling the earlier discussion of the in-feasibility of sampling more than a few points in the space of possible combinations of model complexity and training set size, the computationally expensive iterative parameter estimation needed for MLP optimization emphasizes the following results as illustrative rather than conclusive.

6.3.1 Regularization

In the context of ANN's the simple diagonal regularizer $\mathrm{R}(\mathbf{w}) = \frac{1}{2}\mathbf{w}^\intercal\mathbf{R}\mathbf{w}$ is often called *weight decay* (WD)[5], see also (Hinton, 1986). The name results from the regularization term's ability to force parameters towards zero, as we saw in section 4.4.1.

Since the second order derivative

$$\frac{\partial^2 \mathrm{R}(\mathbf{w})}{\partial\mathbf{w}\partial\mathbf{w}^\intercal} = \mathbf{R} \tag{6.32}$$

is simply the diagonal regularization matrix itself, the regularized Hessian is simple to compute. In (6.31) we saw how simple WD combined with a diagonal Hessian approximation provides for a particularly simple second order optimization scheme.

6.3.2 Parameter pruning

Parameter pruning from saliency measures based on the estimated training error increase was explained in section 4.4.2.2, where the general OBS expression as well as the OBD approximation based on a diagonal Hessian approximation were derived. These may be directly applied to the MLP's considered here.

6.4 Application to the CPH/SAC dataset

In analogy to the linear models of chapter 5 the nonlinear two-layer perceptron was applied to the CPH/SAC dataset in order to investigate the effects of training set size and model complexity. Three hidden units were included in all models discussed in the following. The same bootstrap samples that were used to investigate the linear models were applied to the nonlinear models, meaning that the same test set of $N_{\mathsf{T}} = 16$ observations was left out for empirical generalization error assessment. The remaining 48 observations yielded $M = 20$ training sets of size $N = 48$, as before.

6.4.1 Complexity control

Model flexibility was varied in order to investigate it's influence on estimated generalization error. Again, both regularization and parameter pruning was employed to control model complexity.

6.4.1.1 Regularization

The nonlinear MLP models were regularized using one common regularization parameter. Several regularization parameters are often used in practice, e.g. one for each layer of parameters, or even one for each model parameter. Due to the complexity involved with exploring the multidimensional regularization space spanned by more than a single regularization parameter we here investigate generalization performance dependency for one common parameter only.

Figure 6.4 depicts the results for the microscopic vectors represented using the PCA basis. For both small and large values of α the empirical generalization error estimate

[5]In ANN's the parameters are also called weights.

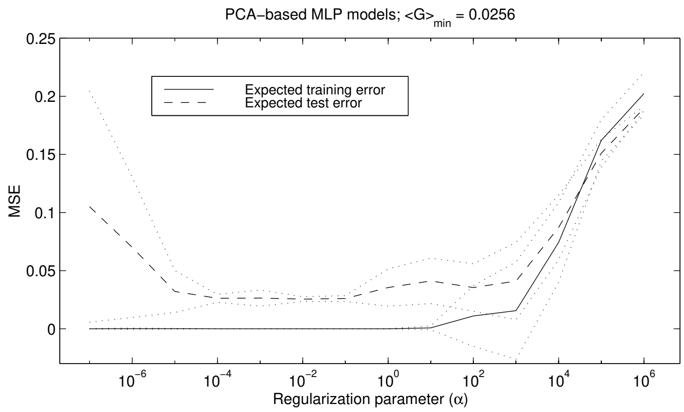

Figure 6.4: The expected generalization error (test error) as function of the regularization parameter for PCA-based nonlinear MLP models of the saccade frequency for the CPH/SAC dataset. The dotted lines represent one standard deviation of the test error estimate. A medium valued regularization parameter appears to yield optimal performance.

is large; only for values in the middle of the investigated range is model performance good. The minimum test error of 0.0256 is achieved for $\alpha = 0.01$. Comparing the figure to the corresponding figure 5.1 for the linear models, the difference for small values of the regularization parameters is obvious; in contrast to the linear models, the nonlinear MLP's are flexible to such a degree that they become overly sensitive to the training set observations unless some amount of regularization is applied. Conversely, for very large values of α the regularization introduces so much model bias that generalization performance suffers. The two effects combine to produce the observe generalization error minimum.

We also observe how the minimum estimated expected generalization error is smaller for the nonlinear MLP models than for the linear models; in chapter 5 the corresponding test error evaluated to 0.0509. The increased model performance can be accounted to the increased model flexibility of the MLP model over it's linear counterpart.

For models based on the ICA basis representation of the microscopic observations the situation is very similar, as shown in figure 6.5. The minimum estimated generalization error of 0.0240 occurs for $\alpha = 0.001$, but as before model performance is close to identical over a range of intermediate regularization parameter values. Only for very small or very large values is model performance significantly affected. The significant average test error difference between optimally regularized, fully connected PCA and ICA models that we observed in the linear case is not reflected to the same degree in figures 6.4 and 6.5; estimated generalization performance is still better for the fully connected ICA-based models, but only slightly. It means that the increased flexibility of the MLP models is better utilized when representing the microscopic observations using the PCA than the ICA basis. The nonlinear nature of the MLP models does not, in other words, render the independent basis vectors more informative—at least not for fully connected models.

6.4.1.2 Parameter pruning

We proceed to control model flexibility more directly by employing parameter pruning. Due to the large number of parameters in the fully connected MLP models, a pruning

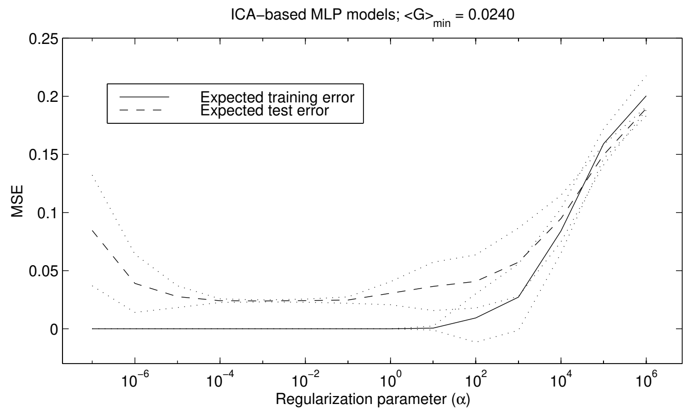

Figure 6.5: The expected generalization error (test error) as function of the regularization parameter for ICA-based nonlinear MLP models of the saccade frequency for the CPH/SAC dataset. The dotted lines represent one standard deviation of the test error estimate. As for the PCA-based models a regularization parameter in the middle of the range of investigated values yields optimal performance.

scheme is adopted in which several parameters are removed simultaneously. More specifically a small fraction of the remaining parameters are removed, meaning that while quite a few parameters are removed simultaneously from large models, finer complexity control is achieved for smaller models. To this end the OBD rather than the OBS parameter saliency estimate is used; experience indicate that while OBD and OBS based parameter pruning often perform comparably, situations sometimes arise in which OBS significantly underestimates parameter saliency, leading to sudden increases in generalization error (Pedersen et al., 1995; Pedersen, 1997).

Figure 6.6 reproduces the evolution of model performance for the PCA-based models as parameters are pruned. The regularization parameter was set to $\alpha = 0.01$ as for the linear models. This value yields close to optimal performance on the fully connected model, as seen from figure 6.4. The error-bars represent one standard deviation of the test error, and at the same time serve as indications of the fractional pruning scheme that was adopted; the number of parameters eliminated at one time decreases as the models get smaller.

Figure 6.7 is identical to figure 6.6 except for the error-bars. This aids identification of the test error minimum which occurs for models with 5 parameters. The average generalization performance for these models evaluates to 0.0140, compared to the value of 0.0172 yielded by the two-parameter linear PCA-based models from figure 5.4. The slight error decrease reveals how the flexibility of the MLP's provides for a small performance increase. The difference is not very large, however, and is from figures 6.6 and 6.7 evidenced only for training sets of size 48. We shall return to this issue in section 6.4.2.

For the nonlinear MLP models based on the ICA representation of the microscopic observations the picture is similar, as seen in figures 6.8 and 6.9. However, the estimated optimal model is relatively complex with it's 186 parameters. The situation closely reflects the linear case where almost all independent projections were likewise retained. The similarity extends to generalization performance in that the ICA-based nonlinear MLP models yield higher average test error, namely 0.0240, than PCA-based models with 5 parameters which average to a value of 0.0140. The interpretation is straightforward;

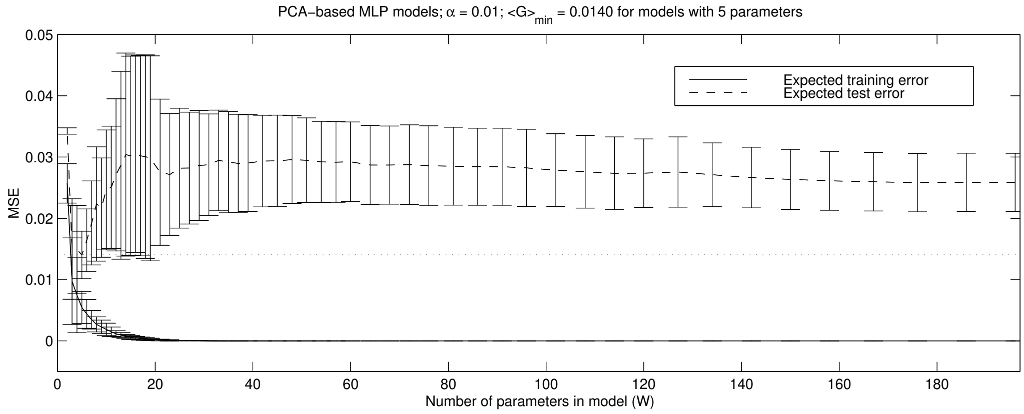

Figure 6.6: The estimated expected generalization error as function of the number of parameters for PCA-based nonlinear MLP models of the saccade frequency for the CPH/SAC dataset. The regularization parameter is $\alpha = 0.01$. The empirical generalization error estimate predicts model performance to be optimal for models with a relatively small number of parameters. Error-bars represent one standard deviation of the error estimates.

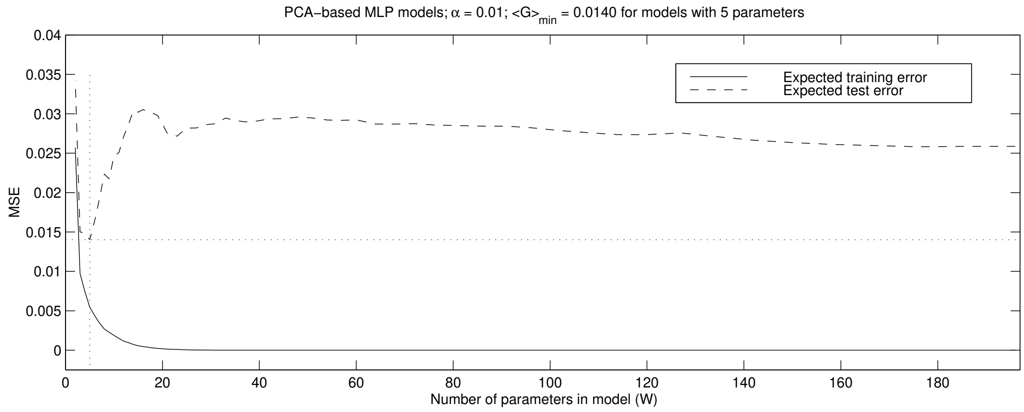

Figure 6.7: The estimated expected generalization error as function of the number of parameters for PCA-based nonlinear MLP models of the saccade frequency for the CPH/SAC dataset. The regularization parameter is $\alpha = 0.01$. The empirical generalization error estimate predicts model performance to be optimal for models with 5 parameters.

an ICA representation of the microscopic observations seems less informative than the corresponding PCA representation. This finding may be the result of the in-feasibility of a linear mixture model for human brain functions, shortcomings in the adopted ICA entropy maximization scheme[6], or issues related specifically to the investigated CPH/SAC dataset. In any event, the applicability of ICA for efficient representation of functional datasets needs to be studied further.

[6]Such shortcomings may lead to the localization of a local rather than the global maximum.

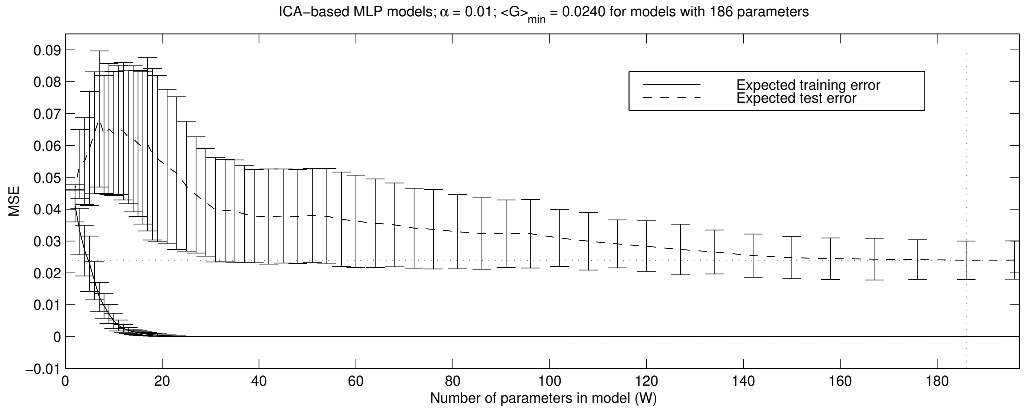

Figure 6.8: The estimated expected generalization error as function of the number of parameters for ICA-based nonlinear MLP models of the saccade frequency for the CPH/SAC dataset. The regularization parameter is $\alpha = 0.01$. The empirical generalization error estimate predicts model performance to be optimal for relatively large models.

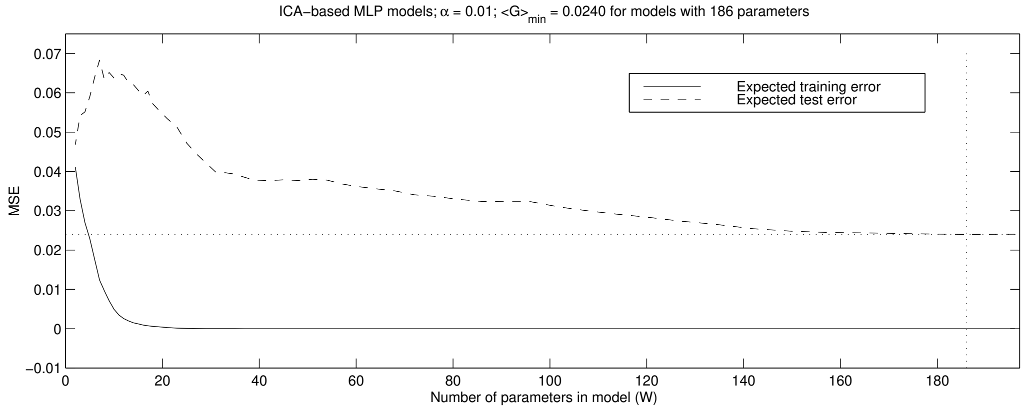

Figure 6.9: The estimated expected generalization error as function of the number of parameters for ICA-based nonlinear MLP models of the saccade frequency for the CPH/SAC dataset. The regularization parameter is $\alpha = 0.01$. The empirical generalization error estimate predicts model performance to be optimal for models with 186 parameters.

6.4.2 Learning curves

To quantify the effect of training set size on model performance the empirical expected generalization error for models based on training sets of increasing size were computed. The bootstrap samples with sizes ranging from 10 to 100 that were generated for the linear models were once again utilized. Regularization was fixed to $\alpha = 0.01$ in accordance with findings above.

The five-parameter PCA-based MLP models result in the learning curve depicted in figure 6.10. We observe how the large generalization error that results from small training sets decreases as more observations become available. For training sets containing more than some fifty observations, i.e. the same size as the training set used in the parameter prunings above, no performance improvement occurs; test error settles at a level very close

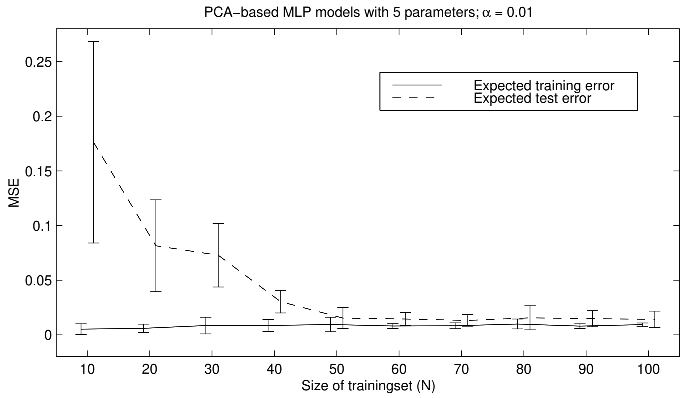

Figure 6.10: Empirical learning curve for PCA-based nonlinear MLP models for the CPH/SAC dataset. The error-bars represent one standard deviation of the error estimates. As the number of training set observations increases the generalization performance improves.

to the 0.0140 achieved by the five-parameter MLP models based on a training set size of 48.

The nonlinear MLP learning curve is very similar to it's linear two-parameter equivalent. Figure 6.11 combines the two curves in one plot. Performance is close to identical for

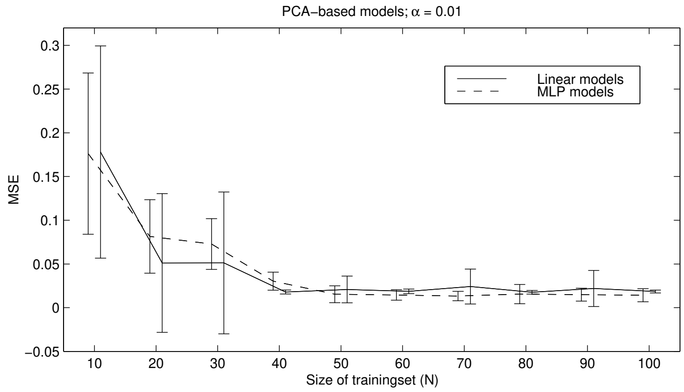

Figure 6.11: Empirical learning curves for PCA-based linear and nonlinear models for the CPH/SAC dataset. As the number of training set observations increases the generalization performance improves for both model types in a very similar manner. Only for large training sets does a small, but essentially insignificant, difference manifest itself.

practically all training set sizes. A small, but essentially insignificant, difference is barely visible for large set sizes. So, despite the small test error difference between the linear two-parameter models averaging to 0.0172 and the nonlinear five-parameter MLP models averaging to 0.0140 when based on training sets containing 48 observations, we must conclude that the low-dimensional model spaces identified by the corresponding principal axis only to a very low degree support the application of flexible nonlinear models over simpler linear ones.

For the more heavily parameterized ICA-based models the picture looks only slightly different. The learning curve of the MLP models, depicted in figure 6.12 and repeated together with the corresponding linear model learning curve in figure 6.13, settles at a level slightly lower than that of the linear learning curve. This again indicates very little

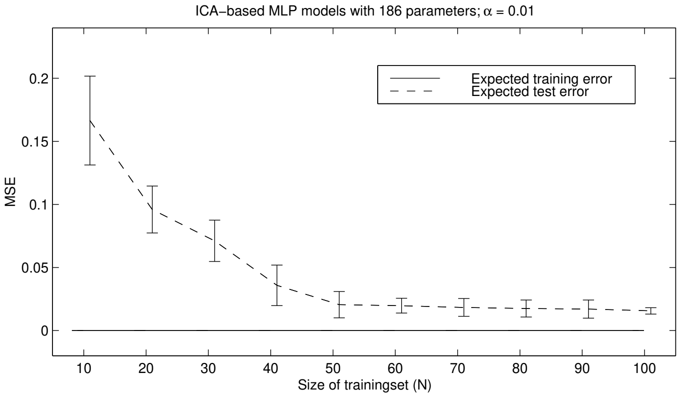

Figure 6.12: Empirical learning curve for ICA-based nonlinear MLP models for the CPH/SAC dataset. More observations yield better performance.

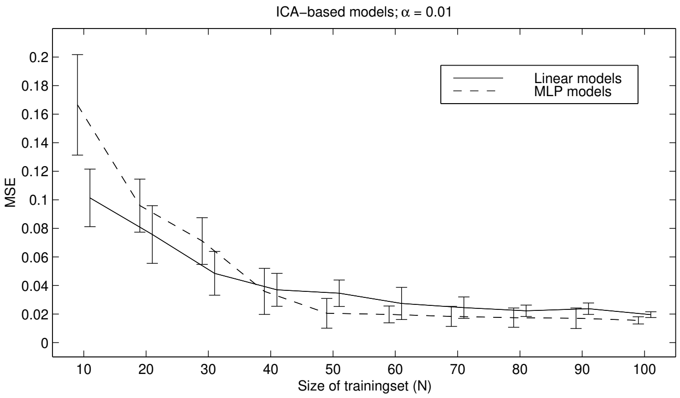

Figure 6.13: Empirical learning curves for ICA-based linear and nonlinear models for the CPH/SAC dataset. Larger training sets increase the generalization performance for both model types. The small difference between the linear and nonlinear models is largest for medium sized training sets.

support for the application of flexible nonlinear models on this particular dataset. The situation may be different, however, for datasets from other functional experiments.

6.4.2.1 Other learning curve examples

As examples of a more clear learning curve or generalization cross-over, we include figures 6.14 and 6.15. They reproduce figures from the paper (Mørch et al., 1997), which appears as appendix F. Figure 6.14 reproduces the empirical learning curves for PCA-based linear and nonlinear classifiers applied to a categorically designed functional PET

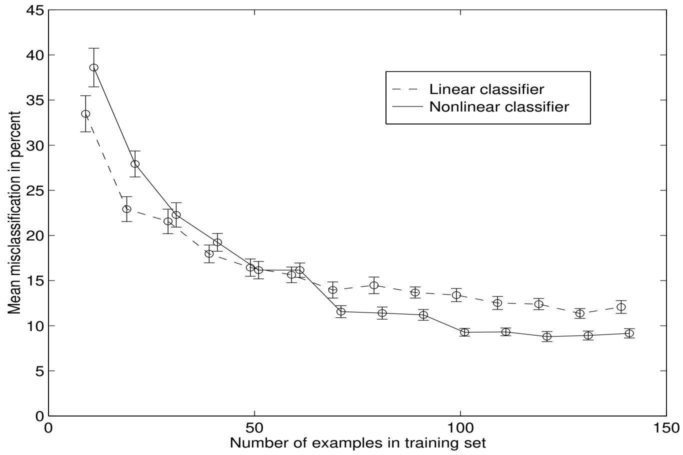

Figure 6.14: Empirical learning curves for PCA-based linear and nonlinear classifiers applied to a categorically designed functional PET experiment involving a simple finger opposition task. Generalization cross-over occurs for medium sized training sets.

experiment involving a simple finger opposition task[7]. For this dataset the linear classifiers seem optimal for small datasets, while the availability of more observations eventually clearly warrants to use of the more flexible nonlinear classifier.

The learning curves for an fMRI experiment involving a finger-to-thumb opposition task are depicted in figure 6.15. Again linear and nonlinear classifiers were employed,

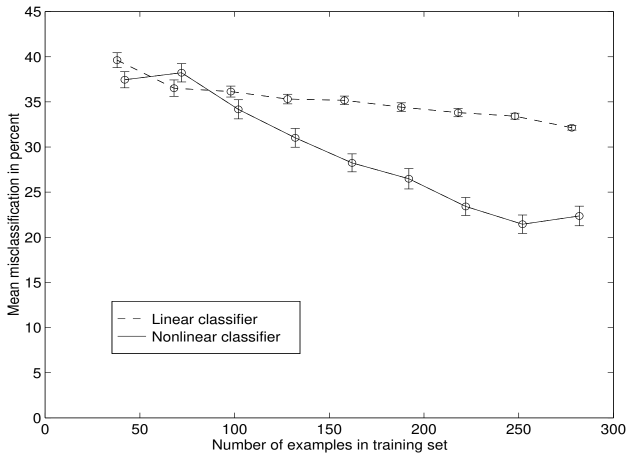

Figure 6.15: Empirical learning curves for PCA-based linear and nonlinear classifiers applied to a categorically designed functional fMRI experiment involving a finger-to-thumb opposition task. Generalization cross-over occurs for very small training sets.

but for this dataset the picture is different; generalization cross-over occurs for relatively small datasets. As the number of training set observations increases the nonlinear models significantly outperform their linear counterparts.

[7]For a more detailed description of the experimental design refer to the paper (Mørch et al., 1997).

6.5 Visualization

Having obtained a nonlinear MLP that appears to perform well it is of interest to investigate which parts of signal space it emphasizes. However, the only thing we can visualize is a direction of input space, i.e. a one-dimensional linear subspace. This complicates MLP visualization since an MLP with three hidden units effectively operates nonlinearly from projections onto the three-dimensional model space spanned by the parameter vectors of the three hidden units. It means that model emphasis is hard to quantify, resulting from a nonlinear manifold in model space.

Referring back to section 5.4.1 where we investigated the relationship between the linear microscopic regression model and the GLM which models the expected microscopic vector from a set of macroscopic variables, we recall how it was possible to derive simple expressions relating the parameters of the two models. In analogy, a natural approach to visualization of nonlinear microscopic regression models, of which the MLP as described in this chapter is an example, would be to attempt to relate the estimated conditional macroscopic model average $\langle g|\mathbf{x}\rangle$ to the conditional *microscopic* density via Bayes theorem. By obtaining an estimate of this density we could compute one-dimensional properties suitable for visualization. An obvious candidate would be the conditional microscopic average $\langle \mathbf{x}|g\rangle$. Encouraging preliminary investigations into this approach appear in (Lundsager and Kristensen, 1996).

6.5.1 The saliency map

In this presentation we shall limit ourselves to consider the so-called *saliency map*, which we proposed in (Mørch et al., 1995); the paper appears as appendix D herein. While the approach targets the identification of model emphasis and as such facilitates MLP visualization, it suffers from some problems. These will be discussed at the end of this section.

In close analogy to the saliency measures of OBD and OBS the idea behind the saliency map is to estimate the increase in training error that results from the elimination of one particular voxel. Assume that the microscopic observations are represented by the basis $\mathbf{E} = [\mathbf{e}_1 \ \cdots \ \mathbf{e}_E]$ consisting of E basis vectors. We label the corresponding microscopic projections $\mathbf{v} = \mathbf{E}^\mathsf{T}\mathbf{x}$. In the following we assume the microscopic vectors to have zero mean. The model depends on the microscopic vectors and thus on the basis used to represent them, so we write

$$y(\mathbf{E}^\mathsf{T}\mathbf{x}, \mathbf{w}) = \sum_{j=0}^{J} \tilde{w}_j \tanh\left(\mathbf{w}_j^\mathsf{T}\mathbf{E}^\mathsf{T}\mathbf{x}\right) \quad , \tag{6.33}$$

where the summation over microscopic vector elements is written as an inner product. We now define the saliency of the i'th voxel as the increase in training error resulting from the elimination of that voxel. Letting $\mathbf{E}^i$ denote the basis where the i'th voxel is removed, i.e.

$$e^i_{e,i'} = \begin{cases} e_{e,i'} & i' \neq i \\ 0 & i' = i \end{cases} \quad , \tag{6.34}$$

and $\mathbf{w}^i$ the corresponding set of optimal model parameters, the saliency can be written as

$$\delta \mathrm{E}_i = \mathrm{E}(\mathsf{D}, \mathbf{w}^i, \mathbf{E}^i) - \mathrm{E}(\mathsf{D}, \mathbf{w}, \mathbf{E}) \quad . \tag{6.35}$$

6.5.1.1 Approximating the saliency map

Straightforward computation of the saliency map is extremely computationally expensive since it involves estimation of d different sets of model parameters, where d is the total number of voxels. Instead we employ an approximation.

A Taylor expansion of the cost function to second order with respect to the basis vectors and the parameter vector yields

$$\delta \mathrm{E} \simeq \sum_{e=1}^{E} \frac{\partial \mathrm{E}}{\partial \mathbf{e}_e^{\top}} \delta \mathbf{e}_e + \frac{\partial \mathrm{E}}{\partial \mathbf{w}^{\top}} \delta \mathbf{w} \tag{6.36}$$

$$+\frac{1}{2} \sum_{e=1}^{E} \delta \mathbf{e}_e^{\top} \frac{\partial^2 \mathrm{E}}{\partial \mathbf{e}_e \partial \mathbf{e}_e^{\top}} \delta \mathbf{e}_e + \frac{1}{2} \delta \mathbf{w}^{\top} \frac{\partial^2 \mathrm{E}}{\partial \mathbf{w} \partial \mathbf{w}^{\top}} \delta \mathbf{w} + \sum_{e=1}^{E} \delta \mathbf{e}_e^{\top} \frac{\partial^2 \mathrm{E}}{\partial \mathbf{e}_e \partial \mathbf{w}^{\top}} \mathbf{w} \quad . \tag{6.37}$$

If the model parameters have been successfully estimated then the second term in (6.36) is zero. We may further ignore all terms involving the parameter change $\delta \mathbf{w}$ as argued in (Mørch et al., 1995), to obtain

$$\delta \mathrm{E} \simeq \sum_{e=1}^{E} \frac{\partial \mathrm{E}}{\partial \mathbf{e}_e^{\top}} \delta \mathbf{e}_e + \frac{1}{2} \sum_{e=1}^{E} \delta \mathbf{e}_e^{\top} \frac{\partial^2 \mathrm{E}}{\partial \mathbf{e}_e \partial \mathbf{e}_e^{\top}} \delta \mathbf{e}_e \quad , \tag{6.38}$$

effectively disregarding parameter re-optimization, i.e. $\delta \mathrm{E}_i = \mathrm{E}(\mathsf{D}, \mathbf{w}, \mathbf{E}^i) - \mathrm{E}(\mathsf{D}, \mathbf{w}, \mathbf{E})$. Finally, the off-diagonal elements of the second order derivative matrix are zero since we compute the saliency for one voxel at a time, leading to

$$\delta \mathrm{E}_i \simeq \sum_{e=1}^{E} \frac{\partial \mathrm{E}}{\partial e_{e,i}} \delta e_{e,i} + \frac{1}{2} \sum_{e=1}^{E} \frac{\partial^2 \mathrm{E}}{\partial e_{e,i}^2} \delta e_{e,i}^2 \tag{6.39}$$

as the voxel saliency estimate.

For the two-layer hyperbolic tangent perceptron and the MSE cost function we find the derivatives

$$\frac{\partial \mathrm{E}}{\partial e_{e,i}} = \frac{1}{N} \sum_{n=1}^{N} [y(\mathbf{E}^{\top} \mathbf{x}_n, \mathbf{w}) - g_n] \sum_{j=0}^{J} \tilde{w}_j (1 - z_j(\mathbf{E}^{\top} \mathbf{x}_n, \mathbf{w})^2) w_{ej} x_{n,i} \tag{6.40}$$

and

$$\frac{\partial^2 \mathrm{E}}{\partial e_{e,i}^2} = \frac{1}{N} \sum_{n=1}^{N} \left[\sum_{j=0}^{J} \tilde{w}_j (1 - z_j(\mathbf{E}^{\top} \mathbf{x}_n, \mathbf{w})^2) w_{ej} \right]^2 x_{n,i}^2 \tag{6.41}$$

by employing the LM approximation. Finally, the basis vector change for the i'th voxel is $\delta e_{e,i} = -e_{e,i}$ trivially; it corresponds to that voxel being removed.

6.5.1.2 The saliency map and correlated voxels

Appealing as the saliency map may appear there are situations were it reveals nothing. Consider a simple example where the signal is zero expect for a few voxels which correlate completely with the macroscopic variable. In this case the training error will remain the same no matter what voxel is removed; the saliency map provides no information whatsoever. Further, as argued in (Mørch et al., 1995), the saliency map is not necessarily confined to the model space spanned by the parameter vectors of the hidden units. These difficulties suggest that the saliency may be unsuited for MLP visualization, something which is a topic of current investigation.

6.5.2 Application to the CPH/SAC dataset

In lack of better visualization tools the saliency map was computed for a single of the optimal PCA-based MLP models with 5 parameters. The result is shown in figure 6.16. The panels depict a transverse, a coronal and a sagittal slice, respectively. When compared to the visualization of the microscopic linear regression model in figure 6.16 a number of similarities are evident, in particular the large emphasized area in the back of the brain which we once again identify as the visual cortex. However, detailed interpretation of the saliency map should be avoided due to it's debatable value as a visualization tool.

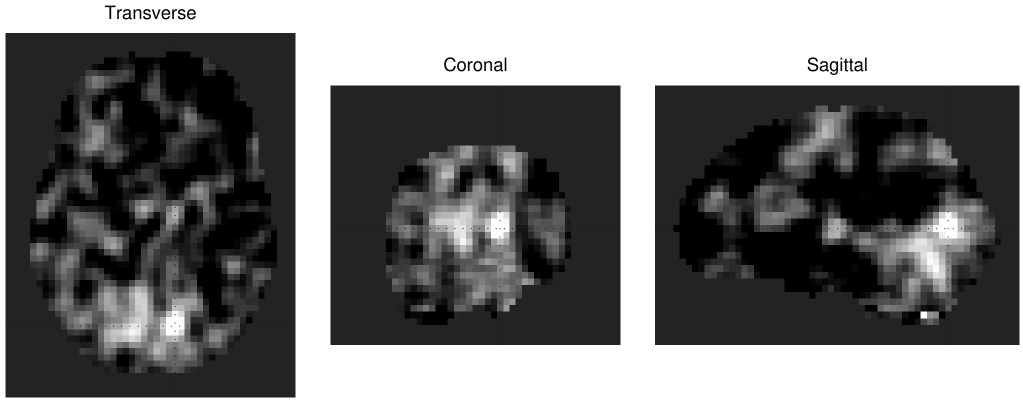

Figure 6.16: A transverse, a coronal, and a sagittal slice of the saliency map for a single nonlinear MLP model with 5 parameters based on the PCA representation of the CPH/SAC dataset. The large emphasized area in the back of the brain is the visual cortex.

6.6 Summary

Using two-layer perceptrons, which form a linear combination of nonlinear adaptive basis functions, we have provided empirical evidence of the applicability of flexible, nonlinear models in functional neuro modeling.

Model performance has been shown to depend on model flexibility and training set size. For the investigated class of two-layer perceptrons optimal model flexibility has been estimated using regularization and parameter pruning techniques. Further, empirical learning curves of model performance measures provide increasing support for the application of flexible nonlinear models with increasing training set size. By comparing the learning curves of linear and nonlinear models a generalization cross-over has been demonstrated, for which model performance of linear and nonlinear models coincide; for larger training set sizes the nonlinear model yields slightly better performance than it's linear counterpart.

Finally, the saliency map has been proposed in an attempt to visualize the emphasis of the nonlinear two-layer perceptron. Nonlinear model visualization deserves further investigation, however, since the proposed approach fails in certain situations.

Chapter 7

Conclusion

We have dealt with the analysis and modeling of data from functional neuro imaging experiments. In particular, we have proposed a generalization theoretical framework which relates model performance to model complexity and dataset size.

7.1 Summary of the proposed framework

Functional neuro imaging facilitates indirect quantitative spatially-distributed measures of brain function at a microscopic level. A functional experiment consists of such microscopic measurements of neuronal activity along with variables governing the macroscopic conditions under which the experiment is performed. The brain governs human behavior, so by assuming variations in the micro- and macroscopic variables to be manifestations of an underlying system we aim to gain insight into the function of the brain via system modeling. Consequently, we have proposed to investigate the joint probability density of sets of microscopic image volumes and corresponding macroscopic variables.

In typical functional datasets the dimensionality of each microscopic observation exceeds the number of observations by orders of magnitude. The ill-posed nature of the datasets holds major implications for analysis and modeling; primarily, efficiency can be increased by representing the microscopic observations using a basis that spans the same space as the set of microscopic observations themselves. Two basis selection procedures have been reviewed; principal component analysis and independent component analysis, providing uncorrelated and independent basis vectors, respectively. The assessment of their applicability has been approached using analysis of variance, which reveals that experimentally induced variance of interest constitutes only a tiny fraction of the total microscopic variance. Also, the variance related to intra-subject effects appears to be concentrated along a single of the principal basis vectors. For the independent basis no single such intra-subject related basis vector can be identified. The reason is not immediately clear; it is possible that the activity of involved neuro-physiological systems combine in a nonlinear fashion to produce the observed microscopic patterns.

To quantitatively assess model performance a statistical framework has been proposed. The approach is centered around measures of model generalization ability, i.e. the performance of models with parameters estimated in the limit of infinitely many observations. While generalization theory is well-studied in many areas, it's application is novel in the context of functional neuro modeling. Specifically, the observation that performance depends on both the number of observations and model complexity is important; it facilitates the determination of the extent to which a given dataset supports the application of complex models over simpler ones.

We have exemplified the proposed generalization theoretical framework by providing estimates of the parameters of linear as well as flexible nonlinear microscopic regression models. These approximate the conditional macroscopic density. Specifically, the dependency of model performance, as quantified by generalization error, on model flexibility and training set size has been demonstrated. For the investigated model classes the proposed framework was shown to provide estimates of optimal model complexity; implicitly as the optimal regularization parameter value, and explicitly as the limited model space identified by parameter pruning. Moreover, the estimated learning curves of model performance as functions of training set size signify the necessity of matching model flexibility to the number of available observations; large datasets support the application of more complex models.

Finally, linear models have been shown to emphasize a one-dimensional linear subspace of input space, which facilitates straightforward visualization of model emphasis. The saliency map has been proposed in an attempt to visualize nonlinear model emphasis, but the topic deserves further investigation due to deficiencies inherent to the saliency map.

7.2 Implications for functional neuro modeling

We have discussed and observed the suitability of generalization error as a measure of model performance. Specifically, training error has been proved a biased estimate of generalization error. This observation, which we have confirmed empirically, has a relatively important implication for functional neuro modeling; model flexibility should be chosen to match both the complexity and the size of the training set at hand. It follows that no model is uniformly better than all others. As a consequence optimal model performance can not be expected from *black-box* models; rather, model flexibility should be matched to each specific modeling task. The potential advantage is a model that more precisely approximates the true nature of the micro- and macroscopic relationship, paving the way for increased insight into the function of the human brain.

7.3 Suggestions for further work

While the realization that model flexibility relates to the number of available observations is important, much work along the lines of the proposed generalization theoretical framework remains. As examples of obvious areas that deserve further attention we mention the extension of the approach to cost functions other than mean square error. General bias-variance decompositions will provide for additional insights into the behavior of e.g. classification models applied to categorical experimental designs. Also, methods for model inversion as derived for the linear models should be examined in the general case; they potentially facilitate nonlinear model visualization and interpretation, which is something otherwise very difficult. Finally, the results herein deserve to be verified on other and larger datasets.

Appendix A

Dataset description

A.1 Copenhagen saccade PET dataset

The Copenhagen saccade PET dataset (CPH/SAC) is the result of a functional study aimed at characterizing the relationship between rate of eye movements and the regional neuronal activity in normal subjects during anti-saccadic eye movements (Law et al., 1995; Law, 1997).

A.1.1 Experimental design

Eight (six male, two female) strongly right-handed (Oldfield, 1971) normal subjects of age 21–33 (median age 24) were examined over two days in six activation and two fixation states. Stimulus was delivered by an array of light emitting diodes (LED's) located at visual angles $-40°$, $0°$ and $40°$ on a black perimeter arch with a radius of 35 cm. Performance was monitored using electrooculography (EOG), measuring eye movement frequency, saccade amplitude, direction, latency and error rate. During activation the subjects performed suppression of a reflexive saccade with performance of a volitional anti-saccade to the mirrored location of a randomly presented visual target. The condition was performed during six different target presentation frequencies: 0.05, 0.1, 0.3, 0.5, 0.7, and 0.9 Hz. Each target was followed by two saccades; the saccade towards the target and the return saccade towards the central LED, so the movement frequency was twice that of the presentation frequency. During each session subjects were fixating on the central LED which disappeared for a 200 ms gab before the appearance of a lateral target (exposure duration 200 ms). The gab was used to facilitate the execution of saccades (Fischer and Breitmeyer, 1987).

A.1.2 Acquisition and variables

Volumes of estimated neuronal activity were acquired with an Advance General Electric PET scanner. Standard preprocessing as described in section 1.1.2 was applied. Scans were intra-subject realignment to the first baseline scan using AIR (Woods et al., 1992), normalized by the injected dose relative to subject weight and spatially smoothed using a 3D boxcar filter. Subsequent inter-subject stereotactic normalization to a simulated PET reference volume in Talairach space (Talairach and Tournoux, 1988) using the 12 parameter linear transformation described in (Woods et al., 1993) yielded volumes with 48 slices, inter-slice distance of 3.4 mm, and in-plane resolution of 3.1 mm $\times$ 3.1 mm.

An intra-cerebral voxel mask was then created for each volume using thresholding and the anatomical knowledge of a trained operator. The intersection of the individual masks produced a common mask with 35701 remaining voxels; this was applied to all scans, leading to the set of microscopic variables $\mathsf{X} = \{\mathbf{x}_n \mid n = 1, \ldots, 64\}$ and the corresponding microscopic data matrix $\mathbf{X}$.

A number of macroscopic variables were recorded during acquisition. They included eye movement frequency, saccade amplitude, direction, latency, and error rate, as well as subject sex, age, and weight, giving rise to the set of macroscopic variables $\mathsf{G} = \{\mathbf{g}_n \mid n = 1, \ldots, 64\}$ and the corresponding macroscopic data matrix $\mathbf{G}$. Of the 64 scans, none were classified as errors (defined as absent saccades, saccades outside a target interval of $\pm 5^\circ$ or misdirected/corrected saccades). For analysis focusing on the frequency of the saccades the frequency of the flashing LED targets was used. The frequency of the actual saccades differed only insignificantly from this estimate.

Appendix B

Information theory

Information theory was originally formulated by Shannon at Bell Labs while working on the problem of efficiently transmitting information over a noisy communication channel (Shannon, 1948). His work has since formed the basis for methods and algorithms in many fields.

B.1 Entropy

Given a random variable x with an associated probability density function (p.d.f.) $f(x)$ the entropy of x is denoted $\mathrm{H}(x)$ and defined as

$$\mathrm{H}[f(x)] \stackrel{\text{def}}{=} -\int f(x) \log f(x)\, dx = -\langle \log f(x) \rangle_{f(x)} = \left\langle \log \frac{1}{f(x)} \right\rangle_{f(x)} \tag{B.1}$$

where we use $\langle \cdot \rangle_{f(x)}$ to denote expectation with respect to probability density $f(x)$. In cases where the p.d.f. of x is clear convenient abuses of notation are

$$\mathrm{H}[f(x)] = \mathrm{H}[f] = \mathrm{H}[x] \quad . \tag{B.2}$$

Entropy measures the *uncertainty* of a random variable. More precisely it measures the uncertainty about the events quantified by the random variable x given its p.d.f. $f(x)$ *prior* to the realization of x. The realized value of x removes the uncertainty, and thus provides *information* equal to the entropy of x. Maximum uncertainty occurs for a random variable with uniform p.d.f., whereas realizations of a random variable with a delta function p.d.f. always are the same. The entropy measure quantifies this difference in uncertainty.

Example B.1 *Let x denote the outcome of a fair die roll. We have*

$$\mathrm{P}(x=1) = \mathrm{P}(x=2) = \ldots = \mathrm{P}(x=6) = \frac{1}{6} \quad .$$

Hence

$$\mathrm{H}[x] = -\frac{1}{6}\log\frac{1}{6} - \ldots - \frac{1}{6}\log\frac{1}{6} = \log 6 = 1.79 \quad ,$$

where we use $\log(\cdot) = \log_e(\cdot)$. If, on the other hand, y denotes the outcome of an unfair *die with a non-uniform p.d.f.*

$$\mathrm{P}(y=1) = \ldots = \mathrm{P}(y=5) = \frac{1}{10}, \quad \mathrm{P}(y=6) = \frac{1}{2}$$

we have

$$\mathrm{H}[y] = -\frac{5}{10}\log\frac{1}{10} - \frac{1}{2}\log\frac{1}{2} = \frac{1}{2}(\log 10 + \log 2) = 1.50 \quad .$$

The entropy is smaller in the non-uniform case due to the less uncertain outcome of the unfair die roll.

The definition (B.1) can be derived from a number of postulates based on our intuitive understanding of uncertainty, see e.g. (Papoulis, 1991) which provides a more rigorous treatment of entropy and related subjects. However, (B.1) may also be viewed simply as a definition with a number of useful properties, of which a few will be introduced below.

B.2 Joint and conditional entropy

Entropy easily generalizes to multivariate random variables, also resulting in the straightforward definition of joint entropy for random variables x and y with joint p.d.f. $f(x,y)$

$$\mathrm{H}[f(x,y)] \stackrel{\text{def}}{=} -\iint f(x,y)\log f(x,y)\,dx\,dy = \left\langle \log\frac{1}{f(x,y)}\right\rangle_{f(x,y)} \quad . \tag{B.3}$$

Similarly, with the conditional p.d.f. $f(x|y)$ the conditional entropy of x on y is defined as

$$\mathrm{H}[f(x|y)] \stackrel{\text{def}}{=} -\iint f(x|y)\log f(x|y)\,dx\,dy = \left\langle \log\frac{1}{f(x|y)}\right\rangle_{f(x|y)} \quad . \tag{B.4}$$

These two definitions will help us understand the important concept of mutual information as explained in section B.4.

B.3 Kullback-Leibler entropy

Given N realizations, $x_1, x_2, \ldots, x_N$, of a random variable x we are often faced with the problem of estimating the associated p.d.f. $f(x)$. In a maximum likelihood (ML) setting this problem leads to the definition of the Kullback-Leibler entropy[1] as follows[2]:

Let $f_\theta(x) \mid \theta \in \Theta$ denote a parametric model of the density $f(x)$. Assuming independent realizations the likelihood that the sample is drawn from distribution f_θ is

$$\mathrm{l}_N(\theta) = \prod_{n=1}^{N} f_\theta(x_n) \tag{B.5}$$

resulting in the corresponding normalized negative log-likelihood

$$-\mathrm{L}_N(\theta) = -\frac{1}{N}\log \mathrm{l}_N(\theta) = -\frac{1}{N}\sum_{n=1}^{N}\log f_\theta(x_n) \quad , \tag{B.6}$$

which in turn is the negative sample average of $\log f_\theta(x)$. As the number of realizations goes to infinity (B.6) converges in probability to the expectation

$$\mathrm{L}(\theta) = -\lim_{N\to\infty}\mathrm{L}_N(\theta) = -\int f(x)\log f_\theta(x)\,dx = \langle -\log f_\theta(x)\rangle_{f(x)} \quad . \tag{B.7}$$

[1] Also called the Kullback-Leibler distance or the Kullback-Leibler divergence.

[2] Derivation inspired by (Cardoso, 1997).

Expression (B.7) is the so-called *cross-entropy* between the densities $f(x)$ and $f_\theta(x)$ and can be regarded as a measure of the extent to which the model density and the true density agree. By observing that the cross-entropy for $f_\theta(x) = f(x)$ evaluates to the entropy of $f(x)$

$$\mathrm{L}(\theta)\big|_{f_\theta(x)=f(x)} = \langle -\log f(x)\rangle_{f(x)} = \mathrm{H}\left[f(x)\right] \quad , \tag{B.8}$$

we can subtract this value from (B.7) to achieve the non-negative Kullback-Leibler entropy

$$\begin{aligned}
\mathrm{K}\left[f(x); f_\theta(x)\right] &= \mathrm{L}(\theta) - \mathrm{H}\left[f(x)\right] && \text{(B.9)}\\
&= \langle -\log f_\theta(x)\rangle_{f(x)} - \langle -\log f(x)\rangle_{f(x)} && \text{(B.10)}\\
&= -\left\langle \log \frac{f_\theta(x)}{f(x)} \right\rangle_{f(x)} && \text{(B.11)}\\
&= \left\langle \log \frac{f(x)}{f_\theta(x)} \right\rangle_{f(x)} \quad . && \text{(B.12)}
\end{aligned}$$

It is important to note that the integration is with respect to $f(x)$, meaning that $\mathrm{K}\left[\cdot;\cdot\right]$ is non-symmetric

$$\mathrm{K}\left[f(x); f_\theta(x)\right] \neq \mathrm{K}\left[f_\theta(x); f(x)\right] \quad . \tag{B.13}$$

The Kullback-Leibler entropy has some important properties. Firstly, it is invariant under an invertible transformation $t(\cdot)$ of the sample space

$$\mathrm{K}\left[f(x); g(x)\right] = \mathrm{K}\left[t(f(x)); t(g(x))\right] = \mathrm{K}\left[t^{-1}(f(x)); t^{-1}(g(x))\right] \quad . \tag{B.14}$$

Secondly, the Kullback-Leibler entropy between a distribution $f(x)$ and the uniform distribution $1(x)$ over the interval $[0;1]$ equals the negative entropy of $f(x)$, as can be seen by

$$\mathrm{K}\left[f(x); 1(x)_{[0;1]}\right] = \left\langle \log \frac{f(x)}{1(x)_{[0;1]}} \right\rangle_{f(x)} = -\mathrm{H}\left[f(x)\right] \quad . \tag{B.15}$$

B.4 Mutual information

For two random variables we label

$$\mathrm{I}\left[x; y\right] \stackrel{\text{def}}{=} \mathrm{H}\left[x\right] + \mathrm{H}\left[y\right] - \mathrm{H}\left[x, y\right] \tag{B.16}$$

the *mutual information* of x and y. Using the definitions of entropy (B.1) and (B.3) to rewrite it as an expected value

$$\begin{aligned}
\mathrm{I}\left[x; y\right] &\stackrel{\text{def}}{=} \mathrm{H}\left[x\right] + \mathrm{H}\left[y\right] - \mathrm{H}\left[x, y\right] && \text{(B.17)}\\
&= \left\langle \log \frac{1}{f(x)} \right\rangle_{f(x)} + \left\langle \log \frac{1}{f(y)} \right\rangle_{f(y)} + \langle \log f(x,y)\rangle_{f(x,y)} && \text{(B.18)}\\
&= \left\langle \log \frac{f(x,y)}{f(x) f(y)} \right\rangle_{f(x,y)} && \text{(B.19)}
\end{aligned}$$

we see that mutual information quantifies the extent to which the joint distribution of two random variables resembles the product of their marginal distributions. Since two random variables x and y are statistically independent when the product of their marginal distributions equals the joint distribution

$$f(x)f(y) = f(x,y) \quad \Leftrightarrow \quad x \text{ and } y \text{ are statistically independent,} \tag{B.20}$$

mutual information is a measure of statistical independence. In fact, by comparing (B.19) and (B.12), we see that mutual information is the Kullback-Leibler distance between $f(x,y)$ and $f(x)f(y)$

$$\mathrm{I}\left[x;y\right] = \mathrm{K}\left[f(x,y);f(x)f(y)\right] \quad . \tag{B.21}$$

Inserting $f(x,y) = f(x|y)f(y)$ into (B.19)

$$\mathrm{I}\left[x;y\right] = \left\langle \log \frac{f(x,y)}{f(x)f(y)} \right\rangle_{f(x,y)} = \left\langle \log \frac{f(x|y)}{f(x)} \right\rangle_{f(x,y)} \tag{B.22}$$

and using the definition of conditional entropy (B.4) we can re-express mutual information as

$$\mathrm{I}\left[x;y\right] = \mathrm{H}\left[x\right] - \mathrm{H}\left[x|y\right] \quad , \tag{B.23}$$

and by substituting y for x and vice versa also as

$$\mathrm{I}\left[x;y\right] = \mathrm{H}\left[y\right] - \mathrm{H}\left[y|x\right] \quad . \tag{B.24}$$

Appendix C

Expected generalization error estimation

The quality of model parameter estimates depends on the number of observations in the training set; more observations provide better estimates. To avoid holding out observations in a test set to obtain an empirical generalization error estimate thus reducing the size of the training set, we derive an algebraic generalization error estimate.

C.1 Assumptions and definitions

We begin with a number of assumptions

- The set of true parameters $\mathbf{w}^*$ falls within the set of relationships that the parameterized model can implement (see the discussion in section 2.2).
- Noise is additive and independent between observations, and has zero mean.
- The number of observations is large.

Let $e(\mathbf{x}_n, \mathbf{w})$ denote the log-likelihood of a single observation so that the unregularized training error can be written as

$$\mathrm{E}(\mathsf{D}, \mathbf{w}) = \frac{1}{N} \sum_{n=1}^{N} e(\mathbf{x}_n, \mathbf{w}) \quad . \tag{C.1}$$

In the following we leave out the explicit mention of the training set D for ease of notation, so we write $\mathrm{E}(\mathbf{w})$ instead of $\mathrm{E}(\mathsf{D}, \mathbf{w})$. Let further $\mathbf{w}_{\mathrm{E}}$ denote a minimum of $\mathrm{E}(\mathbf{w})$, and $\mathbf{w}_{\mathrm{C}}$ a minimum of the regularized cost function $\mathrm{C}(\mathbf{w})$.

The aim is to find a relationship between the expected generalization error

$$\bar{\mathrm{G}} = \langle \mathrm{G} \rangle_{\mathsf{D}} = \int \mathrm{G}(\mathsf{D}, \mathbf{w}) \mathrm{p}(\mathsf{D}) \; \mathrm{d}\mathsf{D} \tag{C.2}$$

and its training error equivalent

$$\bar{\mathrm{E}} = \langle \mathrm{E} \rangle_{\mathsf{D}} = \int \mathrm{E}(\mathsf{D}, \mathbf{w}) \mathrm{p}(\mathsf{D}) \; \mathrm{d}\mathsf{D} \tag{C.3}$$

where we have simplified the notation further by writing D and $\langle \mathrm{G} \rangle_{\mathsf{D}}$ instead of $\mathsf{D}(N)$ and $\langle \mathrm{G}(\mathsf{D}, \mathbf{w}) \rangle_{\mathrm{p}(\mathsf{D}(N))}$, respectively. As the number of training examples increases, $N \to \infty$, the parameters $\mathbf{w}$ approach the true parameters $\mathbf{w}^*$, for which we label the expected[1] generalization and training error $\mathrm{G}(\mathbf{w}^*)$ and $\mathrm{E}(\mathbf{w}^*)$.

C.2 Parameter fluctuations

With the assumptions and definitions in place we investigate a Taylor expansion of the regularized cost function to second order around the true parameters $\mathbf{w}^*$

$$\begin{aligned} \mathrm{C}(\mathbf{w}) &= \mathrm{C}(\mathbf{w}^*) + \frac{\partial \mathrm{C}(\mathbf{w}^*)}{\partial \mathbf{w}^\mathsf{T}} \delta \mathbf{w} + \frac{1}{2} \delta \mathbf{w}^\mathsf{T} \frac{\partial^2 \mathrm{C}(\mathbf{w}^*)}{\partial \mathbf{w} \partial \mathbf{w}^\mathsf{T}} \delta \mathbf{w} + \mathcal{O}\left(|\delta \mathbf{w}|^3 \right) && \text{(C.4)} \\ &= \mathrm{C}(\mathbf{w}^*) + \nabla^\mathsf{T} \mathrm{C}(\mathbf{w}^*) \delta \mathbf{w} + \frac{1}{2} \delta \mathbf{w}^\mathsf{T} \mathbf{J} \delta \mathbf{w} + \mathcal{O}\left(|\delta \mathbf{w}|^3 \right) \quad , && \text{(C.5)} \end{aligned}$$

where we have introduced the cost function gradient

$$\nabla \mathrm{C}(\mathbf{w}^*) = \frac{\partial \mathrm{C}(\mathbf{w}^*)}{\partial \mathbf{w}} = \nabla \mathrm{E}(\mathbf{w}^*) + \frac{1}{N} \nabla \mathrm{R}(\mathbf{w}^*) \tag{C.6}$$

and the *Hessian* matrix of second order derivatives

$$\mathbf{J} = \frac{\partial^2 \mathrm{C}(\mathbf{w}^*)}{\partial \mathbf{w} \partial \mathbf{w}^\mathsf{T}} = \frac{\partial^2 \mathrm{E}(\mathbf{w}^*)}{\partial \mathbf{w} \partial \mathbf{w}^\mathsf{T}} + \frac{1}{N} \frac{\partial^2 \mathrm{R}(\mathbf{w}^*)}{\partial \mathbf{w} \partial \mathbf{w}^\mathsf{T}} = \mathbf{H} + \frac{1}{N} \frac{\partial^2 \mathrm{R}(\mathbf{w}^*)}{\partial \mathbf{w} \partial \mathbf{w}^\mathsf{T}} \quad . \tag{C.7}$$

In the following we ignore the higher order terms $\mathcal{O}\left(|\delta \mathbf{w}|^3 \right)$.

For $\mathbf{w} = \mathbf{w}_\mathrm{C}$ the parameter fluctuation is $\delta \mathbf{w}_\mathrm{C} = \mathbf{w}_\mathrm{C} - \mathbf{w}^*$. Since $\mathbf{w}_\mathrm{C}$ by definition is a minimum of the regularized cost function the derivative of (C.5) evaluated in $\mathbf{w}_\mathrm{C}$ is zero

$$\begin{aligned} \nabla \mathrm{C}(\mathbf{w}_\mathrm{C}) = 0 = \nabla \mathrm{C}(\mathbf{w}^*) + \mathbf{J} \delta \mathbf{w}_\mathrm{C} && \text{(C.8)} \\ \Updownarrow && \text{(C.9)} \\ \delta \mathbf{w}_\mathrm{C} = \mathbf{w}_\mathrm{C} - \mathbf{w}^* = -\mathbf{J}^{-1} \nabla \mathrm{C}(\mathbf{w}^*) \quad . && \text{(C.10)} \end{aligned}$$

We can now compute the first order moment of the parameter fluctuations around $\mathbf{w}^*$

$$\begin{aligned} \langle \delta \mathbf{w}_\mathrm{C} \rangle_\mathsf{D} &= \left\langle -\mathbf{J}^{-1} \left(\nabla \mathrm{E}(\mathbf{w}^*) + \frac{1}{N} \nabla \mathrm{R}(\mathbf{w}^*) \right) \right\rangle_\mathsf{D} && \text{(C.11)} \\ &\simeq -\mathbf{J}^{-1} \left(\langle \nabla \mathrm{E}(\mathbf{w}^*) \rangle_\mathsf{D} + \frac{1}{N} \langle \nabla \mathrm{R}(\mathbf{w}^*) \rangle_\mathsf{D} \right) && \text{(C.12)} \\ &= -\frac{1}{N} \mathbf{J}^{-1} \langle \nabla \mathrm{R}(\mathbf{w}^*) \rangle_\mathsf{D} \quad , && \text{(C.13)} \end{aligned}$$

where we use $\langle \nabla \mathrm{E}(\mathbf{w}^*) \rangle_\mathsf{D} = \nabla \mathrm{G}(\mathbf{w}^*) = 0$. Similarly, the second order moment of the parameter fluctuations is

$$\begin{aligned} \langle \delta \mathbf{w}_\mathrm{C} \delta \mathbf{w}_\mathrm{C}^\mathsf{T} \rangle_\mathsf{D} &= \left\langle -\mathbf{J}^{-1} \nabla \mathrm{C}(\mathbf{w}^*) \nabla^\mathsf{T} \mathrm{C}(\mathbf{w}^*) \left(\mathbf{J}^{-1} \right)^\mathsf{T} \right\rangle_\mathsf{D} && \text{(C.14)} \\ &\simeq \mathbf{J}^{-1} \langle \nabla \mathrm{E}(\mathbf{w}^*) \nabla^\mathsf{T} \mathrm{E}(\mathbf{w}^*) \rangle_\mathsf{D} \left(\mathbf{J}^{-1} \right)^\mathsf{T} \\ &\quad + \frac{1}{N^2} \mathbf{J}^{-1} \langle \nabla \mathrm{R}(\mathbf{w}^*) \nabla^\mathsf{T} \mathrm{R}(\mathbf{w}^*) \rangle_\mathsf{D} \left(\mathbf{J}^{-1} \right)^\mathsf{T} && \text{(C.15)} \\ &= \mathbf{J}^{-1} \langle \nabla \mathrm{E}(\mathbf{w}^*) \nabla^\mathsf{T} \mathrm{E}(\mathbf{w}^*) \rangle_\mathsf{D} \left(\mathbf{J}^{-1} \right)^\mathsf{T} \\ &\quad + \frac{1}{N^2} \mathbf{J}^{-1} \nabla \mathrm{R}(\mathbf{w}^*) \nabla^\mathsf{T} \mathrm{R}(\mathbf{w}^*) \left(\mathbf{J}^{-1} \right)^\mathsf{T} \quad . && \text{(C.16)} \end{aligned}$$

[1] In the limit $N \to \infty$ the expectation over D vanishes so we write $\mathrm{G}(\mathbf{w}^*)$ instead of $\langle \mathrm{G}(\mathbf{w}^*) \rangle_{\mathsf{D}_\infty}$, where D_∞ signifies the infinitely large training set.

In (C.15) we have employed $\langle \nabla \mathrm{E}(\mathbf{w}^*) \nabla^\mathsf{T} \mathrm{R}(\mathbf{w}^*) \rangle_\mathsf{D} = 0$ which holds since $\nabla \mathrm{G}(\mathbf{w}^*) = 0$ as before. Further, for the first expectation in (C.16) we find

$$\begin{aligned}
\langle \nabla \mathrm{E}(\mathbf{w}^*) \nabla^\mathsf{T} \mathrm{E}(\mathbf{w}^*) \rangle_\mathsf{D} &= \frac{1}{N^2} \sum_{nn'} \langle \nabla e(\mathbf{x}_n, \mathbf{w}^*) \nabla^\mathsf{T} e(\mathbf{x}_{n'}, \mathbf{w}^*) \rangle_\mathsf{D} && \text{(C.17)} \\
&\simeq \frac{1}{N} \langle \nabla e(\mathbf{x}, \mathbf{w}^*) \nabla^\mathsf{T} e(\mathbf{x}, \mathbf{w}^*) \rangle_\mathsf{D} && \text{(C.18)} \\
&= \frac{1}{N} \mathbf{Q} \quad , && \text{(C.19)}
\end{aligned}$$

where we identify $\mathbf{Q}$ as *Fisher's information matrix* (Mardia et al., 1979, page 98). In (C.18) we have used to fact that the error is independent between observations. Substituting (C.19) back into (C.16) yields

$$\langle \delta \mathbf{w}_\mathrm{C} \delta \mathbf{w}_\mathrm{C}^\mathsf{T} \rangle_\mathsf{D} = \frac{1}{N} \mathbf{J}^{-1} \mathbf{Q} \left(\mathbf{J}^{-1} \right)^\mathsf{T} + \frac{1}{N^2} \mathbf{J}^{-1} \nabla \mathrm{R}(\mathbf{w}^*) \nabla^\mathsf{T} \mathrm{R}(\mathbf{w}^*) \left(\mathbf{J}^{-1} \right)^\mathsf{T} \quad . \qquad \text{(C.20)}$$

C.3 Estimating the expected generalization error

Using the derived moments (C.13) and (C.20) we can compute the expected generalization error in (C.2). The Taylor expansion to second order around $\mathbf{w}^*$ becomes

$$\begin{aligned}
\langle \mathrm{G} \rangle_\mathsf{D} &\simeq \langle \mathrm{G}(\mathbf{w}^*) \rangle_\mathsf{D} + \nabla^\mathsf{T} \mathrm{G}(\mathbf{w}^*) \langle \delta \mathbf{w}_\mathrm{C} \rangle_\mathsf{D} + \frac{1}{2} \left\langle \delta \mathbf{w}_\mathrm{C}^\mathsf{T} \frac{\partial^2 \mathrm{G}(\mathbf{w}^*)}{\partial \mathbf{w} \partial \mathbf{w}^\mathsf{T}} \delta \mathbf{w}_\mathrm{C} \right\rangle_\mathsf{D} && \text{(C.21)} \\
&\simeq \mathrm{G}(\mathbf{w}^*) + \frac{1}{2} \mathrm{tr} \left[\frac{\partial^2 \mathrm{G}(\mathbf{w}^*)}{\partial \mathbf{w} \partial \mathbf{w}^\mathsf{T}} \langle \delta \mathbf{w}_\mathrm{C} \delta \mathbf{w}_\mathrm{C}^\mathsf{T} \rangle_\mathsf{D} \right] && \text{(C.22)} \\
&= \mathrm{G}(\mathbf{w}^*) + \frac{1}{2N} \mathrm{tr} \left[\frac{\partial^2 \mathrm{G}(\mathbf{w}_\mathrm{C})}{\partial \mathbf{w} \partial \mathbf{w}^\mathsf{T}} \mathbf{J}^{-1} \mathbf{Q} \left(\mathbf{J}^{-1} \right) \right] \\
&\quad + \frac{1}{2N^2} \mathrm{tr} \left[\frac{\partial^2 \mathrm{G}(\mathbf{w}_\mathrm{C}))}{\partial \mathbf{w} \partial \mathbf{w}^\mathsf{T}} \mathbf{J}^{-1} \nabla \mathrm{R}(\mathbf{w}_\mathrm{C})) \nabla^\mathsf{T} \mathrm{R}(\mathbf{w}_\mathrm{C})) \left(\mathbf{J}^{-1} \right)^\mathsf{T} \right] \quad , && \text{(C.23)}
\end{aligned}$$

where we in (C.23) assume that $\mathbf{w}_\mathrm{C}$ is close to $\mathbf{w}^*$ so the second order derivative of the generalization error is close to the second order derivative of the training error and likewise for the regularization gradient. The gradient term in (C.21) vanishes since $\nabla \mathrm{G}(\mathbf{w}^*) = 0$.

C.4 Estimating the expected training error

In a manner similar to (C.23) we approximate the expected training error in (C.3) by a Taylor expansion to second order around $\mathbf{w}^*$

$$\begin{aligned}
\langle \mathrm{E} \rangle_\mathsf{D} &\simeq \left\langle \frac{1}{N} \sum_{n=1}^{N} e(\mathbf{x}_n, \mathbf{w}^*) \right\rangle_\mathsf{D} && \text{(C.24)} \\
&\quad + \left\langle \frac{1}{N} \sum_{n=1}^{N} \nabla^\mathsf{T} e(\mathbf{x}_n, \mathbf{w}^*) \delta \mathbf{w}_\mathrm{C} \right\rangle_\mathsf{D} && \text{(C.25)} \\
&\quad + \frac{1}{2} \left\langle \frac{1}{N} \sum_{n=1}^{N} \delta \mathbf{w}_\mathrm{C}^\mathsf{T} \frac{\partial^2 e(\mathbf{x}_n, \mathbf{w}^*)}{\partial \mathbf{w} \partial \mathbf{w}^\mathsf{T}} \delta \mathbf{w}_\mathrm{C} \right\rangle_\mathsf{D} \quad . && \text{(C.26)}
\end{aligned}$$

For the first term (C.24) we find

$$\left\langle \frac{1}{N}\sum_{n=1}^{N} e(\mathbf{x}_n, \mathbf{w}^*) \right\rangle_{\mathsf{D}} = \mathrm{G}(\mathbf{w}^*) \quad , \tag{C.27}$$

while the second term (C.25) becomes

$$\begin{aligned}
&\left\langle \frac{1}{N}\sum_{n=1}^{N} \nabla^{\mathsf{T}} e(\mathbf{x}_n, \mathbf{w}^*)\delta\mathbf{w}_{\mathrm{C}} \right\rangle_{\mathsf{D}} && \text{(C.28)}\\
&= -\left\langle \frac{1}{N}\sum_{n=1}^{N} \nabla^{\mathsf{T}} e(\mathbf{x}_n, \mathbf{w}^*)\mathbf{J}^{-1}\left(\frac{1}{N}\sum_{n'=1}^{N} \nabla e(\mathbf{x}_{n'}, \mathbf{w}^*) + \frac{1}{N}\nabla \mathrm{R}(\mathbf{w}^*)\right)\right\rangle_{\mathsf{D}} && \text{(C.29)}\\
&= -\mathrm{tr}\left[\mathbf{J}^{-1}\left\langle \frac{1}{N^2}\sum_{nn'} \nabla e(\mathbf{x}_n, \mathbf{w}^*)\nabla^{\mathsf{T}} e(\mathbf{x}_{n'}, \mathbf{w}^*)\right\rangle_{\mathsf{D}}\right] && \text{(C.30)}\\
&= -\frac{1}{N}\mathrm{tr}\left[\mathbf{J}^{-1}\mathbf{Q}\right] \quad , && \text{(C.31)}
\end{aligned}$$

by using (C.19). Finally, the third term (C.26) is

$$\begin{aligned}
&\frac{1}{2}\left\langle \frac{1}{N}\sum_{n=1}^{N} \delta\mathbf{w}_{\mathrm{C}}^{\mathsf{T}} \frac{\partial^2 e(\mathbf{x}_n, \mathbf{w}^*)}{\partial \mathbf{w}\partial \mathbf{w}^{\mathsf{T}}}\delta\mathbf{w}_{\mathrm{C}} \right\rangle_{\mathsf{D}} && \text{(C.32)}\\
&= \frac{1}{2N}\mathrm{tr}\left[\frac{\partial^2 \mathrm{E}(\mathbf{w}_{\mathrm{C}})}{\partial \mathbf{w}\partial \mathbf{w}^{\mathsf{T}}}\mathbf{J}^{-1}\mathbf{Q}\left(\mathbf{J}^{-1}\right)^{\mathsf{T}}\right] && \text{(C.33)}\\
&\quad + \frac{1}{2N^2}\mathrm{tr}\left[\frac{\partial^2 \mathrm{E}(\mathbf{w}_{\mathrm{C}})}{\partial \mathbf{w}\partial \mathbf{w}^{\mathsf{T}}}\mathbf{J}^{-1}\nabla \mathrm{R}(\mathbf{w}_{\mathrm{C}})\nabla^{\mathsf{T}}\mathrm{R}(\mathbf{w}_{\mathrm{C}})\left(\mathbf{J}^{-1}\right)^{\mathsf{T}}\right] \quad . && \text{(C.34)}
\end{aligned}$$

Substituting the terms back into the Taylor expansion of the expected training error we find

$$\begin{aligned}
\langle \mathrm{E}\rangle_{\mathsf{D}} \simeq\ & \mathrm{G}(\mathbf{w}^*) - \frac{1}{N}\mathrm{tr}\left[\mathbf{J}^{-1}\mathbf{Q}\right] + \frac{1}{2N}\mathrm{tr}\left[\frac{\partial^2 \mathrm{E}(\mathbf{w}_{\mathrm{C}})}{\partial \mathbf{w}\partial \mathbf{w}^{\mathsf{T}}}\mathbf{J}^{-1}\mathbf{Q}\left(\mathbf{J}^{-1}\right)^{\mathsf{T}}\right]\\
& + \frac{1}{2N^2}\mathrm{tr}\left[\frac{\partial^2 \mathrm{E}(\mathbf{w}_{\mathrm{C}})}{\partial \mathbf{w}\partial \mathbf{w}^{\mathsf{T}}}\mathbf{J}^{-1}\nabla \mathrm{R}(\mathbf{w}_{\mathrm{C}})\nabla^{\mathsf{T}}\mathrm{R}(\mathbf{w}_{\mathrm{C}})\left(\mathbf{J}^{-1}\right)^{\mathsf{T}}\right] \quad .
\end{aligned} \tag{C.35}$$

C.5 Combining the estimates

Comparing the approximated expected training error (C.35) with the approximated expected generalization error in (C.23) most terms cancel, leaving us with the very interesting relationship

$$\langle \mathrm{G}\rangle_{\mathsf{D}} = \langle \mathrm{E}\rangle_{\mathsf{D}} + \frac{1}{N}\mathrm{tr}\left[\mathbf{J}^{-1}\mathbf{Q}\right] \quad , \tag{C.36}$$

subject to assumptions and approximations as noted throughout this appendix. The implication of (C.36) is that the expected training error is a *biased* estimator of expected generalization error. The derived generalization error estimate quantifies the bias, enabling us to estimate expected generalization performance without setting aside observations in a test set. We must keep in mind, however, the assumption of N being large.

Appendix D

Contribution to ICNN'95

This appendix contains the paper "Visualization of Neural Networks Using Saliency Maps" (Mørch et al., 1995), orally presented at the 1995 IEEE International Conference on Neural Networks (ICNN'95) in Perth, Australia.

Visualization of Neural Networks Using Saliency Maps

Niels J. S. Mørch[+‡] Ulrik Kjems[+] Lars Kai Hansen[+]
Claus Svarer[‡] Ian Law[‡] Benny Lautrup[†] Steve Strother[♮] Kelly Rehm[♮]

[+] Electronics Institute
Technical University of Denmark
DK-2800 Lyngby, Denmark

[‡] Department of Neurology
National University Hospital, Rigshospitalet
DK-2100 Copenhagen Ø, Denmark

[†] Niels Bohr Institute
University of Copenhagen
DK-2100 Copenhagen Ø, Denmark

[♮] PET Imaging Service, Va Medical Center
Radiology and Health Informatics Depts.
University of Minnesota, Minneapolis
Minnesota, 55417, USA

E-Mail : nmorch@ei.dtu.dk

ABSTRACT

The saliency map is proposed as a new method for understanding and visualizing the nonlinearities embedded in feed-forward neural networks, with emphasis on the ill-posed case, where the dimensionality of the input-field by far exceeds the number of examples. Several levels of approximations are discussed. The saliency maps are applied to medical imaging (PET-scans) for identification of paradigm-relevant regions in the human brain.
Keywords: saliency map, model interpretation, ill-posed learning, PCA, SVD, PET.

1. Introduction

Mathematical modeling is of increasing importance in medical informatics. In bio-medical context the aim of neural network modeling is often twofold. Besides using empirical relations established within a given model, there is typically a wish to interpret the model in order to achieve an *understanding* of the processes underlying and generating the data. This paper presents a new tool for such opening of the neural network "black box".

Our method is aimed at neural network applications where the network is trained to provide a relation between huge, highly correlated, measurements and simple "labels". The measurement could e.g. be a spectrum, an image, or as in our particular case a brain scan volume. The label could be a concentration, a diagnosis etc.

In neural network applications, an important aspect of the training process is the architecture synthesis. An architecturally optimized network supplies structural information about the input field as used by the model, thus giving a qualitative measure of importance.

The output of our new procedure is a "map" *quantifying* the importance (*saliency* c.f. [7]) of each individual component of the measurement (i.e. pin, pixel, or voxel) with respect to the obtained empirical relation. Hopefully, this so-called *saliency map* will assist the modeler in interpreting the model and in communicating the interpretation to the end-user.

In bio-medical context it is often hard (not to say expensive) to gather large samples of data. Hence, if modeling from high dimensional data based on small samples, one faces an extremely ill-posed learning problem and standard practice has been to apply hand crafted tools ("a priori knowledge") for preprocessing and data reduction in order to bring down the dimensionality of the neural network. However, we have recently shown that one may cure this extremely ill-posed problem using straightforward linear algebra *without loss of information* [2], [5]. The scheme achieves *massive weight sharing* [7] by projecting the high dimensional data onto a low dimensional basis spanning the so-called signal space of the training set input vectors. The saliency map is an attempt to visualize this induced geometry and the specific manner in which this geometry is used by the trained network.

As a specific case, we consider modeling of images obtained from Positron-Emission-Tomography

(PET)-scans which is a technique offering 3-dimensional volume measurements of human brain activity. A neural network may be trained using supervised learning on a given training set of PET-scans [2], [5]. We investigate two cases, based on two sets of 64 scans each (8 subjects scanned 8 times): one where the subjects perform an eye movement task according to a graduated (parameterized) paradigm [6], and one where they perform a finger opposition task [12]. In the first case the network is trained to predict the paradigm graduation parameter—the frequency of the saccadic eye movements—using the measured activation patterns in the brain volume as input. In the latter the network is trained to classify the measured activation patterns as rest or activated (i.e. doing the finger opposition task). Since the models are nonlinear, the interpretations are not straightforward. In this particular case the saliency map can be viewed as a tool for visualizing the regions in the brain, which are related most strongly to the specific tasks.

2. The Saliency Map

It is well-known that affine preprocessing [8, 10] can assist training and generalization significantly. Affine preprocessing of an input vector $\boldsymbol{x}_j$ (i.e. an element of the training set of inputs $\mathbf{X} = [\boldsymbol{x}_1 \dots \boldsymbol{x}_J]$) can be expressed as $\boldsymbol{v}_j = \mathbf{B}^T(\boldsymbol{x}_j - \boldsymbol{c})$. In fact, translating by the training set averaged input vector $\boldsymbol{c} = \bar{\boldsymbol{x}}$ and computing the projection matrix $\mathbf{B}$ from a diagonalization of the input covariance matrix we may obtain $\boldsymbol{v}_j$ as the principal components[1] of $\mathbf{X}$. For simplicity we replace $\boldsymbol{x}_j - \boldsymbol{c}$ with $\boldsymbol{x}_j$ in the following, without loss of generality.

In image or volume processing, where the number of input channels I is often much greater than the number of examples J, a transformation like above can be used to reduce the dimensionality of the data-representation. However, it should be noted that within our scheme for handling extremely ill-posed problems the preprocessing doesn't necessarily reduce the data[2], in contrast to what is often the purpose when employing PCA, but may merely transform the data to a convenient (orthogonal) basis—thus we may have $\mathrm{rank}(\mathbf{X}) = \mathrm{rank}([\boldsymbol{v}_1 \dots \boldsymbol{v}_J])$. In this way we map the high dimensional input data vector onto a much smaller data vector of *projections*—hence, enforcing relations between elements of the weights connecting input to hidden units of the feed forward neural network, in other words we achieve a massive weight sharing. For a more detailed description see [2], [5]. Spelled out in terms of the neural network this can be written,

$$\begin{aligned} F(\mathcal{W}, \mathbf{B}, \boldsymbol{x}) &= F(\mathcal{W}, \mathbf{B}^T \boldsymbol{x}) \\ &= \sum_a W_a \tanh\left(\boldsymbol{w}_a^T \mathbf{B}^T \boldsymbol{x}\right) \end{aligned} \tag{1}$$

which is now a function of the input $\boldsymbol{x}$ projected on the set of $K \leq \mathrm{rank}(\mathbf{X})$ basis vectors[3] $\boldsymbol{b}_k$ forming the basis $\mathbf{B} = \left[\boldsymbol{b}_1 \dots \boldsymbol{b}_K\right]$ and a set of weight parameters $\mathcal{W} = \{W_a, \boldsymbol{w}_a\}$. The constrained weights are in turn optimized using a training set[4] $\mathcal{T} = \left\{(\boldsymbol{x}_j, y_j) \mid j = 1, \dots, J\right\}$ by minimizing the cost function with respect to $\mathcal{W}$

$$E(\mathcal{W}, \mathbf{B}, \mathcal{T}) = \frac{1}{J} \sum_{j=1}^{J} \left(y_j - F(\mathcal{W}, \mathbf{B}^T \boldsymbol{x}_j)\right)^2, \tag{2}$$

and we define:

> The saliency of input channel i (or pixel i if $\boldsymbol{x}$ is an image vector) is the change in the cost-function when the i'th input channel is removed.

This removal can be thought of as changing the basis vectors in $\mathbf{B}$, resulting in the new basis $\check{\mathbf{B}}^i$

$$\check{\boldsymbol{b}}^i_{k,i'} = \begin{cases} \boldsymbol{b}_{k,i'} & i' \neq i \\ 0 & i' = i \end{cases} \tag{3}$$

i.e. setting the i'th component[5] of all basis vectors to 0. Introducing this new basis, the model should be retrained to yield a new set of weight parameters $\check{\mathcal{W}}^i$. The saliency of input channel i is therefore

$$\delta E_i = E(\check{\mathcal{W}}^i, \check{\mathbf{B}}^i, \mathcal{T}) - E(\mathcal{W}, \mathbf{B}, \mathcal{T}). \tag{4}$$

If pruning is used to eliminate the effect of noise it should be applied to the full network prior to the calculation of the saliency map, so the retraining after removing the individual inputs conserves the network architecture.

Ideally one could estimate the change in generalization ability [11]. Such an estimate would—given a limited amount of data—be quite inaccurate, and since we only want to use the saliency map for comparing the relative input importance, it seems reasonable to consider only the change in the training error as indicated in equation (4).

Further approximations depend on the specific problem: In image processing the number of input channels (pixels) is often much greater than the number of examples, so that the computational burden of the direct computation of the saliency may be impractical. For such applications we develop

[1] The principal components as obtained from SVD (Singular Value Decomposition), or PCA (Principal Component Analysis). In either case the basis vectors correspond to the eigenvectors of the input data covariance matrix, see [4].

[2] In the sense of loosing information.

[3] See also section 2.1 for a more detailed explanation of the notation.

[4] The outputs are assumed scalar for simplicity.

[5] By the notation $\boldsymbol{b}_{k,i}$ we mean the i'th element of $\boldsymbol{b}_k$.

approximations of the saliency map using an expansion of the cost function. This is further described in section 2.1.

Finally, let us note that the saliency map as such is not confined to the ill-posed learning problem. In more conventional neural network applications, where the number of network inputs I is much smaller than the number of examples J, the saliency is similar to the *sensitivity measure* proposed in [14], [13] and [9], and to the Optimal Cell Damage Scheme suggested in [1]. In this case the removal of a single input may cause a notable change in the optimal weights thus making the I network retrainings essential (in contrast to the ill-posed case, as we shall see).

2.1. The Saliency Map in the Ill-Posed Case

As discussed a significant computational reduction can be obtained by projecting on the set of basis vectors $\mathbf{B}$ spanning the signal space[6] $\mathcal{S}$, if $I \gg J$.

It is easily seen [2], [5] that training in this case preserves signal space, i.e., if the initial weights of a hidden unit are confined to signal space they will stay there during training. This is a consequence of the fact that the cost function is independent of any component of the weight parameters outside signal space, $\mathcal{S}$, regardless of the basis $\mathbf{B}$ used for representing the data, as long as $\mathbf{B}$ spans $\mathcal{S}$.

After preprocessing the neural network is not fed the actual pixel data, but the projection of the images on the basis $\mathbf{B}$. This justifies the notation $F(\mathcal{W}, \mathbf{B}^T\boldsymbol{x}_j)$ for the model, in that the model can be said to be working on the projected data $\boldsymbol{v}_j = \mathbf{B}^T\boldsymbol{x}_j$.

2.1.1. Approximating the Saliency Map

If the number of input channels I is large, the task of retraining I networks—i.e. to compute $\check{\mathcal{W}}^i$ as implied by equation (4)—is immense. In this section some approximations are presented to speed up the computation.

The second order expansion of the cost function with respect to the basis vectors and the weight vector $\boldsymbol{u} = \left[\boldsymbol{w}_1^T \ldots \boldsymbol{w}_A^T \; W_1 \ldots W_A\right]^T$ consisting of all the parameters in $\mathcal{W}$ is given by

$$\begin{aligned}\delta E \;\simeq\; & \sum_{k=1}^{K} \frac{\partial E}{\partial \boldsymbol{b}_k^T}\delta\boldsymbol{b}_k + \frac{\partial E}{\partial \boldsymbol{u}^T}\delta\boldsymbol{u} \\ & + \frac{1}{2}\sum_{k=1}^{K}\delta\boldsymbol{b}_k^T\frac{\partial^2 E}{\partial\boldsymbol{b}_k\partial\boldsymbol{b}_k^T}\delta\boldsymbol{b}_k + \frac{1}{2}\delta\boldsymbol{u}^T\frac{\partial^2 E}{\partial\boldsymbol{u}\partial\boldsymbol{u}^T}\delta\boldsymbol{u} \\ & + \sum_{k=1}^{K}\delta\boldsymbol{b}_k^T\frac{\partial^2 E}{\partial\boldsymbol{b}_k\partial\boldsymbol{u}^T}\delta\boldsymbol{u}, \qquad (5)\end{aligned}$$

where $\delta\boldsymbol{b}_k$ is the change in the k'th basis vector, and $\delta\boldsymbol{u}$ is the change in the optimal weight parameters, due to the changed basis. If the network is fully trained $\frac{\partial E}{\partial \boldsymbol{u}} = \mathbf{0}$ so the second term vanishes[7].

In the ill-posed case, modeling will only be meaningful if the stochastic part of the signal is highly correlated, i.e., the individual pixels are spatially correlated. Thus it can be assumed that the term $\delta\boldsymbol{u}$ roughly scales inversely with the number of inputs, i.e. as $1/I$. We therefore neglect all terms scaling with $\delta\boldsymbol{u}$ yielding

$$\delta E \simeq \sum_{k=1}^{K}\frac{\partial E}{\partial\boldsymbol{b}_k^T}\delta\boldsymbol{b}_k + \frac{1}{2}\sum_{k=1}^{K}\delta\boldsymbol{b}_k^T\frac{\partial^2 E}{\partial\boldsymbol{b}_k\partial\boldsymbol{b}_k^T}\delta\boldsymbol{b}_k, \qquad (6)$$

thus eliminating the effect of retraining, effectively estimating $\delta E_i = E(\mathcal{W}, \check{\mathbf{B}}^i, \mathcal{T}) - E(\mathcal{W}, \mathbf{B}, \mathcal{T})$ c.f. equation (4). This is in line with the Optimal Brain Damage scheme [7] for estimating weight saliency and the approximation is indeed supported by the numerical example. Since we compute the saliency for one input channel at a time, the off-diagonal elements of $\frac{\partial^2 E}{\partial\boldsymbol{b}_k\partial\boldsymbol{b}_k^T}$ vanish, so we finally get

$$\delta E_i \simeq \sum_{k=1}^{K}\frac{\partial E}{\partial b_{k,i}}\delta b_{k,i} + \frac{1}{2}\sum_{k=1}^{K}\frac{\partial^2 E}{\partial b_{k,i}^2}\delta b_{k,i}^2. \qquad (7)$$

For the two-layer network specified in equation (1), with $h_{aj} = \tanh\left(\boldsymbol{w}_a^T\mathbf{B}^T\boldsymbol{x}_j\right)$ we find[8]

$$\begin{aligned}\frac{\partial E}{\partial b_{k,i}} &= -\frac{2}{J}\sum_{j=1}^{J}\Big[\left(y_j - F(\mathcal{W}, \mathbf{B}^T\boldsymbol{x}_j)\right) \\ &\qquad \cdot \sum_a W_a(1-h_{aj}^2)w_{a,k}x_{j,i}\Big] \\ &= -\frac{2}{J}\sum_{j=1}^{J} e_j s_{jk} x_{j,i} \qquad (8)\end{aligned}$$

where we have introduced the quantities $e_j = y_j - F(\mathbf{B}^T\boldsymbol{x}_j, \mathcal{W})$ and $s_{jk} = \sum_a W_a(1-h_{aj}^2)w_{a,k}$. By further invoking the Gauss-Newton approximation $\left(\frac{\partial^2 E}{\partial\boldsymbol{b}_k\partial\boldsymbol{b}_k^T} \simeq \sum_{j=1}^{J}\frac{\partial F}{\partial\boldsymbol{b}_k}\frac{\partial F}{\partial\boldsymbol{b}_k^T}\right)$ for least squares problems, see e.g. [7], yielding

$$\frac{\partial^2 E}{\partial b_{k,i}^2} \simeq \frac{2}{J}\sum_{j=1}^{J} s_{jk}^2 x_{j,i}^2, \qquad (9)$$

[6] We denote the space spanned by the input vectors $\boldsymbol{x}_j$ in the training set $\mathcal{T}$ by signal space $\mathcal{S} = \mathrm{span}\{\boldsymbol{x}_j\}$.

[7] If we eliminate overfitting by pruning the network, i.e. forcing some parameters $\boldsymbol{u}'$ to $\mathbf{0}$, only the remaining parameters $\boldsymbol{u}^\star = \boldsymbol{u} \setminus \boldsymbol{u}'$ are optimized so that $\frac{\partial E}{\partial \boldsymbol{u}^\star} = \mathbf{0}$. On the other hand, we will generally have $\frac{\partial E}{\partial \boldsymbol{u}'} \neq \mathbf{0}$, which may cause negative estimates of the saliency. This can be explained as follows: If the network models from a subspace of $\mathcal{S}$, called model-space $\mathcal{M}$, one might say that the basis change in (3) perturbs signal space, so that some of the noise eliminated by pruning re-enters $\mathcal{M}$. Sometimes this will allow the model to perform better on the training set, thus yielding negative saliencies. We therefore choose to interpret these as zero.

[8] Again $w_{a,k}$ means the k'th element of $\boldsymbol{w}_a$, and $x_{j,i}$ the i'th element of $\boldsymbol{x}_j$.

and since we remove only one input channel in the basis, i.e. $\delta \boldsymbol{b}_{k,i} = -\boldsymbol{b}_{k,i}$, we get

$$\delta E_i = \frac{2}{J}\sum_{k=1}^{K}\sum_{j=1}^{J} e_j s_{jk}\boldsymbol{x}_{j,i}\boldsymbol{b}_{k,i} + \frac{1}{J}\sum_{k=1}^{K}\sum_{j=1}^{J} s_{jk}^2 \boldsymbol{x}_{j,i}^2 \boldsymbol{b}_{k,i}^2. \tag{10}$$

as the estimate of the saliency map.

3. Ill-posed Example: Modeling from PET images

We now proceed to demonstrate the practical use of the saliency map. Positron-Emission-Tomography (PET) is a way of indirectly measuring the neural activity of different regions of the human brain, resulting in 3-dimensional images. As the dimension of the images is very large, affine preprocessing (projection of the data on the corresponding PCA-basis) is applied, thus reducing the computational requirement of the modeling.

More specifically, we first examined 64 PET-scans of 8 subjects, each scanned 8 times, exposed to 8 different levels of saccadic eye movement activation [6]. We thus analyze $J = 64$ image vectors of $I = 128 \times 128 \times 48 = 768432$ voxels[9].

A two-layer feed-forward neural network was trained to predict the paradigm activation level (the frequency of the saccadic eye movements) from the 64 3-dimensional brain volumes.

An estimated saliency map was computed employing the approximation in equation (10). In figure 1 iso surfaces (surfaces of equal saliency) capturing the most salient voxels are depicted as bright bodies floating in a box. To help localize the salient areas, slices of a corresponding anatomical brain image (an MR scan) are shown on the walls of the box, with the shadows of the salient bodies projected in black. The slices correspond to the middle of the brain, one in each of the three dimensions.

The result is in correspondence with what has been found using other analysis methods—e.g. Statistical Parametric Mapping (SPM), and the Scaled Subprofile Model (SSM)—on the same data [6], [12]. The larger cluster of salient pixels, as seen in the back of the brain, is identified as the *visual cortex.*

To demonstrate the accuracy of the 1st and 2nd order approximations of the saliency, c.f. equation (10), we computed the images shown in figure 2. The first column shows the true change in the cost function[10] for horizontal slices through the volume corresponding the AC-PC[11] level -17mm, AC-PC, and AC-PC + 17mm. This corresponds to expanding E to infinitely high order with respect to $\boldsymbol{b}$. The second and third columns are the 1st and 2nd order

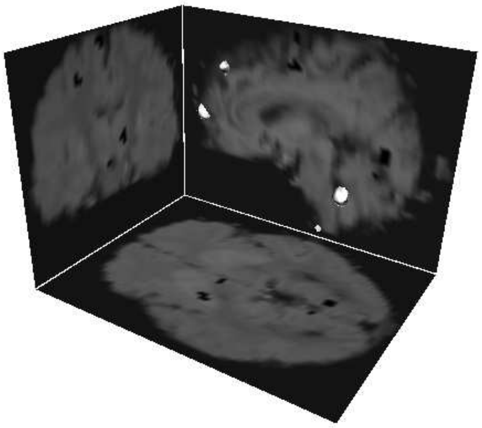

Fig. 1: Using the saliency map to asses paradigm related brain regions in the saccadic eye movement task. The most salient voxels are depicted as iso surfaces (surfaces of equal saliency) here seen as bright bodies floating in a box with slices of a corresponding anatomical brain scan depicted on the walls. Shadows of the iso surfaces are projected in black on the walls. The larger cluster in the back of the brain is the *visual cortex.*

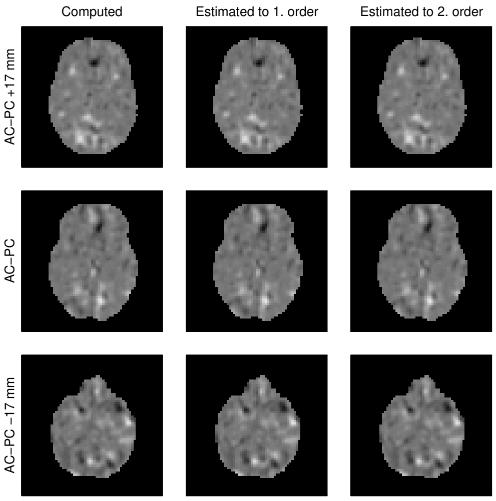

Fig. 2: From left to right: Computed saliency map, 1st order, and 2nd order approximations, all for 3 different slices of the brain. The slices correspond to the AC-PC level -17 mm, the AC-PC level and the AC-PC level + 17mm. Bright areas have high saliencies. In the specific case (I = 34863 pixels) all columns are almost identical—thus validating the approximations. In fact, the 2nd order term seems visually negligible.

[9] Of these a large portion is masked out, leaving vectors of "only" active 34863 voxels.

[10] Computed as the change in the cost function *without* retraining $\delta E_i = E(\mathcal{W}, \tilde{\mathbf{B}}^i, \mathcal{T}) - E(\mathcal{W}, \mathbf{B}, \mathcal{T})$, so that only the effects of neglecting the higher order 'pure' $\delta \boldsymbol{b}_k$ terms of (5) and (6) are assessed.

[11] Anterior Comisura - Posterior Comisura, which are easily identified centers in the brain, and thus used for reference.

approximations of (10). It is evident, that even the 1st order term alone is a useful approximation in the case of $I = 34863$ voxels.

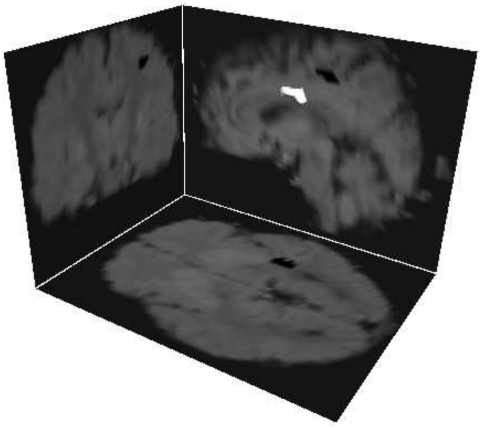

Fig. 3: Saliency map of the finger opposition task. The most salient voxels are depicted as iso surfaces (surfaces of equal saliency) here seen as bright bodies floating in a box with slices of a corresponding anatomical brain scan depicted on the walls. Shadows of the iso surfaces are projected in black on the walls. The salient area identified is the *primary sensory-motor cortex.*

Secondly, the saliency map was computed for a neural network modeling the finger opposition task, which involves areas of the brain controlling motion. The data has previously been analyzed in [12]. Again, 8 subjects were scanned 8 times each, 4 times resting and 4 times doing the finger opposition task. Thus the paradigm is on/off corresponding to a problem of classification[12]. Figure 3 shows the saliency map in a manner similar to figure 1. The method clearly identifies the area known as the *primary sensory-motor cortex.*

Further, we investigated the effect of the dimension of the input-field I, on the approximation (10). For simplicity this is done on a single slice, which is sub-sampled to yield $Q = 9$ datasets with decreasing I. After performing the entire modeling procedure Q times, we measure as a function of I the normalized mean squared error for both the 1st and 2nd order expansions, i.e

$$
\begin{aligned}
f_1(I) &= \frac{\sum_{i=1}^{I} \left(\delta E_{c,i} - \delta E_{1,i}\right)^2}{\sum_{i=1}^{I} \delta E_{c,i}^2} \\
f_2(I) &= \frac{\sum_{i=1}^{I} \left(\delta E_{c,i} - \delta E_{2,i}\right)^2}{\sum_{i=1}^{I} \delta E_{c,i}^2} \\
\delta E_{1,i} &= \sum_{k=1}^{K} \frac{\partial E}{\partial \boldsymbol{b}_{k,i}} \delta \boldsymbol{b}_{k,i} \qquad (11) \\
\delta E_{2,i} &= \delta E_{1,i} + \frac{1}{2} \sum_{k=1}^{K} \frac{\partial^2 E}{\partial \boldsymbol{b}_{k,i}^2} \delta \boldsymbol{b}_{k,i}^2 \\
\delta E_{c,i} &= E(\mathcal{T}, \mathcal{W}, \check{\mathbf{B}}^i) - E(\mathcal{T}, \mathcal{W}, \mathbf{B})
\end{aligned}
$$

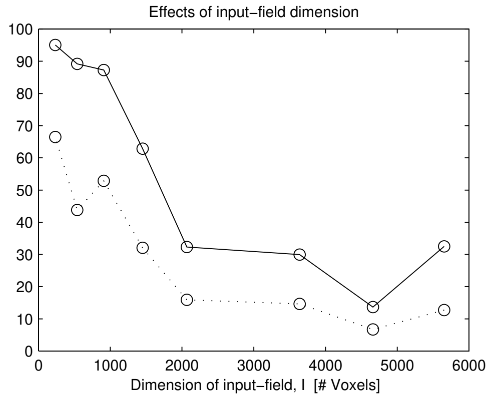

Fig. 4: Normalized mean squared error of the 1st (—) and 2nd ($\cdots$) order approximations of the saliency. With increasing input-field dimension I, the errors decrease—for large I the 1st order approximation suffices.

These quantities are shown in figure 4. We see that the error introduced by the approximations decreases when I gets large. Further, for very large I, the 2nd order term seems negligible. This is in line with the visual impression of figure 2.

Finally, let us note that the saliency map easily computes for linear models as well.

4. Discussion

We have proposed the saliency map as a new method for understanding and visualizing feed-forward neural networks. Furthermore, several levels of approximations have been derived providing significant computational savings. The viability of the approach was demonstrated on a series of 3D brain activation volumes.

Though the emphasis has been on the so-called ill-posed case, the proposed technique can easily be

[12] Note that for classification problems better optimization schemes (costfunctions) exist, see e.g [3].

applied to the more standard setting, i.e. the well-posed case.

5. Acknowledgments

This research has been supported by the Danish Research Councils for the Natural and Technical Sciences through the Danish Computational Neural Network Center CONNECT, and the US National Institutes of Health's Human Brain Project through grant DA09246.

References

[1] T. Cibas *et al.*, "Variable selection with optimal cell damage," *Proceedings of the International Conference on Artificial Neural Networks, pp. 727–730*, 1994.

[2] L. K. Hansen, B. Lautrup, I. Law, N. Mørch, and J. Thomsen, "Extremely ill-posed learning," CONNECT *Preprint.* Available via anonymous ftp `ei.dtu.dk : dist/hansen.ill-posed.ps.Z.`, Aug. 1994.

[3] M. Hintz-Madsen *et al.*, "Design and evaluation of neural classifiers - application to skin lesion classification," *To appear: 1995 IEEE Workshop on Neural Networks for Signal Processing (NNSP'95).* Available via anonymous ftp `ei.dtu.dk : dist/1995/hintz.nnsp95.ps.Z`, 1995.

[4] J. E. Jackson, *A User's Guide to Principal Components.* Wiley Series on Probability and Statistics, John Wiley and Sons, 1991.

[5] B. Lautrup, L. K. Hansen, I. Law, N. Mørch, C. Svarer, and S. Strother, "Massive weight sharing: A cure for extremely ill-posed problems," in *Proceedings of Workshop on Supercomputing in Brain Research: From Tomography to Neural Networks, HLRZ, KFA Jülich, Germany*, (H. J. Hermann, D. E. Wolf, and E. P. Pöppel, eds.), pp. 137–148, Nov. 1994.

[6] I. Law *et al.*, "A characterization of the frequency related cerebral response during sensory-guided saccades," *In preparation*, 1995.

[7] Y. Le Cun, J. S. Denker, and S. Solla, "Optimal brain damage," *Advances in Neural Information Processing Systems 2, pp. 598–605*, 1990.

[8] Le Cun, Y., I. Kanter, and S. Solla, "Eigenvalues of covariance matrices: Application to neural-network learning," *Physical Review Letters*, vol. 66, Number 18:pp 2396–2399, May 1991.

[9] J. Moody, "Prediction risk and architecture selection for neural networks," in *From Statistics to Neural Networks, Theory and Pattern Recognition Applications*, (V. Cherkassky, J. H. F. H., and H. Wechsler, eds.), pp. 147–165, Springer Verlag, 1992.

[10] J. S. Orfanidis, "Gram-schmidt neural nets," *Neural Computation*, vol. 2, pp 116–126, 1990.

[11] M. W. Pedersen, L. K. Hansen, and J. Larsen, "Pruning with generalization based saliencies: γOBD, γOBS," *Submitted to: Advances in Neural Information Processing Systems (NIPS95).* Available via anonymous ftp `ei.dtu.dk : dist/1995/with.nips95.ps.Z`, 1995.

[12] S. C. Strother, J. R. Anderson, K. A. Schaper, J. J. Sidtis, J. S. Liow, R. P. Woods, and D. A. Rottenberg, "Principal component analysis and the scaled subprofile model compared to intersubject averaging and statistical parametric mapping: I. "Functional connectivity" of the human motor system studied with [15-o]water pet," *Journal of Cerebral Blood Flow and Metabolism*, vol. 15, pp. 738–753, 1995.

[13] J. Utans and J. Moody, "Principled architecture selection for neural networks: Application to corporate bond rating prediction," *Advances in Neural Information Processing Systems 4*, 1991.

[14] J. Utans and J. Moody, "Selecting neural network architectures via the prediction risk: Application to corporate bond rating prediction," *Proc. First International Conference in Artificial Intelligence Applications on Wall Street*, 1991.

Appendix E

Contribution to HBM'96

This appendix contains the abstract "Generalization Performance of Nonlinear vs. Linear Models for $[^{15}O]$water PET Functional Activation Studies" (Mørch et al., 1996b), orally presented at the Second International Conference on Functional Mapping of the Human Brain, 1996 (HBM'96) in Boston, USA.

Generalization Performance of Nonlinear vs. Linear Models for [^{15}O]water PET Functional Activation Studies

N. Mørch[1,2], L.K. Hansen[2], S.C. Strother[3,4], I. Law[1], C. Svarer[1], B. Lautrup[5], J.R. Anderson[3], N. Lange[6] and O.B. Paulson[1]

[1]Department of Neurology, National University Hospital, Copenhagen, Denmark
[2]Department of Mathematical Modelling, Technical University of Denmark, Denmark
[3]Veterans Affairs Medical Center and the [4]University of Minnesota, Minneapolis, USA
[5]Niels Bohr Institute, Denmark and [6]National Institutes of Health, Bethesda, USA

Introduction. We use empirical measures of predictive performance (i.e., generalization error) to demonstrate a sig-nificant improvement in a nonlinear model (artificial neural network, ANN) over a linear model. We obtained improved predictive performance for the analysis of a set of 64 [^{15}O]water PET scans of 8 subjects performing a saccadic eye-movement task (1). A recently developed visualization tool for ANN's, the "saliency map," provides spatial patterns for the optimized nonlinear model (2) that may be compared with linear techniques, e.g., SSM/PCA (3).

Methods. *Generalization* refers to a quantitative measure of the extent to which model parameters estimated from one (training) dataset predict the structure of another (test) dataset. Generalization may be defined for a finite training set of functional neuroimages, x. With a label, y, attached to each neuroimage (e.g., activation/rest) the modeling problem is to estimate the joint distribution p(x,y) expressed as a mapping between neuroimages and labels through a set of model parameters, θ. For a fixed number of neuroimages the *bias-variance trade-off* within a family of models relates to model complexity. If a model is too simple (i.e., the dimension of θ is too small), it is biased and makes systematic prediction errors, whereas if a model is too complex (i.e., the dimension of θ is too large), it will overfit and produce unreliable predictions. In order to estimate generalization errors we split the saccadic scans into a training set containing 6 scans from each of the subjects, and a test set containing the remaining 2 scans for each subject. Using the training set, families of linear models and ANN's were estimated. Starting with a complex model, parameters were eliminated sequentially ("pruned", 4) in order to select a model with optimal performance (i.e., minimum empirical generalization error) measured using the test set. Such generalization error estimates are unbiased because they are based on an independent test set, and stochastic because they depend on the training set via the estimated θ. Techniques are being developed for rigorously comparing generalization error estimates within model families. Optimized ANN models may be visualized using saliency maps, which depict the relative importance of individual voxels in model generalization (2).

Results. In panel B of Fig. 1, the mean of the estimated generalization error is depicted for both linear and ANN model families. The optimal ANN performs significantly better than its optimal linear counterpart (i.e., smallest generalization error in lower-right graph). In panel A we compare the optimal ANN model's saliency map with functional activation images from SPM'95 (6) and SSM/PCA (3). The spatial patterns are reproducible in that all methods produce the expected visual/occipital cortical activation, but there are important similarities and differences between the saliency map and the other models' patterns that need to be further investigated.

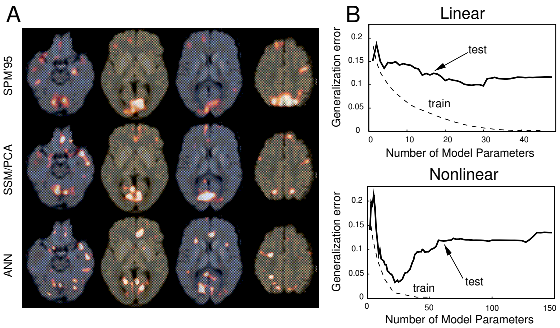

Figure 1. Functional activation patterns with bias-variance plots

Conclusions. It is important to assess the predictive performance of a model of functional activation based on test data not used to estimate the model parameters. We have shown that nonlinear artificial neural network models have better generalization performance than linear models. Our results demonstrate that "complex" nonlinear models, such as ANN's, may have an important role to play in the analysis of functional activation datasets.

Acknowledgements. Funded in part by Human Brain Project R01 DA092461, the Danish Research Councils for the Natural and Technical Sciences through the Computational Neural Network Center, and the Danish Research Council for Medical Science.

References.

1. Law I, Svarer C, Paulson OB. Hum. Brain Mapp. (Suppl. 1):323, 1995
2. Mørch N, Kjems U, Hansen LK, et al. In: Proc. IEEE Int. Conf. on Neural Networks (ICNN'95), 4:2085-2089
3. Strother SC, Anderson JR, Schaper KA, et al. J Cereb. Blood Flow Metab. 15:738-775, 1995
4. Svarer C, Hansen LK, Larsen J. In: Proc. IEEE Conf. on Neural Networks (ICNN'93), pp. 45-51
5. Larsen J, Hansen LK. In: Proc. IEEE Workshop on Neural Networks for Signal Processing (NNSP'95), pp. 30-39
6. Friston KJ, Holmes AP, Worsley KJ, et al. Hum. Brain Mapp. 2:189-210, 1995

Appendix F

Contribution to IPMI'97

This appendix contains the paper "Nonlinear versus Linear Models in Functional Neuroimaging: Learning Curves and Generalization Cross-Over" (Mørch et al., 1997), orally presented at the 15th International Conference on Information Processing in Medical Imaging 1997 (IPMI'97) in Vermont, USA.

Nonlinear versus Linear Models in Functional Neuroimaging: Learning Curves and Generalization Crossover

Niels Mørch[1,2], Lars K. Hansen[2], Stephen C. Strother[3], Claus Svarer[1]
David A. Rottenberg[3], Benny Lautrup[4], Robert Savoy[5], Olaf B. Paulson[1]

[1] Neurobiology Research Unit
Copenhagen University Hospital, Rigshospitalet
DK-2100 Copenhagen Ø, Denmark

[2] Department for Mathematical Modelling
Technical University of Denmark
DK-2800 Lyngby, Denmark

[3] Radiology and Neurology Departments
University of Minnesota
and
PET Imaging Service
Minneapolis VA Medical Center
Minnesota, 55417, USA

[4] Niels Bohr Institute
University of Copenhagen
DK-2100 Copenhagen Ø

[5] Massachusetts General Hospital
Boston, Massachusetts, USA

Abstract. We introduce the concept of generalization for models of functional neuroactivation, and show how it is affected by the number, N, of neuroimaging scans available. By plotting generalization as a function of N (i.e. a "learning curve") we demonstrate that while simple, linear models may generalize better for small N's, more flexible, low-biased nonlinear models, based on artificial neural networks (ANN's), generalize better for larger N's. We demonstrate that for sets of scans of two simple motor tasks—one set acquired with $[O^{15}]$water using PET, and the other using fMRI—practical N's exist for which "generalization crossover" occurs. This observation supports the application of highly flexible, ANN models to sufficiently large functional activation datasets.
Keywords: Multivariate brain modeling, ill-posed learning, generalization, learning curves.

1 Introduction

Datasets that result from functional activation studies of the living, human brain typically consist of two corresponding sets of observables, the *microscopic* and the *macroscopic* [26]. The brains haemodynamic response, reflecting the microscopic neuronal firing pattern, is measured by modern three-dimensional (3D) imaging techniques such as positron emission tomography (PET) and functional magnetic resonance imaging (fMRI) by integrating in space and time [21]. Along with the resulting set of 3D image volumes (scans) a corresponding set of macroscopic descriptors governs the overall conditions of the experiment. This set can include experimentally controlled factors, such as paradigm labels and variables, and physiological and demographic measures, such as age and heart-rate. The micro- and macroscopic observables are generally both sets of multivariate, stochastic variables. Arranging the microscopic variables (the 3D image volumes) in vectors $\mathbf{x}$ and the macroscopic variables in vectors $\mathbf{g}$ a functional activation dataset $\mathcal{D}$ consisting of N observations can be written as

$$\mathcal{D} = \{(\mathbf{x}_j, \mathbf{g}_j) \mid j = 1, \ldots, N\} \ . \tag{1}$$

Generally, we will assume the observations to be random, independent samples of an underlying stationary process with distribution $\mathrm{P}(\mathbf{x}, \mathbf{g})$. As we shall see this distribution plays a central role in the analysis of functional activation datasets [18].

In the following we discuss the so-called "curse of dimensionality" that results from the extremely ill-posed nature of typical functional activation datasets [6,23]. The problem is discussed in terms of probability density estimation and we briefly mention ways to remedy the inevitable over-parameterization that otherwise occurs in modeling procedures based on such datasets [12]. The main point we hope to convey is how model generalization—as studied intensively in other fields dealing with probability density estimation and multivariate modeling [8,13,17,20]—applies to functional neuroimaging [18], and specifically how it is affected by the number, N, of available observations.

2 Models of Functional Activation Datasets

In terms of $\mathbf{x}$ and $\mathbf{g}$ the analysis of functional activation datasets can be phrased as the estimation (of properties) of $\mathrm{P}(\mathbf{x}, \mathbf{g})$. For instance, we can estimate the conditional mean, $\mathrm{E}\{\mathbf{x}|\mathbf{g}\}$, using multivariate linear models as in [7], thus effectively modeling the expected scan from a set of macroscopic variables. Or, we can estimate the alternative conditional mean $\mathrm{E}\{\mathbf{g}|\mathbf{x}\}$, using multivariate linear models as in [18], effectively modeling the expected value of a set of macroscopic variables from the scan[1].

In general, we employ parameterized models of the properties we wish to estimate. In this work we focus on models that estimate $\mathrm{E}\{\mathbf{g}|\mathbf{x}\}$. Being a function of $\mathbf{x}$ we denote these models $f_\theta(\mathbf{x})$, explicitly indicating the dependency on the set of parameters θ. Parameter values are estimated using some or all of the available data. We call such a set of data used for parameter estimation the *training set*,

$$\mathcal{D}_{train} = \{(\mathbf{x}_j, \mathbf{g}_j) \mid j = 1, \ldots, N_{train}\} \ . \tag{2}$$

[1] In fact, it can be shown that the two linear models are analogous and simple relations between the parameters exist.

For a given set of parameters model performance is quantified using the *cost function*, $\mathrm{c}(\mathbf{x}, \mathbf{g}, \theta)$, which is often derived from maximum likelihood (ML) arguments [4,10,14]. Parameter values are estimated by optimizing the cost function based on the observations in the training set (we say that the model is trained, hence the name). Averaged over the training set this evaluates to

$$\mathrm{C}(\mathcal{D}_{train}, \theta) = \iint \mathrm{c}(\mathbf{x}, \mathbf{g}, \theta) \mathrm{P}_{train}(\mathbf{x}, \mathbf{g}) \, \mathrm{d}\mathbf{x} \, \mathrm{d}\mathbf{g} \ . \tag{3}$$

By using the empirical density estimate $\mathrm{P}_{train}(\mathbf{x}, \mathbf{g}) = \frac{1}{N_{train}} \sum_{j=1}^{N_{train}} \delta(\mathbf{x} - \mathbf{x}_j, \mathbf{g} - \mathbf{g}_j)$ we get the so-called *training error*

$$\mathrm{C}(\mathcal{D}_{train}, \theta) = \frac{1}{N_{train}} \sum_{j=1}^{N_{train}} \mathrm{c}(\mathbf{x}_j, \mathbf{g}_j, \theta), \quad (\mathbf{x}_j, \mathbf{g}_j) \in \mathcal{D}_{train} \ . \tag{4}$$

The choice of cost function will depend on the noise model and potential constraints we impose on the model outputs (e.g. to make them interpretable as probabilities). For more details on these issues see [3,10,14].

Equipped with a training set, a model, and a cost function we are ready to gain knowledge about $\mathrm{P}(\mathbf{x}, \mathbf{g})$ and, hopefully, underlying information processing relationships in the human brain. However, several important additional issues must be considered before attempting to build practical models. Rather than using (4) to model $\mathrm{E}\{\mathbf{g}|\mathbf{x}\}$ from the observations directly we can reduce the computational burden dramatically by taking the extremely ill-posed nature of typical functional activation datasets into account.

2.1 Ill-posed Datasets

While we often include only a few descriptors in the macroscopic variables $\mathbf{g}$ making them low-dimensional, the microscopic variables $\mathbf{x}$ that represent the scans are often high-dimensional. Despite preprocessing that, among other things, mask out voxels outside the brain more than 40000 voxels often remain. Using $\mathcal{I}$ to denote the space in which all possible observations fall (i.e., the *input space*) we have $\dim(\mathcal{I}) \sim 10^4$. The space spanned by the actual observations in the dataset is called *signal space* and denoted $\mathcal{S}$. Often no more than a few hundred observations are available, so $\dim(\mathcal{S}) \sim 10^2$.

Typically $\dim(\mathcal{S}) \ll \dim(\mathcal{I})$, making $\mathcal{S}$ a small subspace of $\mathcal{I}$. This is exactly what characterizes extremely ill-posed datasets. In Fig. 1 an ill-posed situation is illustrated. Input space is 3D Euclidean space indicated by the dashed vectors. With only two observations in the dataset represented by the solid vectors, signal space is a 2D subspace, i.e. a plane. The dataset does not contain information about the parts of $\mathcal{I}$ that are orthogonal to $\mathcal{S}$.

Because the dimension of $\mathcal{S}$ is low we have a correspondingly low number of degrees of freedom available in any subsequent modeling, and naive estimation based directly on the observation pairs $(\mathbf{x}, \mathbf{g})$ will result in strong linear relations between the estimated parameters; the original basis in which observations in

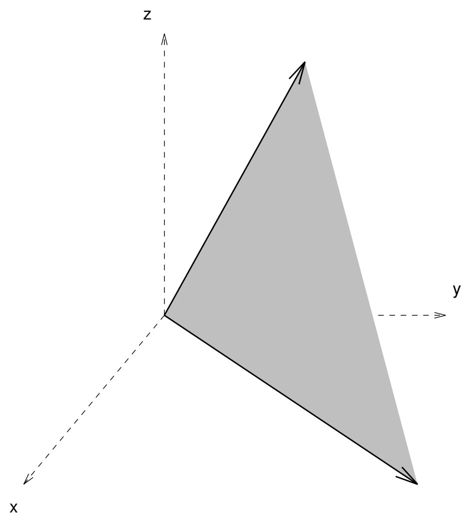

Fig. 1. Illustration of an ill-posed dataset. With input space, $\mathcal{I}$, being three-dimensional (represented by the dashed vectors) the signal space, $\mathcal{S}$, which is the space spanned by the two observations in the dataset (represented by the solid vectors), is the plane indicated in gray. The dataset contains no information about the parts of input space that are orthogonal to signal space because $\dim(\mathcal{S}) < \dim(\mathcal{I})$.

input space are represented is a poor choice when it comes to representing observations efficiently in signal space. We can easily construct other, more efficient bases, however, that reduce the dimensionality of the *representation* without loss of information [12,19]. The only requirement is that the basis chosen spans signal space. One particularly choice of basis is to use the observations in the dataset themselves. Even-though efficient in reducing an extremely ill-posed problem to an only marginally ill-posed one bases that reveal more about the signal structure are available. In particular, a singular value decomposition (SVD) basis [11,15,16] has been shown to reveal an interesting subspace structure [12,22,23]. In the following $\mathbf{v}$ will denote the projection of a scan $\mathbf{x}$ onto an efficient basis that spans signal space; for more details see [18].

2.2 Model Flexibility and Bias

Having reduced the extremely ill-posed dataset to a marginally ill-posed one where the dimension of each observation, $\mathbf{v}$, equals the number of observations, it is now part of the modeling task to impose further constraints in order to avoid over-fitting. Different model families approach this in various ways, by limiting model flexibility and thus the effective dimensionality of the parameterization to match the available degrees of freedom.

In the following we focus on models for classification. Assuming the macroscopic variables to be univariate labels we seek to build models that optimally classify the microscopic variables[2], $\mathbf{x}$, into the correct classes. In other words, we seek a *decision boundary* in signal space that allows the observations to be correctly classified according to their macroscopic labels. More specifically we will apply two model families that differ in model flexibility:

- **Fishers Linear Discriminant**
  Fishers Linear Discriminant (FLD) is a family of linear classifier that are based on a cost function that measures the difference between class means relative to the within class variance [4,14]. The term linear refers to the fact that the models are linear in the parameters which makes parameter estimation straight forward. However, this relatively high *bias* limits the flexibility of the relationships (decision boundaries) that the models can implement.

- **Artificial neural network (ANN) classifiers**
  Artificial neural networks is a family of parameter efficient models that deal with the curse of dimensionality by employing nonlinearities [2,9]. The models are nonlinear in the parameters in contrast to FLD. This complicates parameter estimation but makes the models less biased and allow them to implement a much more flexible and wider range of relationships (decision boundaries) [10,24].

3 Generalization

Although cost functions allow us to quantify model performance the training error in (3) is the average over the *specific* and *limited* training set only. If the distribution of observations in this set, $\mathrm{P}_{train}(\mathbf{x},\mathbf{g})$, does not match the true distribution, $\mathrm{P}(\mathbf{x},\mathbf{g})$, sufficiently well the cost function value will not reflect model performance correctly. Rather, as training sets are often small we should use *generalization error*,

$$\mathrm{G}(\theta_{train}) = \iint \mathrm{c}(\mathbf{x},\mathbf{g},\theta_{train})\mathrm{P}(\mathbf{x},\mathbf{g})\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{g} \quad . \tag{5}$$

as our measure of model quality. Unfortunately this requires complete knowledge of $\mathrm{P}(\mathbf{x},\mathbf{g})$ which, of course, we do not have. Instead we can estimate generalization either analytically [1,20] or empirically [24]. The latter is often called *test error*

$$\widehat{\mathrm{G}}(\theta_{train}) = \mathrm{C}(\mathcal{D}_{test},\theta_{train}) \tag{6}$$

$$= \frac{1}{N_{test}} \sum_{j=1}^{N_{test}} \mathrm{c}(\mathbf{x}_j,\mathbf{g}_j,\theta_{train}), \quad (\mathbf{x}_j,\mathbf{g}_j) \in \mathcal{D}_{test} \tag{7}$$

and evaluated using an independent set of observations organized in a *test set*

$$\mathcal{D}_{test} = \{(\mathbf{x}_j,\mathbf{g}_j) \mid j = 1,\ldots,N_{test}\} \quad . \tag{8}$$

[2] In practice we use $\mathbf{v}$ of course, thus efficiently representing the scans using a basis that spans signal space.

In (5) we have indicated how generalization error depends on the training set via the estimated parameters θ_{train}. To eliminate this dependency we average over training sets of size N_{train} to yield the *expected generalization error*

$$\mathrm{E}_{N_{train}}(\mathrm{G}) = \int \mathrm{G}(\theta_{train})\mathrm{P}(\mathcal{D}_{N_{train}})\,\mathrm{d}\mathcal{D}_{N_{train}} \quad , \tag{9}$$

which can be estimated empirically by using the test error in (7) to estimate $\mathrm{G}(\theta_{train})$. Clearly, using a set of the available observations to independently estimate generalization reduces the number of observations left for training. The optimal split of the available data into training- and test sets constitutes a non-trivial problem that has been studied in the context of ANN's and statistical re-sampling techniques [5]. In the remainder of this paper we will fix the size of the test set as well as the observations therein to allow measures of model performance that are unbiased—or at least comparable between different model families.

3.1 Learning Curves and Generalization Crossover

Using generalization we are now ready to investigate how the number of observations in the training set, N_{train}, affects model performance. We hypothesize that, as N_{train} increases, generalization error will decrease. This downwards slope of the so-called *learning curve* is caused by the improved estimates of $\mathrm{P}(\mathbf{x}, \mathbf{g})$ (on which the models are based) that increasingly larger training sets provide.

For a given model family the learning curve will eventually flatten out as additional observations no longer improve model performance due to limitations in the models themselves. This naturally leads to the further hypothesis that learning curves look different for different model families. Models that are very flexible typically need many examples to obtain stable parameter estimates. These models will in return generalize very well. In contrast, the implicit constraints in highly biased models enable them to obtain stable parameter estimates from fewer observations. However, they may not generalize as well as their more flexible counterparts. Thus, while generalization error is highest for very flexible models for small training sets, it decreases to a lower level than for highly biased, less flexible models as N_{train} increases. This means that a *generalization crossover* occurs at which point the data support the use of the more flexible models. The situation is illustrated in Fig. 2.

4 Methods

To estimate learning curves data from two functional activation studies, both involving simple motor tasks, was used.

4.1 [O^{15}]Water PET Scanning

A set of 30 subjects were each scanned 8 times using a Siemens-ECAT 953B PET scanner while alternately resting and performing a simple finger opposition

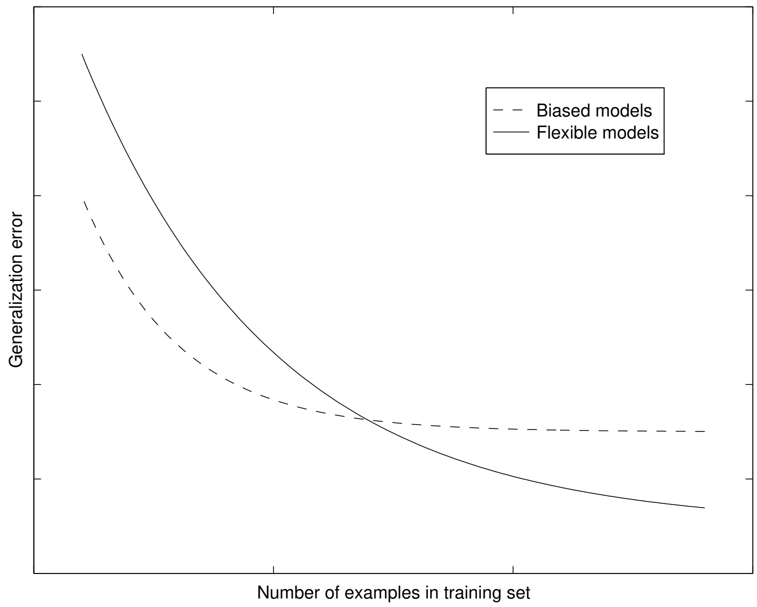

Fig. 2. Model generalization as a function of number of observations, N_{train}, used to estimate model parameters. Generalization error decreases with increasing N_{train} for both highly flexible and more biased models. The decrease is more rapid for the latter, whereas the former reaches a lower level for large values of N_{train}. At the point of generalization crossover enough data is available to support the use of more flexible, low-biased models.

task with their left hand [22]. For each subject four scans were acquired in each of the two states yielding a total of 240 scans.

Before scanning [O^{15}]water was automatically injected in the subjects right arm leaving the left arm free to perform the task. With the eyes covered by a patch an auditory timing signal was delivered by insert earphones.

For baseline (rest) scans, subjects were instructed to lie still and remain awake; they received no stimulation. For motor activation scans, the subjects left arm was positioned perpendicular to the scanning couch. At the start of the injection, the timing signal was initiated and the finger-thumb opposition task continued for 60 s. The finger-thumb opposition task consisted of sequential opposition of the thumb and successive digits, and back again $(2, 3, 4, 5, 4, 3, 2, 3, 4, \ldots)$ at a rate of 1 Hz.

PET scanning commenced when the radioactive material reached the brain, typically 10–20 s after injection, and data acquisition continued for 90 s. Each scanning session consisted of eight 90 s PET scans separated by 10 min rest periods to allow for O^{15} decay, for a total experimental time of approximately 90 min. The first, third, fifth, and seventh scans were acquired in the baseline state, and the second, fourth, sixth, and eighth scans were acquired in the activated state. Scans corrected for randoms, dead-time, and attenuation, but not for scatter, were reconstructed using 3D filtered back-projection.

4.2 fMRI Scanning

A single subject performing a left-handed finger-to-thumb opposition task was scanned during eight 180 s runs. In each run 24 baseline, 24 activation, and 24 baseline whole brain echo planar scans were acquired (2.5s/scan) with an interslice distance of 8 mm and an in plane voxel resolution of 3.1×3.1 mm^2. This yielded a total of 576 scans. During activation the task was timed with an auditory signal at a rate of 1 Hz.

4.3 Scan Alignment and Preprocessing

The PET and fMRI scans were intra-subject aligned using AIR (Automated Image Registration) [27] and only the PET scans were then stereo-tactically normalized to a simulated PET reference volume in Talairach space [25] using the 12 parameter linear transformation described in [28] (see [22] for more details). This yielded scans with 48 slices, inter-slice distance of 3.4 mm and in plane voxel resolution of 3.1×3.1 mm^2. After masking out voxels outside the brain an SVD basis was computed based on the entire[3] set of scans.

4.4 Modeling

After normalizing the singular vectors, $\mathbf{v}$, to zero mean and a standard deviation of one, a fixed test set was randomly selected (100 for the PET data and 200 for the fMRI data). The remaining observations were utilized to yield training sets of increasing size. A number of training sets of each size (25 for the PET data and 20 for the fMRI data) were randomly sampled with replacement[4] from the singular vectors. For each of the resulting training sets a linear (FLD) and a nonlinear (ANN) classifier were estimated. Model performance was then assessed using the fixed test set. The linear and nonlinear classifiers are based on different cost functions, so to allow a quantitative comparison generalization was measured as the mean misclassification on the independent test set.

5 Results

Figure 3 depicts the learning curves for the linear and nonlinear classifiers on the PET data. The two curves are slightly offset horizontally to better show the

[3] Basing models on an SVD of the entire set of observation limits results from generalization measures to the specific set of subjects in the PET case, and the specific subject in the fMRI case. Thus, generalization error does not implicate the extent to which models generalize to subjects other than those included in the datasets.

[4] Estimators based on sampling with replacement (also known as bootstrapping), where the same observation may appear more than once in the same sample, are asymptotically central [5]—however counter-intuitive this may seem.

error-bars that indicate one standard deviation of the mean for each training set size. As hypothesized both learning curves decrease. The nonlinear classifier

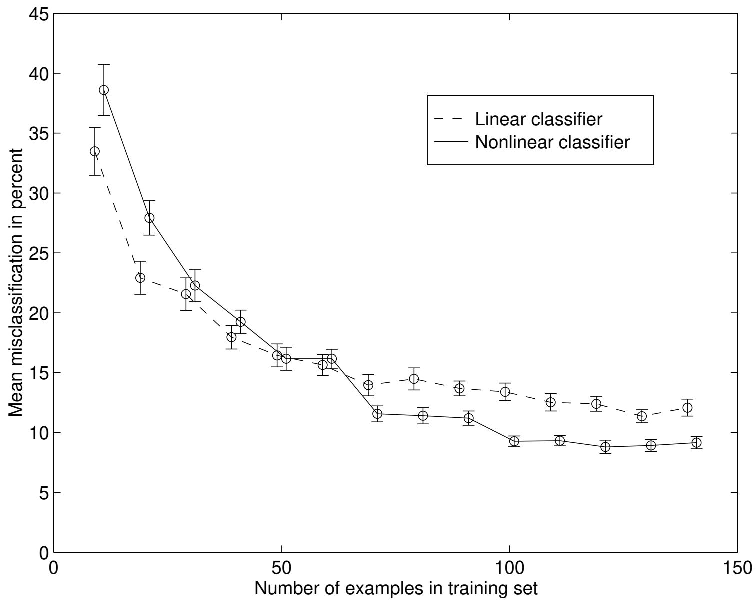

Fig. 3. For an [O^{15}]water PET study of a simple finger opposition task model generalization (measured as the mean misclassification on an independent test set) is plotted as a function of number of observations, N_{train}, used to estimate model parameters. Generalization error decreases with increasing N_{train} for the linear as well as the nonlinear classifiers. However, generalization error decreases more rapidly and settles at a higher level for the linear classifier than for its nonlinear counterpart. Thus, for this task linear classifiers seem optimal for small datasets. As more observations become available we are better off using the more flexible nonlinear classifiers.

seems to generalize worse for small training sets but perform relatively better as N_{train} increases. Indeed, a generalization crossover occurs for training sets with around 60 examples, and as N_{train} increases further generalization error for the nonlinear classifier settles at a lower level than that of its linear counterpart.

For the fMRI dataset Fig. 4 shows a similar picture. Again the learning curves for the linear and nonlinear classifiers cross as the number of observations in the training set is increased. Thus, for small training sets we can not reject the linear model.

6 Discussion

We have introduced a general framework for the analysis of functional activation datasets. In this framework the extremely ill-posed nature of typical datasets

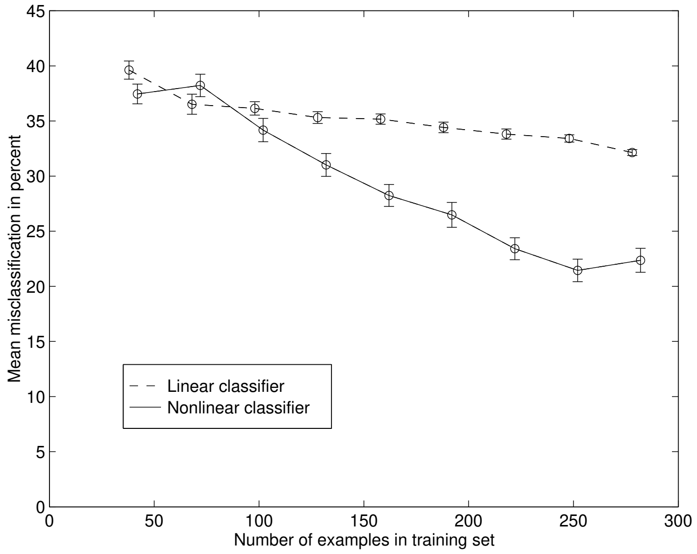

Fig. 4. For an fMRI study of a left-handed finger-to-thumb opposition task model generalization (measured as the mean misclassification on an independent test set) is plotted as a function of number of observations, N_{train}, used to estimate model parameters. Generalization error decreases with increasing N_{train} for the linear as well as the nonlinear classifiers. However, generalization error decreases more rapidly and settles at a higher level for the linear classifier than for its nonlinear counterpart. Again, the linear classifiers can not be rejected for small datasets. As more observations become available we are better off using the more flexible nonlinear classifiers.

imposes an immense computational burden on any modeling procedures. We have shown how a simple coordinate transform reduces data representation without loss of information, thus minimizing the computational load.

The importance of not measuring model performance on the same set of data used to estimate the model parameters has been stressed, and we have sketched how independent test sets provide empirical estimates of generalization. We have hypothesized how generalization error decreases as more observations become available for parameter estimation. Decreasing learning curves satisfying our hypothesis have been demonstrated on two functional activation datasets of PET and fMRI scans of subjects performing simple motor tasks.

By employing model families that differ in flexibility we have further shown the effect of model flexibility on the slope of the learning curves. For the studied tasks we have identified generalization crossovers, at which point enough observations are available to support the use of a more flexible, nonlinear model. We believe this to have implications for the future of modeling in functional neuroimaging; as more and more data become available the support for more sophisticated and flexible models increase. While introducing problems of their own (by e.g. not being linear in their parameters), these models can potentially lead to increased knowledge of the systems that govern information processing in the living, human brain.

7 Acknowledgments

This work has been funded in part by the Human Brain Project grant P20 MH57180, the Danish Research Councils for the Natural and Technical Sciences through the Danish Computational Neural Network Center, CONNECT, the Danish Research Council for Medical Science, and the Danish Research Academy.

References

1. H. Akaike. Fitting autoregressive models for prediction. *Annals of the Institute of Statistical Mathematics*, 21:243–247, 1969.
2. C. M. Bishop. *Neural Networks for Pattern Recognition.* Clarendon Press, Oxford, 1995.
3. J. S. Bridle. Training stochastic model recognition algorithms as networks can lead to maximum mutual information estimation of parameters. *Advances in Neural Information Processing Systems*, 2:211–217, 1990.
4. R. O. Duda and P. E. Hart. *Pattern Classification and Scene Analysis.* John Wiley & Sons, 1973.
5. B. Efron and R. J. Tibshirani. *An Introduction to the Bootstrap.* Monographs on Statistics and Applied Probability. Chapman & Hall, 1993.
6. J. H. Friedman. On bias, variance, 0/1-loss, and the curse-of-dimensionality. *Journal of Knowledge Discovery and Data Mining*, 1996. In press.
7. K. J. Friston, J.-P. Poline, A. P. Holmes, C. D. Frith, and R. S. J. Frackowiak. A multivariate analysis of PET activation studies. *Human Brain Mapping*, 4:140–151, 1996.
8. B. Hassibi and D. G. Stork. Optimal brain surgeon. *Advances in Neural Information Processing Systems*, 5:164–174, 1992.
9. J. Hertz, A. Krogh, and R. G. Palmer. *Introduction to the Theory of Neural Computation.* Addison-Wesley, 1994.
10. M. Hintz-Madsen, M. W. Pederson, L. K. Hansen, and J. Larsen. Design and evaluation of neural skin classifiers. In Y. Tohkura, S. Katagiri, and E. Wilson, editors, *Proceedings of 1996 IEEE Workshop on Neural Networks for Signal Processing*, pages 223–230, 1996.
11. J. E. Jackson. *A User's Guide to Principal Components.* Wiley Series on Probability and Statistics, John Wiley and Sons, 1991.
12. B. Lautrup, L. K. Hansen, I. Law, N. Mørch, C. Svarer, and S. C. Strother. Massive weight sharing: A cure for extremely ill-posed problems. In H. J. Hermann, D. E. Wolf, and E. P. Pöppel, editors, *Proceedings of Workshop on Supercomputing in Brain Research: From Tomography to Neural Networks, HLRZ, KFA Jülich, Germany*, pages 137–148, 1994.
13. Le Cun, Y., J. S. Denker, and S. Solla. Optimal brain damage. *Advances in Neural Information Processing Systems*, 2:598–605, 1990.
14. K. V. Mardia, J. T. Kent, and J. M. Bibby. *Multivariate Analysis.* Academic Press, 1979.

15. J. R. Moeller and S. C. Strother. A regional covariance approach to the analysis of functional patterns in positron emission tomographic data. *Journal of Cerebral Blood Flow and Metabolism*, 11:A121–A135, 1991.
16. J. R. Moeller, S. C. Strother, J. J. Sidtis, and D. A. Rottenberg. Scaled subprofile model: A statistical approach to the analysis of functional patterns in positron emission tomographic data. *Journal of Cerebral Blood Flow and Metabolism*, 7:649–658, 1987.
17. J. Moody. Prediction risk and architecture selection for neural networks. In V. Cherkassky, J. H. F. H., and H. Wechsler, editors, *From Statistics to Neural Networks, Theory and Pattern Recognition Applications*, pages 147–165. Springer Verlag, 1992.
18. N. Mørch, L. K. Hansen, I. Law, S. C. Strother, C. Svarer, B. Lautrup, U. Kjems, N. Lange, and O. B. Paulson. Generalization and the bias-variance trade-off in models of functional activation. *IEEE Transactions on Medical Imaging*, 1996. Submitted.
19. N. Mørch, U. Kjems, L. K. Hansen, C. Svarer, I. Law, B. Lautrup, S. Strother, and K. Rehm. Visualization of neural networks using saliency maps. In *Proceedings of 1995 IEEE International Conference on Neural Networks*, volume 4, pages 2085–2090, 1995.
20. N. Murata, S. Yoshizawa, and S.-I. Amari. Network information criterion—determining the number of hidden units for an artificial neural network model. *IEEE Transactions on Neural Networks*, 5:865–872, 1994.
21. M. I. Posner and M. E. Raichle. *Images of Mind.* W. H. Freeman, 1994.
22. S. C. Strother, J. R. Anderson, K. A. Schaper, J. J. Sidtis, J. S. Liow, R. P. Woods, and D. A. Rottenberg. Principal component analysis and the scaled subprofile model compared to intersubject averaging and statistical parametric mapping: I. "Functional connectivity" of the human motor system studied with [^{15}O]water PET. *Journal of Cerebral Blood Flow and Metabolism*, 15:738–753, 1995.
23. S. C. Strother, J. R. Anderson, K. A. Schaper, J. J. Sidtis, and D. A. Rottenberg. Linear models of orthogonal subspaces & networks from functional activation PET studies of the human brain. In Y. Bizais, C. Barillot, and R. D. Paola, editors, *Proceedings of the 14th International Conference on Information Processing in Medical Imaging*, pages 299–310. Kluwer Academic Publishers, 1995.
24. C. Svarer, L. K. Hansen, and J. Larsen. On design and evaluation of tapped-delay neural network architectures. In H. R. Berenji et al., editors, *Proceedings of 1993 IEEE International Conference on Neural Networks*, pages 45–51, 1993.
25. J. Talairach and P. Tournoux. *Co-planar stereotaxic atlas of the human brain.* Thieme Medical Publishers Inc., New York, 1988.
26. A. W. Toga and J. C. Mazziotta. *Brain Mapping.* Academic Press, 1996.
27. R. P. Woods, S. R. Cherry, and J. C. Mazziotta. A rapid automated algorithm for accurately aligning and reslicing positron emission tomography images. *Journal of Computer Assisted Tomography*, 16:620–633, 1992.
28. R. P. Woods, J. C. Mazziotta, and S. R. Cherry. Automated image registration. In K. Uemura et al., editors, *Quantification of Brain Function. Tracer Kinetics and Image Analysis in Brain PET*, pages 391–400. Elsevier Science Publishers B. V., 1993.

Bibliography

Akaike, H. (1969). Fitting autoregressive models for prediction. *Annals of the Institute of Statistical Mathematics*, 21:243–247.

Amari, S., Cichocki, A., and Yang, H. H. (1996). A new learning algorithm for blind signal separation. *Proceedings of Advances in Neural Information Processing Systems*, 8:757–763.

Bandettini, P. A., Wong, E. C., Hinks, R. S., Tikofsky, R. S., and Hyde, J. S. (1992). Time course EPI of human brain function during task activation. *Magnetic Resonance in Medicine*, 25:390–398.

Bell, A. J. and Sejnowski, T. J. (1995a). Fast blind separation based on information theory. In *Proceedings 1995 International Symposium on Non-linear Theory and Applications*, volume 1, pages 43–47.

Bell, A. J. and Sejnowski, T. J. (1995b). An information-maximization approach to blind separation and blind deconvolution. *Neural Computation*, 7:1129—1159.

Bell, A. J. and Sejnowski, T. J. (1996). The 'independent components' of natural scenes are edge filters. *Vision Research.* To appear.

Bell, D. J. (1990). *Mathematics of linear and nonlinear systems.* Clarendon Press, Oxford.

Bellman, R. (1961). *Adaptive Control Processes: A Guided Tour.* Princeton University Press.

Bishop, C. M. (1995). *Neural Networks for Pattern Recognition.* Clarendon Press, Oxford.

Bridle, J. S. (1990). Training stochastic model recognition algorithms as networks can lead to maximum mutual information estimation of parameters. *Advances in Neural Information Processing Systems*, 2:211–217.

Buntine, W. L. and Weigend, A. S. (1994). Computing second derivatives in feed-forward networks. *IEEE transactions on Neural Networks*, 5(3):480–488.

Cardoso, J.-F. (1997). Infomax and maximum likelihood for blind source separation. *IEEE Signal processing letters.*

Cardoso, J.-F. and Laheld, B. H. (1996). Equivariant adaptive source separation. *IEEE Transactions on Signal Processing.*

Comon, P. (1994). Independent component analysis, a new concept? *Signal Processing*, 36:287–314.

Cybenko, G. (1990). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals, and Systems.*

Duda, R. O. and Hart, P. E. (1973). *Pattern Classification and Scene Analysis.* John Wiley & Sons.

Efron, B. (1982). *The Jackknife, the Bootstrap and Other Resampling Plans.* Society for industrial and applied mathematics. Capital City Press.

Efron, B. (1983). Estimating the error rate of a prediction rule: Improvement on cross-validation. *Journal of the American Statistical Association*, 78(382):316–331.

Efron, B. and Tibshirani, R. J. (1993). *An Introduction to the Bootstrap.* Monographs on Statistics and Applied Probability. Chapman & Hall.

Fahlman, S. E. and Lebiere, C. (1990). The cascade-correlation learning architecture. *Advances in Neural Information Processing Systems*, 2:524–532.

Fischer, B. and Breitmeyer, B. (1987). Mechanisms of visual attention revealed by saccadic eye movements. *Neuropsychologia*, 25:73–83.

Fox, P. T. and Mintun, M. A. (1989). Noninvasive function brain mapping by change-distribution analysis of averaged PET images of $H_2{}^{15}O$ tissue activity. *Journal of Nuclear Medicine*, 30:141–149.

Frackowiak, R., Friston, K. J., Frith, C. D., Dolan, R. J., and Mazziotta, J. C. (1997). *Human Brain Function.* Academic Press.

Friston, K. J. (1994). Statistical parametric mapping. In Thatcher, R. W., Hallet, M., Zeffiro, T., John, E. R., and Huerta, M., editors, *Functional Neuroimaging: Technical Foundations*, chapter 8, pages 79–91. Academic Press.

Friston, K. J., Frith, C. D., Liddle, P. F., Dolan, R. J., Lammertsma, A. A., and Frackowiak, R. S. J. (1990). The relationship between global and local changes in PET scans. *Journal of Cerebral Blood Flow and Metabolism*, 10:458–466.

Friston, K. J., Holmes, A. P., Worsley, K. J., Poline, J.-P., Frith, C. D., and Frackowiak, R. S. J. (1995). Statistical parametric maps in functional imaging: A general linear approach. *Human Brain Mapping*, 2(4):189–210.

Friston, K. J., Poline, J.-P., Holmes, A. P., Frith, C. D., and Frackowiak, R. S. J. (1996). A multivariate analysis of PET activation studies. *Human Brain Mapping*, 4:140–151.

Geman, S., Bienenstock, E., and Doursat, R. (1992). Neural networks and the bias/variance dilemma. *Neural Computation*, 4:1–58.

Hansen, L. K. and Larsen, J. (1996). Linear unlearning for cross-validation. *Advances in Computational Mathematics*, 5:269–280.

Hansen, L. K. and Larsen, J. (1998). Source separation in short image sequences using delayed correlation. In *Proceedings of NORSIG'98, Vigsø, Denmark.* To appear.

Hansen, L. K. and Pedersen, M. W. (1994). Controlled growth of cascade correlation nets. In Marinaro, M. and Morasso, P. G., editors, *International Conference on Artificial Neural Networks ICANN'94*, pages 797–801. Springer.

Hansen, P. S., Bendsøe, M. P., and Nielsen, H. B. (1987). *Lineær Algebra - Datamatorienteret*. Department of Mathematics, Technical University of Denmark. In Danish.

Hassibi, B. and Stork, D. G. (1992). Optimal brain surgeon. *Advances in Neural Information Processing Systems*, 5:164–174.

Hertz, J., Krogh, A., and Palmer, R. G. (1994). *Introduction to the Theory of Neural Computation*. Addison-Wesley.

Hinton, G. E. (1986). Learning distributed representations of concepts. In *Proceedings of the Eighth Annual Conference on the Cognitive Science Society*, pages 1–12.

Hintz-Madsen, M. et al. (1995). Design and evaluation of neural classifiers - application to skin lesion classification. In Girosi, F., Makhoul, J., Manolakos, E., and Wilson, E., editors, *Proceedings of 1995 IEEE Workshop on Neural Networks for Signal Processing*, pages 484–493.

Hoerl, A. E. and Kennard, R. W. (1970). Ridge regression. *Technometrics*, 12:55–67, 69–82.

Hornik, K., Stinchcombe, M., and White, H. (1990). Universal approximation of an unknown mapping and its derivatives using multilayer feedforward networks. *Neural Networks*, 3.

Jackson, J. E. (1991). *A User's Guide to Principal Components*. Wiley Series on Probability and Statistics, John Wiley and Sons.

Jutten, C. and Herault, J. (1991). Blind separation of sources, part I: An adaptive algorithm based on neuromimetic architecture. *Signal Processing*, 24:1–10.

Kendall, M. G. and Stuart, A. (1967). *The Advanced Theory of Statistics*. Charles Griffin & Co.

Kim, S.-G. and Ugurbil, K. (1997). Functional magnetic resonance imaging of the human brain. *Journal of Neuroscience Methods*, 74(2):229–243.

Kjems, U. (1998). *Bayesian Signal Processing and Interpretation of Brain Scans*. PhD thesis, Technical University of Denmark.

Kjems, U., Philipsen, P. A., Hansen, L. K., Chen, C.-T., and Anderson, J. (1996). A non-linear 3D MRI brain co-registration method. In *Proceedings of the Interdisciplinary Inversion Workshop '96*.

Kjems, U., Strother, S. C., Anderson, J., Law, I., and Hansen, L. K. (1997). A new 3D non-linear brain MRI registration algorithm improving functional [^{15}O]water PET registration. *IEEE Transactions on Medical Imaging*. Submitted.

Kramer, C. L. and Buonanno, F. S. (1985). Physical principles of nuclear magnetic resonance and its application to imaging. In Ganzales, C. F., Grossman, C. B., and Masdeu, J. C., editors, *Head and Spine Imaging*, Wiley Medical Publication, chapter 26, pages 859–887. John Wiley & Sons, New York, 1. edition.

Larsen, J. (1994). *Design of Neural Network Filters*. PhD thesis, Electronics Institute, Technical University of Denmark.

Larsen, J. and Hansen, L. K. (1995b). Empirical generalization assessment of neural netwoek models. In Girosi, F., Makhoul, J., Manolakos, E., and Wilson, E., editors, *Proceedings of the IEEE Workshop on Neural Networks for Signal Processing V*, number 5, pages 30–39.

Larsen, J. and Hansen, L. K. (1995a). Empirical generalization assessment of neural network models. In *Proceedings of 1995 IEEE Workshop on Neural Networks for Signal Processing*, pages 30–39.

Larsen, J., Hansen, L. K., Svarer, C., and Ohlsson, M. (1996). Design and regularization of neural networks: The optimal use of a validation set. In Usui, S., Tohkura, Y., Katagiri, S., and Wilson, E., editors, *Proceedings of the IEEE Workshop on Neural Networks for Signal Processing VI*, number 6, pages 62–71.

Lautrup, B., Hansen, L. K., Law, I., Mørch, N., Svarer, C., and Strother, S. C. (1994). Massive weight sharing: A cure for extremely ill-posed problems. In Hermann, H. J., Wolf, D. E., and Pöppel, E. P., editors, *Proceedings of Workshop on Supercomputing in Brain Research: From Tomography to Neural Networks, HLRZ, KFA Jülich, Germany*, pages 137–148.

Law, I. (1997). *Saccadic eye movements: A functional brain mapping approach.* PhD thesis, Neurobiology Research Unit, Rigshospitalet, Copenhagen University Hospital.

Law, I., Svarer, C., and Paulson, O. B. (1995). Characterization of cortical responses during the performance of reflexive and antisaccadic eye movements. *Human Brain Mapping*, Supplement 1:323.

Le Cun, Y., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., and Jackel, L. D. (1989). Backpropagation applied to handwritten ZIP code recognition. *Neural Computation*, 1:541–551.

Le Cun, Y., Denker, J. S., and Solla, S. (1990). Optimal brain damage. *Advances in Neural Information Processing Systems*, 2:598–605.

Ljung, L. (1987). *System Identification: Theory for the User.* Information and System Sciences Series. Prentice-Hall.

Lundsager, B. and Kristensen, B. L. (1996). Lineær og ulineær modellering af positron emissions tomografier. Master's thesis, Technical University of Denmark. In Danish.

MacKay, D. J. C. (1996). Maximum likelihood and covariant algorithms for independent component analysis. In preparation.

Malonek, D. and Grinvald, A. (1996). Interactions between electrical activity and cortical microcirculation revealed by imaging spectroscopy—Implications for functional brain mapping. *Science*, 272:551–554.

Mardia, K. V., Kent, J. T., and Bibby, J. M. (1979). *Multivariate Analysis.* Academic Press.

Marquardt, D. W. (1963). An algorithm for least-squares estimation of non-linear parameters. *Journal of the Society of Industrial and Applied Mathematics*, 11(2):431–441.

McKeown, M. J., Jung, T.-P., Makeig, S., Brown, G., Kindermann, S., Lee, T.-W., and Sejnowski, T. J. (1998). Spatially independent activity patterns in functional MRI data during the stroop color-naming task. *Proccedings of the National Academy of Sciences*, 95:803–810.

Moeller, J. R. and Strother, S. C. (1991). A regional covariance approach to the analysis of functional patterns in positron emission tomographic data. *Journal of Cerebral Blood Flow and Metabolism*, 11:A121–A135.

Moeller, J. R., Strother, S. C., Sidtis, J. J., and Rottenberg, D. A. (1987). Scaled subprofile model: A statistical approach to the analysis of functional patterns in positron emission tomographic data. *Journal of Cerebral Blood Flow and Metabolism*, 7:649–658.

Molgedey, L. and Schuster, H. (1997). Separation of independent signals using time-delayed correlations. *Physical Review Letters*, 72(23):3634–3637.

Moody, J. (1992). The effective number of parameters: An analysis of generalization and regularization in nonlinear learning systems. *Advances in Neural Information Processing Systems*, 4:847–854.

Moody, J. and Darken, C. (1989). Fast learning in networks of locally-tuned processing units. *Neural Computation*, 1:281–294.

Mørch, N., Hansen, L. K., Law, I., Strother, S. C., Svarer, C., Lautrup, B., Kjems, U., Lange, N., and Paulson, O. B. (1996a). Generalization and the bias-variance trade-off in models of functional activation. Preprint; Department of Mathematical Modelling, Technical University of Denmark.

Mørch, N., Hansen, L. K., Strother, S. C., Law, I., Svarer, C., Lautrup, B., Anderson, J. R., Lange, N., and Paulson, O. B. (1996b). Generalization performance of nonlinear vs. linear models for [^{15}O]water PET functional activation studies. *NeuroImage*, 3(3):258.

Mørch, N., Hansen, L. K., Strother, S. C., Svarer, C., Rottenberg, D. A., Lautrup, B., Savoy, R., and Paulson, O. B. (1997). Nonlinear versus linear models in functional neuroimaging: Learning curves and generalization crossover. In *Proceedings of the 15th International Conference on Information Processing in Medical Imaging*, pages 259–270.

Mørch, N., Kjems, U., Hansen, L. K., Svarer, C., Law, I., Lautrup, B., Strother, S., and Rehm, K. (1995). Visualization of neural networks using saliency maps. In *Proceedings of 1995 IEEE International Conference on Neural Networks*, volume 4, pages 2085–2090.

Mørch, N. and Thomsen, J. (1994). Statistisk analyse af positron emissions tomografier. Master's thesis, Technical University of Denmark. In Danish.

Murata, N., Yoshizawa, S., and Amari, S.-I. (1994). Network information criterion—determining the number of hidden units for an artificial neural network model. *IEEE Transactions on Neural Networks*, 5:865–872.

Nolte, J. (1993). *The Human Brain*. Mosby-Year Book.

Ogawa, S., Lee, T. M., Nayak, A. S., and Glynn, P. (1990). Oxygenation-sensitive contrast in magnetic resonance image of rodent brain at high magnetic fields. *Magnetic Resonance Imaging*, 14:68–78.

Oldfield, R. C. (1971). The assessment and analysis of handedness: The Edingburgh inventory. *Neuropsychologia*, 9:97–113.

Papoulis, A. (1991). *Probability, Random Variables, and Stochastic Processes.* McGraw-Hill Series in Electrical Engineering. McGraw-Hill, Inc., third edition.

Pedersen, M. W. (1997). *Optimization of Recurrent Neural Networks for Time Series Modeling.* PhD thesis, Technical University of Denmark.

Pedersen, M. W., Hansen, L. K., and Larsen, J. (1995). Pruning with generalization based saliencies: γOBD, γOBS. *Proceedings of Advances in Neural Information Processing Systems.*

Phelps, M. E. (1986). *Positron Emission Tomography and Autoradiography: Principles and Applications for the Brain and Heart.* Raven Press, New York.

Radon, J. (1917). Über die bestimmung von funktionen durch ihre integralwerte längs gewisser mannigfaltigkeiten. *Ber. Ver. Sächs. Akad. Wiss. Leipzig, Math-Phys. Kl.*, 69:262–277. In German.

Ripley, B. D. (1996). *Pattern Recognition and Neural Networks.* Cambridge University Press.

Rosenblatt, F. (1962). *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanics.* Spartan.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning internal representations by error propagation. In Rumelhart, D. E., McClelland, J. L., and the PDP Research Group, editors, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, volume 1: Foundations, pages 318–362. MIT Press.

Scharf, L. L. (1991). *Statistical Signal Processing: Detection, Estimation, and Time Series Analysis.* Addison-Wesley.

Shannon, C. E. (1948). A mathematical theory of communication. *The Bell System Technical Journal*, 27(3):379–423, 623–656.

Stone, M. (1974). Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society*, 36(1):111–147.

Strother, S. C., Anderson, J. R., Schaper, K. A., Sidtis, J. J., Liow, J. S., Woods, R. P., and Rottenberg, D. A. (1995a). Principal component analysis and the scaled subprofile model compared to intersubject averaging and statistical parametric mapping: I. "Functional connectivity" of the human motor system studied with [^{15}O]water PET. *Journal of Cerebral Blood Flow and Metabolism*, 15:738–753.

Strother, S. C., Anderson, J. R., Schaper, K. A., Sidtis, J. J., and Rottenberg, D. A. (1995b). Linear models of orthogonal subspaces & networks from functional activation PET studies of the human brain. In Bizais, Y., Barillot, C., and Paola, R. D., editors, *Proceedings of the 14th International Conference on Information Processing in Medical Imaging*, pages 299–310. Kluwer Academic Publishers.

Svarer, C., Hansen, L. K., and Larsen, J. (1993). On design and evaluation of tapped-delay neural network architectures. In Berenji, H. R. et al., editors, *Proceedings of 1993 IEEE International Conference on Neural Networks*, pages 45–51.

Talairach, J. and Tournoux, P. (1988). *Co-planar stereotaxic atlas of the human brain.* Thieme Medical Publishers Inc., New York.

Toft, P. (1996). *The Radon Transform.* PhD thesis, Technical University of Denmark.

Toussaint, G. T. (1974). Bibliography on estimation of misclassification. *IEEE Transactions on Information Theory*, 20(4):472–479.

Werbos, P. J. (1974). *Beyond regression: New tools for prediction and analysis in the behavioural sciences.* PhD thesis, Harvard University.

Woods, R. P., Cherry, S. R., and Mazziotta, J. C. (1992). A rapid automated algorithm for accurately aligning and reslicing positron emission tomography images. *Journal of Computer Assisted Tomography*, 16:620–633.

Woods, R. P., Mazziotta, J. C., and Cherry, S. R. (1993). Automated image registration. In Uemura, K. et al., editors, *Quantification of Brain Function. Tracer Kinetics and Image Analysis in Brain PET*, pages 391–400. Elsevier Science Publishers B. V.

Worsley, K. J., Poline, J. B., Friston, K. J., and Evans, A. C. (1998). Characterizing the response of PET and fMRI data using multivariate linear models (MLM). *NeuroImage*, 6:305–319.

Young, G. A. (1994). Bootstrap: More than a stab in the dark? *Statistical Science*, 9(3):382–415.