# Automatic Differentiation Algorithms in Model Analysis

| | |
|---|---|
| **Promotoren:** | prof.dr.ir. J. Grasman<br>hoogleraar in de wiskundige en statistische methoden |
| | prof.dr.ir. A. Stein<br>hoogleraar in de statistiek,<br>met speciale aandacht voor de ruimtelijke statistiek |
| **Co-promotor:** | dr. M. de Gee<br>universitair hoofddocent,<br>leerstoelgroep wiskundige en statistische methoden |
| **Promotiecommissie:** | dr. J.J.M. Bedaux, Vrije Universiteit Amsterdam<br>dr. M.J.W. Jansen, Plant Research International, Wageningen<br>prof.dr.ir. G. van Straten, Wageningen Universiteit<br>prof.dr. L. Hordijk, Wageningen Universiteit |

Mark Jeroen Huiskes

# Automatic Differentiation Algorithms in Model Analysis

Proefschrift
ter verkrijging van de graad van doctor
op gezag van de rector magnificus
van Wageningen Universiteit,
prof. dr. ir. L. Speelman,
in het openbaar te verdedigen
op dinsdag 19 maart 2002
des namiddags te vier uur in de Aula.

ISBN 90-5808-601-1

# Voorwoord

Dit voorwoord schrijf ik aan het eind van de periode van ongeveer vijf jaar die ik aan dit promotie-onderzoek heb gewerkt bij de leerstoelgroep Mathematische en Statistische Methoden van de Wageningen Universiteit. Ik wil hier van de gelegenheid gebruik maken om een aantal mensen in het bijzonder te bedanken voor hun directe of indirecte bijdrage aan het tot stand komen van dit proefschrift.
Allereerst dank ik mijn promotor Johan Grasman voor de plezierige samenwerking. Je deur stond altijd voor me open, en ik ben je dankbaar dat je me de gelegenheid en de tijd gegeven hebt die ik nodig had (of nodig dacht te hebben) om dit proefschrift met een goed gevoel af te ronden. Ook co-promotor Maarten de Gee wil ik bedanken voor zijn bijdrage hieraan. De bijeenkomsten met jullie beiden hebben tot veel nuttige discussies en ideeën geleid; daarbij hebben jullie me in het onderzoek de vrijheid gegeven mijn eigen interesse te volgen – dit is een belangrijke factor geweest in het plezier waarmee ik de afgelopen jaren heb gewerkt!
Alfred Stein dank ik dat hij bereid is geweest om nog in een laat stadium van het onderzoek promotor te worden. Het was prettig om het statistische deel van het proefschrift met je door te kunnen spreken, en ik dank je voor de nuttige suggesties.

Alle overige collega's van het wiskundegebouw wil ik bedanken voor de gezellige, interessante en leerzame tijd die ik met hen heb doorgebracht. Onno van Herwaarden en Jo Gielen dank ik ook voor hun begeleiding bij mijn eerste stappen op het onderwijspad; Aad van Eijnsbergen dank ik voor zijn hulp bij het afleiden van een uitdrukking in hoofdstuk 3; en Gerrit Gort, Fred van Eeuwijk, Theo Hendriks en Cees Wiersum dank ik voor het afzien op de squashbaan. Cees, ik kon het ook erg waarderen dat je aan het begin van mijn nieuwe leven in Wageningen altijd in was om leuke dingen te gaan doen; daarbij ben ik dankbaar dat ik je muzikale bochtenwerk nog kan navertellen. Van de spelletjesavonden (georganiseerd door Eligius Hendrix en Lia Hemerik) zou ik graag binnenkort een keer een herstart beleven.
Met mijn mede-promovendi (in volgorde van aankomst) Annemarie Pielaat, Eric Boer, João Paulo en Anna Rzepiela heb ik het lief en leed van het AIO-bestaan kunnen delen. Graag zet ik de traditie voort om jullie persoonlijk te bedanken (Anna, as you are a special case, I will save my thanks to you until the very end.)

Annemarie, ik denk met veel plezier terug aan onze tijd als kamergenoten; vooral mis ik de dagelijkse slecht-maar-verder-gaat-het-goed type gesprekjes (of priet-praat, zoals jij zou zeggen). Daarbij heb ik de combinatie van creativiteit en discipline waarmee je je leven organiseert altijd bewonderd; ik dank je voor de vele leuke momenten en ondernemingen. Ik wens jou, Gerrit en Louise een goede tijd toe in Canada en ik hoop dat ik door middel van je emailschrijfsels nog vaak van je inzichten kan blijven genieten.
Eric, ook met jou heb ik de afgelopen jaren meer meegemaakt dan ik hier in een paar zinnetjes kwijt kan – denk alleen al maar aan al de tochtjes met jouw giant Suzuki. Bij de verdediging van het proefschrift hoop ik dat jouw rust en bedachtzaamheid me tot steun zullen zijn – ik dank je dat je mijn paranimf wilt zijn. Daarnaast rekenen ik en "the couch" ook op je bij de volgende verhuizing.
João, jou dank ik voor de interessante gesprekken over koken, boeken, films en de meest uiteenlopende andere onderwerpen. Naast je kamergenote, maakten ook jouw open vrolijkheid en enthousiasme het altijd erg verleidelijk om jullie kamer te bezoeken.

Ook buiten de universiteit hebben een aantal mensen een belangrijke rol gespeeld bij dit onderzoek. Met name Martin Pastoors van het Rijksinstituut voor Visserij Onderzoek wil ik bedanken voor de goede samenwerking; speciaal ook voor de gastvrijheid in IJmuiden en het mogelijk maken van een bezoek aan een haring-workshop in Kopenhagen. Willem Dekker dank ik voor het advies in de eerste fase van het onderzoek.
As part of the Young Scientist Summer Program, I spent 3 months at the International Institute for Applied Systems Analysis in Vienna. These turned out to be very inspiring surroundings, both in terms of the research and in terms of the company (and the mountaineous setting): I thank the organizers and the other participants for a beautiful summer! I am also grateful to Alexander Tarasyev, Arkadii Kriajimskii and Anatoli Shvidenko for their mentoring during my project there.
Eric Pauwels van het Centrum voor Wiskunde en Informatica dank ik voor zijn geduld en begrip bij het afronden van het proefschrift.

Mijn vader, moeder en Pascal mogen hier niet ontbreken: ik dank jullie voor de voortdurende steun en voor de wetenschap dat ik altijd op jullie kan rekenen. Mijn overige familie en vrienden ben ik dankbaar voor hun geduld en de aangename tijden naast het werk.

Finally, Anna. You have been the greatest fortune on this journey – thank you for your love and support. Kocham Cię kochanie moje!

Mark Huiskes
Wageningen, 11 februari 2002.

# Abstract

In this thesis automatic differentiation algorithms and derivative-based methods are combined to develop efficient tools for model analysis. Automatic differentiation algorithms comprise a class of algorithms aimed at the derivative computation of functions that are represented as computer code. Derivative-based methods that may be implemented using these algorithms are presented for sensitivity analysis and statistical inference, particularly in the context of nonlinear parameter estimation.

Local methods of sensitivity analysis are discussed for both explicit and implicit relations between variables. Particular attention is paid to propagation of uncertainty, and to the subsequent uncertainty decomposition of output uncertainty in the various sources of input uncertainty.

Statistical methods are presented for the computation of accurate inferential information for nonlinear parameter estimation problems by means of higher-order derivatives of the model functions. Methods are also discussed for the assessment of the appropriateness of model structure complexity in relation to quality of data.

To realize and demonstrate the potential of routines for model analysis based on automatic differentiation a software library is developed: a C++ library for the analysis of nonlinear models that can be represented by differentiable functions in which the methods for parameter estimation, statistical inference, model selection and sensitivity analysis are implemented.

Several experiments are performed to assess the performance of the library. The application of the derivative-based methods and the routines of the library is further demonstrated by means of a number of case studies in ecological assessment. In two studies, large parameter estimation procedures for fish stock assessment are analyzed: for the Pacific halibut and North Sea herring species. The derivative-based methods of sensitivity analysis are applied in a study on the contribution of Russian forests to the global carbon cycle.

# Contents

# Chapter 1

# Introduction

In many fields of science, large mathematical models are used for the analysis of complex systems. Examples are models for weather prediction in meteorology, macro-economic models in the field of econometrics and atmospheric transport models in environmental science.

Such models may come with a variety of purposes: to predict or reconstruct the dynamics of a process, to identify mechanisms that are important in determining the behavior of a process, to assist in making decisions, to analyze risk, or, to explore scientific ideas with respect to the perceived nature of a process.

Using models one attempts to describe certain aspects of reality in mathematical terms. Data can be used to assess the likelihood of the various alternative descriptions. Often, this takes the form of model fitting: a number of tuning parameters are left unspecified and observations of process variables are used to calibrate those parameters. The parameters that correspond to the alternative which fits the data best, are chosen to provide the model outcomes of choice. Accepting such model outcomes as 'the truth', however, without any further analysis of the uncertainty of the results would be highly undesirable. Rather, one wants to be able to assess the uncertainty of the parameters and other model outcomes, and also be able to quantify the sensitivity of the outcomes with respect to assumptions underlying the model.

Model fitting is, in statistical terms, a problem of *parameter estimation*. Using statistical models not only systematic effects can be described but also the nature and magnitude of the unexplained variation and uncertainty are taken into account. Methods of *statistical inference* quantify relative support for alternative models and different parameter values. By *sensitivity analysis* uncertainty with respect to assumptions is related to uncertainty in outcomes of the model. Moreover, uncertainty in the outcomes may be decomposed into various sources of uncertainty in the model assumptions.

The feasibility of these types of analysis depends on the nature of the model. For reasons that will become clear below, we restrict ourselves in this thesis to models where systematic effects are modelled by differentiable functions.

For such models, theory for statistical inference and sensitivity analysis is usually available, even though nonlinearity of the functions often takes particular care. In the practice of modeling complex systems, however, functions may typically consist of thousands of lines of computer code and contain hundreds of parameters. For such large models the required computations may turn out to be prohibitively expensive. This problem leads to an important aim of this thesis, namely of making statistical inference and sensitivity analysis more feasible in practice. Our approach to this effect is to exploit a combination of automatic differentiation algorithms and derivative-based methods for model analysis.

This first demands an explanation of the role of derivatives in model analysis. This role can hardly be overstated in general modeling; think for instance of the use of derivatives in the characterization of optima and in differential equations for modeling of change. In the analysis of models, the importance of first and higher-order derivatives can be explained by their role in providing simple local approximations to both explicitly and implicitly defined functions. Such approximations are used in function optimization, in bifurcation analysis and also, as will be discussed in sections 1.2 and 1.3, in sensitivity analysis and statistical inference. In these fields derivative-based methods can often provide an alternative for more expensive global methods of analysis.

Several methods exist for the computation of derivatives. Well-known approaches are numerical differentiation, for example by means of finite differences, and symbolic differentiation, i.e. the computer aided differentiation of mathematical expressions by computer algebra systems. Automatic differentiation algorithms make up a new class of algorithms for derivative computation, aimed at functions that are represented as computer code. The characteristics and benefits of these algorithms are discussed in section 1.1.

An important element of this study is the development of a software library to realize and demonstrate the potential of routines for model analysis based on automatic differentiation. This has resulted in a C++ library, named MAP, for the analysis of nonlinear models that can be represented by differentiable functions and in which the methods for sensitivity analysis, statistical inference and model selection discussed in this thesis are implemented. The library is available under a public license; more information can be obtained at `www.cwi.nl/~markh`.

## 1.1 Automatic differentiation (AD)

Implementations of AD algorithms take code of a function as input and provide code of its derivatives as output. Various types of implementations are available to this end; an example is the operator-overloading type of AD, for which some simple notation in the function code is sufficient to allow a regular compiler to generate the derivative code. The algorithms are able to deal with various difficulties such as program branching, iterative loops and joint allocations.

Moreover, the generated derivative code is code for the *exact* derivatives. Unlike for numerical differentiation methods, no approximation is necessary. This

also means that no stepsize has to be determined in order to trade off roundoff error and approximation error.

Similar to methods of symbolic differentiation, AD is based on the chain rule of differentiation. In AD algorithms, however, the chain rule is often applied in reverse direction. Depending upon the structure of the function, the direction of application can lead to considerable differences in efficiency. For example, a famous result in AD theory states that the evaluation cost of a gradient of a function by reverse automatic differentiation is only a small multiple of the cost of the evaluation of the function itself. This means that the cost is independent of the number of input variables of the function, whereas for numerical differentiation this cost is linear in the number of input variables. Automatic differentiation also avoids the expression swell commonly encountered with symbolic differentiation. The memory requirements of reverse automatic differentiation may, however, be proportional to the number of computation steps in the function. For very large codes, this can be a problem. Algorithms that reduce these requirements, for example by means of so-called checkpointing, are currently subject of active research.

In some approaches to model analysis, the computation of derivatives is intricately entangled with the analysis method itself. This is the case with the so-called adjoint method for data assimilation. This method and its relation to automatic differentiation will also be considered in this thesis.

## 1.2 Sensitivity analysis

We start out the overview of derivative-based methods for model analysis by taking a sensitivity analysis point of view. Sensitivity analysis deals with 'What if...?'-scenarios, for example with questions of the type 'What happens to my result if I change the value of this input variable by so much?' In this respect, sensitivity analysis is very widely applicable, and is mathematically uncomplicated in the sense that one can compute results for different scenarios and simply compare the outcomes. A possible outcome could be for instance the finding of sensitive dependence on initial conditions in the case of a chaotic system.

In practice, however, it may not be feasible to consider many scenarios and it will be beneficial to use a systematic approach for both the choice of scenarios and the parsimonious use of computational resources.

The most straightforward methods of sensitivity analysis are based on sampling. In these methods samples are taken using the probability distribution of the input variables, then the output variables associated with the sample elements are calculated, and these are used to approximate the probability distribution of the output variables. Generally, however, these methods require a large number of evaluations of the function relating the input variables to the output variables. We will focus on local methods that require only the evaluation of the function at variable values of interest, and of function derivatives at these values.

The local methods proceed in two steps:

1. Construct a local approximation for the function relating input to output

variables. For both explicit and implicit relations derivatives for a Taylor expansion that can serve as the approximation can be obtained automatically.

2. Use this approximation for the propagation of the uncertainty of the input variables to the uncertainty of the output variables. Output uncertainty can be decomposed into contributions of the sources of input uncertainty.

This subdivision of analysis will turn out to be useful in the discussion on statistical inference for nonlinear models where the same methods are used for the computation of estimation bias and estimation variance-covariance.

## 1.3 Statistical inference and model selection

Parameter estimation requires a mathematical description of the relationship between parameters and the probability distribution of the observed variables. To assess if model and data are together sufficiently informative with respect to the parameters, this relationship should be subject to both pre-data and post-data investigation. We will approach these investigations mainly from a likelihood perspective, stressing the importance of the likelihood ratio for the assessment of relative support provided by the data for the alternative hypotheses.

Inference computations are, in general, more complicated if the relationship between observation variables and parameters is taken to be nonlinear. Several derivative-based methods will be discussed to quantify model nonlinearity and to obtain accurate inferential estimates in the presence of nonlinearity. We also present a method to compute the non-asymptotic moments of maximum likelihood estimators for nonlinear and misspecified models.

Another important issue in likelihood estimation is the balance between underfitting and overfitting: the complexity of the model structure should be appropriate for the amount and quality of the available data. To this end we discuss model selection criteria which provide a 'rate of exchange' between support and model simplicity.

## 1.4 General objectives and outline of the thesis

The general objectives of the thesis can be summarized as follows:

1. To present and develop derivative-based methods for sensitivity analysis of modeling outcomes, and to investigate their relative merits with respect to other types of methods, in particular in the context of nonlinear parameter estimation.

2. To present and develop statistical methods that allow for the computation of accurate inferential information for nonlinear models, and for the assessment of the appropriateness of model structure complexity in relation to quality of data.

3. To develop a C++ library of routines for model analysis based on automatic differentiation, and to show the benefits of implementation by means of automatic differentiation.

4. To demonstrate the application of the derivative-based methods and the routines of the library by means of real-life case studies.

To realize these objectives, the subsequent chapters are organized in the following manner:

**Chapter 2** discusses the principles underlying automatic differentiation algorithms. Its aim is to provide the reader with both a concise introduction to the subject and the necessary background information to understand the content of this thesis.

**Chapter 3** presents an overview of local sensitivity analysis based on first and higher-order derivative information and of its implementation using automatic differentiation. A detailed description is provided of a derivative-based method for the propagation of uncertainty information and, for the subsequent uncertainty decomposition. The properties of this method are studied and its performance is compared to global decomposition methods. For a case study on the carbon contribution of forest phytomass this method is applied to investigate so-called upscaling sensitivity. Also two new elements of sensitivity analysis are introduced, precision sensitivity and implicit feature sensitivity, as further examples on the use of derivative information for sensitivity analysis.

**Chapter 4** addresses methods for nonlinear parameter estimation. Two subjects take a central place: statistical inference and model selection. First, estimation and statistical inference are introduced from a likelihood perspective. Next, it is shown that strict likelihood maximization is generally not desirable and methods are presented to select models with an appropriate balance between support from the data and structural parsimony. In the remainder of the chapter, the influence of nonlinearity on these issues is investigated in more detail. Measures of nonlinearity and various methods to deal with model nonlinearity in statistical inference are described.

**Chapter 5** studies a maximum likelihood estimation procedure used for the stock assessment of North Sea herring. Several methods discussed in the previous chapters are applied in this case study. The case study aims to demonstrate that the implementation (and re-implementation) of large ecological assessment procedures by means of the MAP library is feasible.

**Chapter 6** provides a study on the use of the adjoint method of data assimilation for the Virtual Population Analysis stock assessment procedure. It is shown that the discrete adjoint method is equivalent to reverse automatic computation of

gradients. A case study is presented on the stock assessment of the Pacific halibut species.

**Chapter 7** describes the MAP library as well as the characterization of nonlinear parameter estimation on which its design is based. A number of experiments based on the case study of Chapter 5 are discussed that demonstrate the performance of the library routines. The development of the library has also provided inspiration on a number of ideas on the design of automatic differentiation implementations.

**Chapter 8** provides a summary and conclusions.

# Chapter 2

# Automatic differentiation

## 2.1 Introduction

In this chapter we will discuss the principles of automatic differentiation (AD). Our aim is to provide the reader with both a concise introduction to the subject and the necessary background information to understand the content of this thesis.

For anyone who wants to get acquainted with the technical details of the subject, the current standard work is Griewank (2000). A wealth of material on AD can also be found in the proceedings of the international workshops on automatic differentiation held in Breckenridge 1991 (Griewank and Corliss, 1991), Santa Fe 1996 (Berz et al., 1996) and Nice 2000 (Corliss, 2001). A historical overview of the development of the subject is given in Iri (1991).

We start by exploring the general concept of the derivative of a function, i.e. the result of the differentiation of a function. Let $f : \mathbb{R}^n \to \mathbb{R}^m$ be a function with domain $\mathbb{R}^n$ (or an open subset of $\mathbb{R}^n$) that maps into $\mathbb{R}^m$. The derivative of $f$ is a function $D_x f : \mathbb{R}^n \to \mathcal{L}(\mathbb{R}^n, \mathbb{R}^m)$ where $\mathcal{L}(\mathbb{R}^n, \mathbb{R}^m)$ is the function space of linear maps of $\mathbb{R}^n$ into $\mathbb{R}^m$. The elements of the standard matrix representation of the linear map $D_x f(x)$ can be explicitly denoted by

$$((D_x f)(x))_{ij} = \frac{\partial f_i}{\partial x_j}(x), \tag{2.1}$$

where

$$\frac{\partial f_i}{\partial x_j}(x) = \lim_{h \to 0} \frac{f_i(x + he_j) - f_i(x)}{h}, \quad i = 1, \ldots, m, \quad j = 1, \ldots, n, \tag{2.2}$$

and where $e_j$ is the $j$-th unit vector and $h$ a real number. The definition states that, at each point $x$ of the domain of $f$, the derivative can be considered as a linear function $D_x f(x)$ that maps $\mathbb{R}^n$ into $\mathbb{R}^m$, provided, of course, that all limits in (2.2) exist. From now on, we will assume that the limits indeed exist. Thus, functions are assumed to be sufficiently smooth to have first and, if required also, higher order derivatives unless explicitly stated otherwise.

An alternative characterisation of the derivative of a function is that at each point of the domain, the derivative constitutes a local approximation of the function itself by means of a linear map, in the sense that

$$f(x+h) - f(x) = f'(x)h + r(h), \tag{2.3}$$

where $h$ is now a vector in $\mathbb{R}^n$ and the remainder $r(h)$ satisfies

$$\lim_{h \to 0} \frac{|r(h)|}{|h|} = 0. \tag{2.4}$$

This property can also be used to define differentiability, see for instance Rudin (1976) or Dieudonné (1960). The property that the derivative is locally an approximation to the function itself, is exactly what makes derivatives so useful in so many fields of application, such as optimization, iterative solving of nonlinear equations and bifurcation analysis and, as will be investigated in this thesis, also in uncertainty analysis of mathematical models.

Higher order derivatives, which are derivatives of derivatives, can be defined in a manner analogous to first order derivatives. The second order derivative, for instance, is denoted by $D^2_{xx} f : \mathbb{R}^n \to \mathcal{L}(\mathbb{R}^n, \mathcal{L}(\mathbb{R}^n, \mathbb{R}^m))$. Explicitly, the elements of $D^2_{xx} f(x)$ are given by,

$$(D^2_{xx} f(x))_{ij_1j_2} = \frac{\partial^2 f_i}{\partial x_{j_1} \partial x_{j_2}}(x), \quad i = 1, \ldots, m, \quad j_1, j_2 = 1, \ldots, n. \tag{2.5}$$

The second derivative constitutes a local linear approximation of the first derivative of $f$, in the sense that it describes how $D_x f$ changes in a neighborhood of a domain point $x$.

In this chapter we will focus on the computation of derivatives of functions that are implemented as computer code. In fact, the subject of study in automatic differentiation are the algorithms for generating code for the derivatives from the code of the function itself. As we will see, these algorithms offer high precision and efficiency, while being based on simple principles.

To describe these algorithms, we will use various abstractions to represent the computer code computations, avoiding the the need to refer to a single specific computer language, or AD implementation. To be useful in practice, these abstractions also have to deal with difficulties such as for-loops, iterative assignments and program branching.

In the AD algorithms the derivative computation of a function ultimately rests on two ideas:

1. The function is a composition of elementary operations with known derivatives.

2. The derivative of a composition of functions is equal to the composition of the derivatives of the composing functions.

The first idea is that the function is given explicitly as a sequence of elementary operations with known derivatives. As elementary operations one can think of standard binary operations such as addition and multiplication, standard unary operations, e.g. taking the negative or reciprocal of a variable, and standard univariate functions such as the sine and exponential functions. The second idea is in fact the chain rule which states that the local linear approximation of a composition of functions is equal to the composition of the local linear approximations of the composing functions.

There are two directions, contravariant and covariant, in which the chain rule may be used. These correspond to the two main types of automatic differentiation: forward and reverse. These will be discussed in section 2.4 and section 2.5, respectively.

In the course of this chapter we will discuss various criteria with respect to which the quality of various algorithms for computing derivatives can be judged. In section 2.6 the complexity of the forward and reverse differentiation algorithms are considered in terms of operation count and memory requirements. Two remarkable results with respect to the performance of AD algorithms already deserve to be mentioned here. To this end the algorithms are compared to the finite difference method for numerical differentiation.

First, the finite diffence method is based on definition 2.2 to arrive at an approximation to a partial derivative. Instead of taking the limit $h \to 0$ as in this definition, a small value for $h$ is substituted to obtain the difference quotient $(D_x f)_{h,ij}(x)$ giving

$$(D_x f)_{h,ij}(x) = \frac{f_i(x + he_j) - f_i(x)}{h}. \tag{2.6}$$

In choosing $h$ there is a trade-off between the truncation error due to the approximation and the roundoff error. In contrast, the AD algorithms generate code for the exact derivative, so no additional error through approximation is introduced.

Second, consider a function with $n$ independent (input) variables that maps into a single dependent (output) variable. We can then consider the operation count required for the evaluation of the function derivative relative to the operation count for the evaluation of the function itself. For the method of finite differences the cost to compute the derivative is a multiple *linear* in $n$ of the function evaluation cost. For the reverse AD algorithm, however, the operation count required to compute the derivative is bounded by a small *constant* multiple of the operation count of the function itself. This multiple does not depend on the number of input variables $n$.

We further note that AD is not the same as symbolic differentiation. As will become clear in the course of this chapter, using AD algorithms only a few numerical values are computed and stored. In symbolic differentiation programs entire expressions must be stored and usually no advantage is taken of common sub-expressions. For large computer codes this leads, in general, to memory requirements that cannot be met.

After discussing the complexity properties we look at the various possible techniques for implementation of AD algorithms in section 2.7. In section 2.8 we discuss a number of additional topics related to automatic differentiation.

## 2.2 An example function

In the following sections the application of the AD algorithms will be illustrated by means of an example function. To this end we take Richards' function

$$W(t) = \frac{A}{(1 + be^{-kt})^{\theta}} \tag{2.7}$$

with parameters $A$, $b$, $k$ and $\theta$. The function $W(t)$ is the solution of the differential equation

$$\frac{dW}{dt} = k\theta W(1 - \left(\frac{W}{A}\right)^{\frac{1}{\theta}}), \qquad W(0) = \frac{A}{(1+b)^{\theta}}. \tag{2.8}$$

Notice that the parameter $b$ only appears in the initial condition of the differential equation. Richards' function was first described in Richards (1959) as a flexible growth function for empirical use. Fitting of this function to data is discussed in Nelder (1961). It was further used as the main example in Clarke (1980) on moments of least-squares estimators in nonlinear regression models.

The function is used to model limited growth. It includes several other well-known growth functions as special cases: for $\theta = -1$ we have the 'monomolecular' (Von Bertalanffy) growth function $W = A(1 - be^{-kt})$ (the sign of $b$ is reversed here). For $\theta = 1$ we obtain the logistic growth function $W = A/(1 + be^{-kt})$ and for $\theta \to \infty$ we get the Gompertz curve $W = Ae^{-be^{-kt}}$. Also the exponential curve can be obtained by taking $\theta \downarrow 0$ and keeping $k\theta$ constant while $k \to \infty$.

As in Chapter 3, we use Richards' function to model the growing stock volume of trees as a function of age. To make the example slightly more simple, we assume that $\theta = 1$ and $b = 1$ , i.e. we take

$$V = \frac{A}{1 + e^{-kt}} \tag{2.9}$$

for the growing stock volume $V$ (in $m^3$). We then use two mass fractions to compute the phytomass corresponding to two tree phytomass types, say stem phytomass and green leaves phytomass, according to

$$M_s = f_s V = \frac{f_s A}{1 + e^{-kt}}, \tag{2.10}$$

$$M_l = f_l V = \frac{f_l A}{1 + e^{-kt}}, \tag{2.11}$$

where $M_s$ and $M_l$ are the stem and leaves phytomass (in kg), and $f_s$ and $f_l$ are the stem and leaves phytomass densities (in $kg/m^3$), respectively.

The function $f$ which we will use for the examples, is then given by

$$f : \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix} = \begin{pmatrix} k \\ A \\ t \\ f_s \\ f_l \end{pmatrix} \rightarrow \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} M_s \\ M_l \end{pmatrix} = \begin{pmatrix} \frac{f_s A}{1+e^{-kt}} \\ \frac{f_l A}{1+e^{-kt}} \end{pmatrix} \tag{2.12}$$

In this chapter we shall refer to this function as the phytomass map.

## 2.3 Function representations

As already indicated in the introduction, AD algorithms rely on a clever application of the chain rule. In order to demonstrate this, we will need a representation of the function to be differentiated that explicitly represents the function as a sequence of simple functions. This will realized by the *evaluation procedure* of a function, see Algorithm 2.3.1. The definition is taken from Griewank (2000). In the remainder of this chapter we will rely on his notation for the description of the forward and reverse differentiation algorithms. An evaluation procedure consists

| | | | | | |
|---|---|---|---|---|---|
| $v_{i-n}$ | $=$ | $x_i,$ | $i$ | $=$ | $1 \ldots n$ |
| $v_i$ | $=$ | $\phi_i(v_j)_{j \prec i},$ | $i$ | $=$ | $1 \ldots l$ |
| $y_i$ | $=$ | $v_{l-m+i},$ | $i$ | $=$ | $1 \ldots m$ |

Algorithm 2.3.1: General evaluation procedure of a function $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$, $y = f(x)$. The function is represented as a sequence of elementary operations for a given value of $x$.

of three stages. First the $n$ input values $x_1$ through $x_n$ of the independent variables are assigned to variables $v_{-n+1}$ through $v_0$. Then, one by one, the variables $v_1$ through $v_l$ are computed using the elementary functions $\phi_i, i = 1, \ldots, l$. In the remainder of this chapter we will refer to such a computation as a *computation step*. The notation $j \prec i$ is used to denote the set of indices $j < i$ for which $v_i$ depends directly on $v_j$ through $\phi_i$, i.e. for $j \prec i$ we have that $v_j$ is an argument of $\phi_i$. In the last stage, the $m$ variables $v_{l-m+1}$ through $v_l$ are assigned to the dependent variables $y_1$ through $y_m$, i.e. it is assumed that the $m$ variables computed last, take the values of the output variables. The notation $i = 1 \ldots n$ denotes that $i$ takes the values 1 through $n$ in sequence.

As an example, Algorithm 2.3.2 shows an evaluation procedure for the phytomass map, which was described in section 2.2.

$$
\begin{array}{rcl}
v_{-4} & = & x_1 = k \\
v_{-3} & = & x_2 = A \\
v_{-2} & = & x_3 = t \\
v_{-1} & = & x_4 = f_s \\
v_0 & = & x_5 = f_l \\
\hline
v_1 & = & v_{-2} \cdot v_{-4} \\
v_2 & = & -v_1 \\
v_3 & = & \exp(v_2) \\
v_4 & = & 1 + v_3 \\
v_5 & = & 1/v_4 \\
v_6 & = & v_{-3} \cdot v_5 \\
v_7 & = & v_{-1} \cdot v_6 \\
v_8 & = & v_0 \cdot v_6 \\
\hline
y_1 & = & v_7 = M_s \\
y_2 & = & v_8 = M_l
\end{array}
$$

Algorithm 2.3.2: An evaluation procedure for the phytomass map.

General functions represented by computer code may be using a variety of selection and repetition structures, such as for example corresponding to `if` and `for` statements. For known values of the input variables, however, all conditionals evaluate to a fixed value and the code may be expanded into a sequence of assignments of the kind represented by the evaluation procedure. Notice, however, that the evaluation procedure is not unique. Here, the only condition that we impose on the evaluation procedure is that the derivatives of the elementary operations are known. We will return to the issue of decomposing a piece of computer code into a sequence of elementary operations in section 2.7, where the implementation of automatic differentiation is discussed.

We assume that for the variables $v_i$ space is declared in memory. If a variable value is no longer needed, overwriting of this space is allowed. See Griewank (2000) for the details on memory addressing consistency.

The dependency relations between the variables $v_i$ can be depicted by means of a *computational graph*. See Figure 2.1 for a computational graph corresponding to the evaluation procedure of Algorithm 2.3.2. Each vertex corresponds to a variable. There is a connection from $v_j$ to $v_i$ if $j \prec i$, i.e. if the function $\phi_i$ depends on $v_j$. Since each $v_i$ may only depend on variables with smaller index the computational graph is acyclic.

Using the function representations discussed in this section we will now describe the two main types of automatic differentiation: forward and reverse differentiation. These names correspond to the direction in which the chain rule is applied to the evaluation procedure composition.

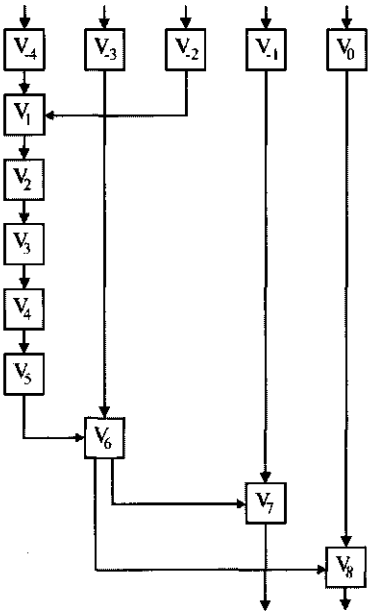



Figure 2.1: Computational graph of the phytomass map corresponding to the evaluation procedure of Algorithm 2.3.2.

## 2.4 Forward differentiation

Forward differentiation is conceptually the most simple of the two types. We consider the function $f : \mathbb{R}^n \to \mathbb{R}^m$ with an evaluation procedure given by Algorithm 2.3.1. For ease of understanding we first assume a one dimensional domain space, i.e. we take $n = 1$. The single independent or input variable is named $t$ here. We are interested in computing

$$\dot{y}_i = \frac{dy_i}{dt}, \quad i = 1, \ldots, m. \tag{2.13}$$

To this end, we associate with each variable $v_i$ of Algorithm 2.3.1 a variable $\dot{v}_i$ defined by

$$\dot{v}_i = \frac{dy_i}{dt}, \tag{2.14}$$

for $i = 1, \ldots, l$. The variables $\dot{v}_i$ can be computed in the same order as the $v_i$ are in the evaluation procedure. This follows from the chain rule, i.e. we use

$$\dot{v}_i = \sum_{j \prec i} \frac{\partial \phi_i}{\partial v_j} \dot{v}_j. \tag{2.15}$$

For the case of an $n$ dimensional domain space, a slightly more general procedure is taken. Instead of considering one independent variable at a time, it is possible

to consider one domain space direction at a time. To this end, we first define the *directional derivative* corresponding to a direction vector $\dot{x}$. Let $\dot{x}$ be a vector in $\mathbb{R}^n$ of length one, then the associated directional derivative of $f_i$ is defined by

$$D_{\dot{x}} f_i(x) = \lim_{h \to 0} \frac{f_i(x + h\dot{x}) - f_i(x)}{h}, \quad i = 1, \ldots, m. \tag{2.16}$$

It can be shown that

$$D_{\dot{x}} f_i = ((D_x f_i)(x))\dot{x}. \tag{2.17}$$

The directional derivative thus constitutes a linear combination of the partial derivatives of $f_i$. A more geometric view of the directional derivative can be obtained by considering a curve $x(t)$ in the domain space of $f$ which passes for $t = 0$ through the point $x$, at which we want to compute the derivative. If the tangent vector of this curve at $t = 0$ is given by $\frac{dx}{dt} = \dot{x}$, then

$$\dot{y} = \frac{d}{dt} y_i(x(t))|_{t=0} = (D_x f_i)(x)\dot{x} = D_{\dot{x}} y_i(x). \tag{2.18}$$

This means that $\dot{y}$ is the tangent vector to the curve $y(x(t))$ at $y(x)$ in the range space of $f$. For this reason forward differentiation is also called tangent propagation.

To compute the tangent vector $\dot{y}$ corresponding to the tangent vector $\dot{x}$, we now associate with each variable $v_i$ of the evaluation procedure a *tangent* variable $\dot{v}_i$. This tangent variable is defined in the same manner as $\dot{y}_i$, namely as

$$\dot{v}_i = D_{\dot{x}} v_i(x) = (D_x v_i)(x)\dot{x}, \tag{2.19}$$

or as, using the curve $x(t)$ with tangent vector $\dot{x}$ at $t = 0$,

$$\frac{d}{dt} v_i(x(t)) \quad i = 1, \ldots, l. \tag{2.20}$$

Notice that if we again take a function with only one independent variable, the tangent variable $\dot{v}_i$ is simply a multiple of the derivative of $v_i$ with respect to this variable.

Now, consider Algorithm 2.4.1.

Again the algorithm consists of three stages. In the first stage the variables are initialized. The tangent variables are set to their chosen values to specify the required directional derivative. In the second stage the real work is done. Just as in the evaluation procedure the variables $v_1$ through $v_l$ are computed one by one. The tangent variables can be computed in the same order. From differentiation of the expression

$$v_i = \phi(v_j)_{j \prec i} \tag{2.21}$$

by means of the chain rule, it follows that

$$\dot{v}_i = \sum_{j \prec i} \frac{\partial}{\partial v_j} \phi_i \, \dot{v}_j, \quad i = 1, \ldots, l. \tag{2.22}$$

$$
\begin{array}{rcll}
v_{i-n} & = & x_i; & \\
\dot{v}_{i-n} & = & \dot{x}_i, & i = 1 \ldots n \\
\hline
v_i & = & \phi_i(v_j)_{j \prec i}; & \\
\dot{v}_i & = & \sum_{j \prec i} \frac{\partial}{\partial v_j} \phi_i \, \dot{v}_j, & i = 1 \ldots l \\
\hline
y_i & = & v_{l-m+i}; & \\
\dot{y}_i & = & \dot{v}_{l-m+i}, & i = 1 \ldots m
\end{array}
$$

Algorithm 2.4.1: Forward differentiation of evaluation procedure.

The order in which a tangent variable $\dot{v}_i$, and the variable $v_i$ are computed, can make a difference in efficiency depending on the elementary operation used. We will not consider this implementation issue here.

In the last stage, the resulting variables and their associated tangent variables are assigned to the appropriate output variables.

In Algorithm 2.4.2 we have listed the forward differentiation algorithm for the computation of the time derivative of the phytomass map.

$$
\begin{array}{rclrcl}
v_{-4} & = & x_1 = k; & \dot{v}_{-4} & = & \dot{x}_{-4} = 0 \\
v_{-3} & = & x_2 = A; & \dot{v}_{-3} & = & \dot{x}_{-3} = 0 \\
v_{-2} & = & x_3 = t; & \dot{v}_{-2} & = & \dot{x}_{-2} = 1 \\
v_{-1} & = & x_3 = f_s; & \dot{v}_{-1} & = & \dot{x}_{-1} = 0 \\
v_0 & = & x_4 = f_l; & \dot{v}_0 & = & \dot{x}_0 = 0 \\
\hline
v_1 & = & v_{-2} \cdot v_{-4}; & \dot{v}_1 & = & v_{-2} \cdot \dot{v}_{-4} + \dot{v}_{-2} \cdot v_{-4} \\
v_2 & = & -v_1; & \dot{v}_2 & = & -\dot{v}_1 \\
v_3 & = & \exp(v_2); & \dot{v}_3 & = & \exp(v_2) \cdot \dot{v}_2 \\
v_4 & = & 1 + v_3; & \dot{v}_4 & = & \dot{v}_3 \\
v_5 & = & 1/v_4; & \dot{v}_5 & = & -1/v_4^2 \cdot \dot{v}_4 \\
v_6 & = & v_{-3} \cdot v_5; & \dot{v}_6 & = & v_{-3} \cdot \dot{v}_5 + \dot{v}_{-3} \cdot v_5 \\
v_7 & = & v_{-1} \cdot v_6; & \dot{v}_7 & = & v_{-1} \cdot \dot{v}_6 + \dot{v}_{-1} \cdot v_6 \\
v_8 & = & v_0 \cdot v_6; & \dot{v}_8 & = & v_0 \cdot \dot{v}_6 + \dot{v}_0 \cdot v_6 \\
\hline
y_1 & = & v_7; & \dot{y}_1 & = & \dot{v}_7 \\
y_2 & = & v_8; & \dot{y}_1 & = & \dot{v}_8
\end{array}
$$

Algorithm 2.4.2: Forward differentiation for the phytomass map.

In section 2.6 we will go into the details of the complexity of the forward differentiation algorithm. Notice that to obtain the complete derivative of $f$, this algorithm has to be applied $n$ times, once for each unit vector taken as input tangent vector $\dot{x}$.

## 2.5 Reverse differentiation

In the forward differentiation algorithm 2.5.1, the derivative of the output quantities is computed for one direction in domain space, i.e. for one linear combination of the independent variables, at a time. For the reverse method we again take one direction at a time, now in the range space of $f$. This direction will be denoted by $\bar{y}$, where $\bar{y}' \in \mathbb{R}^m$, i.e. $\bar{y}$ is a row vector. Using $\bar{y}$, we consider the derivative of one linear combination of the components of $f$, i.e. of the $m$ dependent variables, at a time. The derivative is taken with respect to the $n$ independent variables. So, we define

$$\bar{x}_j = \frac{\partial(\bar{y}y)}{\partial x_j} (= \bar{y}(D_x f)(x)), \quad j = 1, \ldots, n. \tag{2.23}$$

As in the treatment of forward differentiation, the now following procedure will be easier to understand if a unit vector is taken for $\bar{y}$. In this case $\bar{y}y = e_i' y = y_i$, and $\bar{x}$ represents the gradient of this component with respect to the independent variables. So the reader may consider $\bar{y}y$ simply as one of the components of $y$ (or, if $m = 1$, as $y$ itself).

For forward differentiation we introduced tangent variables as the directional derivatives of the corresponding variables. For reverse differentiation we associate with each variable $v_i$ an *adjoint variable* which is defined as

$$\bar{v}_i \equiv \frac{\partial(\bar{y}y)}{\partial v_i}, \quad i = (1-n), \ldots, l. \tag{2.24}$$

This is a somewhat peculiar notation, since $\bar{y}y$ does not formally depend on $v_i$: $\bar{y}y$ and $v_i$ both depend on $x$. This can be circumvented by introducing a new variable $\delta_i$ in the computation of $v_i$, according to

$$v_i = \phi_i(v_j)_{j \prec i} + \delta_i, \tag{2.25}$$

and by letting $y$ also depend on this variable. We then take

$$\bar{v}_i = \frac{\partial(\bar{y}y)}{\partial \delta_i} \equiv \frac{\partial(\bar{y}y)}{\partial v_i}. \tag{2.26}$$

Now consider a certain variable $v_j$ and suppose that we already know all adjoint variables $\bar{v}_i$ for $i > j$. We then introduce the disturbance $\delta_j$ in the computation of $v_j$ as in (2.25). This disturbance $\delta_j$ of $v_j$ causes a disturbance $\delta_j \frac{\partial \phi_i}{\partial v_j}$ in each variable that directly depends on the variable $v_j$. Since we already know the adjoint variables of these variables, it is then possible to compute the total disturbance on the quantity $\bar{y}y$ through

$$\delta(\bar{y}y) = \sum_{i \succ j} \bar{v}_i \frac{\partial \phi_i}{\partial v_j} \delta_j. \tag{2.27}$$

Here $i \succ j$ means all indices $i$ for which $v_i$ directly depends on $v_j$. It follows that

$$\bar{v}_j = \frac{\partial(\bar{y}y)}{\partial \delta_j} = \sum_{i \succ j} \bar{v}_i \frac{\partial \phi_i}{\partial v_j}. \tag{2.28}$$

This means that we can compute adjoint variables in the direction reverse to that in which the variables $v_i$ are computed. This is accomplished in algorithm 2.5.1. The derivation and notation are again for a large part taken from Griewank (2000).

| | | | | | |
|---|---|---|---|---|---|
| $v_{i-n}$ | $=$ | $x_i,$ | $i$ | $=$ | $1 \ldots n$ |
| **push** | $v_i;$ | | | | |
| $v_i$ | $=$ | $\phi_i(v_j)_{j \prec i},$ | $i$ | $=$ | $1 \ldots l$ |
| $y_i$ | $=$ | $v_{l-m+i},$ | $i$ | $=$ | $1 \ldots m$ |
| $\bar{v}_i$ | $=$ | $0,$ | $i$ | $=$ | $(1-n) \ldots (l-m)$ |
| $\bar{v}_{l-m+i}$ | $=$ | $\bar{y}_i,$ | $i$ | $=$ | $1 \ldots m$ |
| **pop** | $v_i;$ | | | | |
| $\bar{v}_j$ | $+=$ | $\bar{v}_i \frac{\partial}{\partial v_j} \phi_i$ for $j \prec i,$ | $i$ | $=$ | $l \ldots 1$ |
| $\bar{x}_i$ | $=$ | $\bar{v}_{i-n},$ | $i$ | $=$ | $1 \ldots n$ |

Algorithm 2.5.1: Reverse differentiation of evaluation procedure.

In order to understand this algorithm we first ignore the **push** and **pop** statements. Further, notice that

$$\frac{\partial(\bar{y}y)}{\partial y_i} = \bar{y}_i, \quad i = 1, \ldots, m. \tag{2.29}$$

Since we have from the evaluation procedure that

$$y_i = v_{l-m+i}, \quad i = 1 \ldots m, \tag{2.30}$$

we can already set

$$\bar{v}_{l-m+i} = \bar{y}_i, \quad i = 1 \ldots m. \tag{2.31}$$

Now, the first three stages of the algorithm constitute the so-called forward sweep. It computes the variables $v_i$ by means of the general evaluation procedure. The last three stages are called the return sweep. In the return sweep the adjoint variables will be computed in a direction opposite to that of the evaluation procedure. We saw in (2.28) that if the adjoint variables are known for $i > j$, then also $\bar{v}_j$ can be computed. It follows from the same equation that the assignment statement for $v_i$ makes a contribution of $\bar{v}_i \frac{\partial \phi_i}{\partial v_j}$ to the adjoint variable $\bar{v}_j$. We can then go through the evaluation procedure in opposite direction, for each assignment statement updating the adjoint variables corresponding to the variables occuring at the righthandside of the assignment. The notation += is taken from C++ and

means 'add the righthand side to the variable on the lefthand side'. After completing this procedure for all assignment statements, the derivatives $\bar{x}_j$ have obtained their correct value. These derivatives together form the gradient of $(\bar{y}y)(x)$,

To further illustrate this procedure, the return sweep for the phytomass map is given in Algorithm 2.5.2 for the case that the gradient of the leaves phytomass fraction is computed.

We now turn to the **push** and **pop** statements in the algorithm. We have seen that for computation of derivatives by means of reverse differentiation we need to go through the statements of the evaluation procedure in reverse order. How this is realized is mainly an implementation issue, and this problem can be solved for an important part at compile-time. We will discuss these solutions further in section 2.7. It must be possible, however, to allow the values of the independent variables to be known only at runtime. This means that at least the values needed for the return sweep, which depend on the independent variables, must be stored as part of the algorithm.

In order to be able to perform the adjoint variable update statements for the computation step $i$, information must be available on (i) the elementary operation $\phi_i$, (ii) its argument variables, and (iii) the values of these argument variables. How the first two pieces of information are made available is discussed in section 2.7, while the values of the arguments are available through the **push/pop** construction in the algorithm. With **push** $v_i$ the value of $v_i$ is pushed on a stack device. This stack device is usually referred to as a *tape* in the AD literature. With **pop** $v_i$ the variable $v_i$ is given the value that was stored on the tape with the **push** statement and this value is then removed from the tape. The value of each variable $v_i$ is pushed on the tape before it is computed in its associated computation step. Consequently, after each **pop** statement in the return sweep, all variables have exactly the same value as just before the corresponding computation step in the evaluation procedure. This is so even if overwriting of the variable by subsequent statements has taken place. Notice that the same could have been accomplished by putting at each computation step all the argument values on a tape. This would, in general, require more memory operations.

## 2.6 Algorithmic complexity

The complexity of the differentiation algorithms studied in the previous two sections can be related to the complexity of the corresponding evaluation procedure. In this respect, we will both consider the runtime performance and the memory requirements of the forward and reverse differentiation algorithms. For implementations of certain types there are additional runtime costs at the return sweep of the reverse differentiation algorithm. This is due to the recording of information on the structure of the evaluation procedure at runtime. These will be discussed in the next section, here we neglect such costs.

First, we compare the operations count for the differentiation algorithms and the evaluation procedure. Here we consider required numbers of floating point

| | | | | | |
|---|---|---|---|---|---|
| $\bar{v}_8$ | $=$ | $\bar{y}_2 = 0$ | | | |
| $\bar{v}_7$ | $=$ | $\bar{y}_1 = 1$ | | | |
| $\bar{v}_6$ | $+=$ | $v_0 \cdot \bar{v}_8;$ | $\bar{v}_0$ | $+=$ | $v_6 \cdot \bar{v}_8$ |
| $\bar{v}_6$ | $+=$ | $v_{-1} \cdot \bar{v}_7;$ | $\bar{v}_{-1}$ | $+=$ | $v_6 \cdot \bar{v}_7$ |
| $\bar{v}_5$ | $+=$ | $v_{-3} \cdot \bar{v}_6;$ | $\bar{v}_{-3}$ | $+=$ | $v_5 \cdot \bar{v}_6$ |
| $\bar{v}_4$ | $-=$ | $1/v_4^2 \cdot \bar{v}_5$ | | | |
| $\bar{v}_3$ | $+=$ | $\bar{v}_4$ | | | |
| $\bar{v}_2$ | $+=$ | $\exp(v_2) \cdot \bar{v}_3$ | | | |
| $\bar{v}_1$ | $-=$ | $\bar{v}_2$ | | | |
| $\bar{v}_{-2}$ | $+=$ | $v_{-4} \cdot \bar{v}_1;$ | $\bar{v}_{-4}$ | $+=$ | $v_{-2} \cdot \bar{v}_1$ |
| $\bar{x}_1$ | $=$ | $\bar{v}_{-4}$ | | | |
| $\bar{x}_2$ | $=$ | $\bar{v}_{-3}$ | | | |
| $\bar{x}_3$ | $=$ | $\bar{v}_{-2}$ | | | |
| $\bar{x}_4$ | $=$ | $\bar{v}_{-1}$ | | | |
| $\bar{x}_5$ | $=$ | $\bar{v}_0$ | | | |

Algorithm 2.5.2: Return sweep for the phytomass map.

operations, and of memory stores and fetches. The relative cost of performing different floating point operations and different types of memory access is architecture dependent. The type of memory used is becoming increasingly important in this respect. Griewank (2000) presents a detailed analysis in which such costs are split out into several types. Here we follow a less complicated type of analysis, along the lines of Griewank (1989). We compare the cost of evaluating the function and its derivative in an abstract manner. This is realized by considering, for one computation step of the evaluation procedure at a time, the operations in the two algorithms that are associated with the particular step (see section 2.3 for the definition of a computation step).

We denote the cost of one operation of the evaluation procedure, i.e. the cost of computation step $i$, by $\text{cost}(\phi_i)$, for $i = 1, \ldots, l$. The cost function may be interpreted as needed. One could for instance choose to measure the cost in runtime seconds on a specific architecture, or alternatively, in number of floating point operations performed. The forward and reverse algorithms are organized such, that each of the operations of the algorithm can be associated with one particular computation step. For both algorithms we denote the cost of the operations associated with computation step $i$ by $\text{cost}(\text{task}(\phi_i))$, for $i = 1, \ldots, l$. The complete cost of the algorithms is then denoted by $\text{cost}(\text{task}(f))$, whereas the cost of the evaluation procedure itself is denoted by $\text{cost}(f)$. We now show that if we can bound $\text{cost}(\text{task}(\phi_i))$ relative to $\text{cost}(\phi_i)$, then also $\text{cost}(\text{task}(f))$ can be bounded relative to $\text{cost}(f)$.

We assume that the cost of the evaluation procedure is equal to the sum of the

costs of the computation steps, i.e.

$$\text{cost}(f) = \sum_{i=1}^{l} \text{cost}(\phi_i). \tag{2.32}$$

We consider algorithms that are based on performing operations which are each associated with one of the computation steps of the evaluation procedure and further assume that

$$\text{cost}(\text{task}(f)) = \sum_{i=1}^{l} \text{cost}(\text{task}(\phi_i)). \tag{2.33}$$

To quantify the algorithm complexity we use the *cost ratio* $\mathcal{C}(f)$ defined by

$$\mathcal{C}(f) = \frac{\text{cost}(\text{task}(f))}{\text{cost}(f)}. \tag{2.34}$$

We can also define an *elementary cost ratio* for each of the computation steps

$$\mathcal{C}(\phi_i) = \frac{\text{cost}(\text{task}(\phi_i))}{\text{cost}(\phi_i)}, \quad i = 1, \ldots, l. \tag{2.35}$$

Assume that each elementary cost ratio is bounded by a constant $\omega$, i.e. $\mathcal{C}(\phi_i) \leq \omega$, for $i = 1, \ldots, l$.

We then find that the cost ratio $\mathcal{C}(f)$ satisfies

$$\mathcal{C}(f) = \frac{\text{cost}(\text{task}(f))}{\text{cost}(f)} = \frac{\sum_{i=1}^{l} \text{cost}(\text{task}(\phi_i))}{\sum_{i=1}^{l} \text{cost}(\phi_i)} \leq \frac{\sum_{i=1}^{l} \omega \text{cost}(\phi_i)}{\sum_{i=1}^{l} \text{cost}(\phi_i)} = \omega. \tag{2.36}$$

We conclude that the cost ratio $\mathcal{C}(f)$ is bounded by the worst case of the elementary cost ratios. If the elementary operations are taken from a fixed set of standard operations and functions, then this worst case elementary cost ratio can be determined in advance. In Griewank (1989) it is shown that for reverse differentiation, under a number of reasonable assumptions, for the sine and cosine function the cost ratio lies just above two and for most other system functions it is close to one. Multiplication is the most expensive arithmetic operation with a cost ratio just under 5. It thus follows that under the used assumptions the cost of evaluating a function with one dependent variable and its gradient by means of reverse automatic differentiation is at most 5 times the cost of just evaluating the function itself.

For functions with one dependent variable reverse differentiation is thus the method of choice, since in this case a single use of Algorithm 2.5.1 is sufficient. For functions with one independent variable the forward differentiation algorithm is the natural choice, since in this case Algorithm 2.4.1 needs to be used only once.

In both cases the cost ratio is bounded by the worst case elementary cost ratio. Depending on the assumptions regarding the relative costs of memory and floating point operations, this bound will be a certain small constant. For functions with $n$ independent, and $m$ dependent variables, either $n$ forward, or $m$ reverse calls to the algorithm implementation are required. Several types of mixed strategies, also referred to as accumulation strategies, are possible in order to obtain a higher efficiency in comparison to either pure forward or pure reverse algorithms. The optimal accumulation, however, of full first derivatives is conjectured to be NP hard. See Griewank (1993) and Griewank (2000) for a discussion of mixed accumulation strategies.

In Griewank (1993) bounds are derived for the memory requirements of forward and reverse automatic differentiation. The memory requirement of the forward mode is approximately linear in the number of independent variables in comparison to the memory requirement of the original code. For reverse differentiation the memory requirement is potentially much larger. In a straightforward implementation it may be proportional to the number of computation steps of the evaluation procedure, i.e. proportional to the total runtime of the code. However, Griewank (1992) shows how memory requirements may be reduced by trading these off against runtime performance. This is accomplished by so-called checkpointing, see also Griewank (2000). This method may be used to assure that the tape of the computation is small enough to stay in internal memory.

## 2.7 Implementation of AD algorithms

In sections 2.4 and 2.5 we discussed two types of automatic differentiation algorithms. We now give an overview of how these algorithms are implemented in practice. All AD tools assume a specific source language in which the function to be differentiated is represented. The AD implementation must transform this source representation to a representation of the derivative function which can be translated, at a later stage, in object code. To be able to perform this task the tools have to be presented with certain information. This includes the independent and dependent variables, a specification of the derivatives that are required and, possibly, further information on the structure of the function that can be exploited by the tool.

In Juedes (1991), 29 software packages for automatic differentiation are compared that vary greatly in their possibilities. A first criterion for comparison is whether either forward or reverse differentiation is used, or whether both types are available. In the paper a further five categories of implementation types are discerned. We will distinguish two main implementation types here. We refer to these types as the *preprocessor* and *overloading* types.

A preprocessor is a compiler for which the source and target language are the same. An AD tool of preprocessor type takes code describing a function in a certain language as input and generates code in the same language for the indicated derivatives as output. Software packages that use the preprocessor approach

to AD are for instance GRESS (Horwedel, 1991), PADRE2 (Kubota, 1991), ADIFOR (Bischof et al., 1996) developed at Argonne National Laboratory and Odyssée (Rostaing et al., 1993) developed at INRIA, Sophia Antipolis. These preprocessors all use Fortran 77 as their source language. The preprocessor approach has the advantage that the generated code can be more easily optimized than for the overloading approach, discussed next. For implementation of the return sweep of the reverse differentiation algorithm all information is readily available. Relations between variables can be analyzed by the preprocessor and the code can be optimized accordingly.

For the overloading implementation type no preprocessor is required. Instead a regular compiler is used. The language constructs of operator overloading and polymorphic functions are required to be able to perform the automatic differentiation. To this end a new floating point data type is introduced for which all elementary operations are overloaded. While the function code is evaluated, the structure of the evaluation procedure is recorded. The recorded information may later be used by the AD tool to compute the derivatives of the function. AD tools based on operator overloading are available in several languages: in C++ (Griewank et al., 1996; Bendsten and Stauning, 1996; Aubert et al., 1999), ADA (Bartholemew-Biggs et al., 1994), Fortran90 (Davis et al., 1990) and also in Matlab (Hill and Rich, 1992). The overloading approach is generally easier to understand and to implement. The implementation achievements come within a small constant multiple of the theoretical bounds discussed in the section 2.6. For reverse differentiation this is provided that the code is not so large that slow type memory is needed. This is due to the fact that for the overloading approach also the function structure must be stored on a tape (see Chapter 5 of Griewank (2000)).

For the MAP library discussed in this thesis (see Chapter 7), we have used the ADOL-C library (Griewank et al., 1996). In ADOL-C both forward and reverse automatic differentiation are implemented using the overloading approach. In Chapter 7 the performance of ADOL-C is investigated for a function describing a model for the North Sea herring fishery. In Program 2.7.1 an example of C++ code that uses the ADOL-C library is shown. In this example the code for the calculation of the phytomass map is extended by some additional ADOL-C notation in order to be able to use the automatic differentiation routines of the library. The computation takes place between the statements `trace_on` and `trace_off`. The independent and dependent variables are indicated by means of the operators `<<=` and `>>=`, respectively. All variables that depend directly or indirectly on the independent variables are called *active* variables and must be of type `adouble`.

The recorded structure of the evaluation procedure is stored sequentially in main memory and is automatically paged to disk when necessary. During subsequent derivative evaluations, tapes are always accessed sequentially. After construction of the tape, the gradient function of the library is called to evaluate the gradient of the dependent variable with respect to the independent variables using the variable `tag` to indicate the required tape. The gradient may also be computed for different independent variable values by means of the same tape, provided that no new program branches are taken.

```
#include <adolc.h>

int main()
{
    const int n=5;
    adouble* x=new adouble[n];
    double* x_in=new double[n];
    double k=1., A=3., t=5., fs=.3, fl=.4;
    x_in[0]=k; x_in[1]=A; x_in[2]=t; x_in[3]=fs; x_in[4]=fl;

    double y_out;
    double* grad=new double[n];

    \\ Start tracing of computational graph.
    short int tag=1;
    trace_on(tag);

    \\ Indicate independent variables.
    for (int i=0;i<n;i++) x[i]<<=x_in;

    \\ Perform computations using active variables.
    adouble V=x[1]/(1+exp(-x[0]*x[2]));
    adouble Ml=x[3]*V;

    \\ Indicate dependent variable(s).
    Ml>>=y_out;

    \\ End trace of computational graph.
    trace_off(tag);

    \\ Use function gradient from ADOL-C library.
    gradient(tag,n,x,grad);

    delete[] grad; delete[] x_in; delete[] x;
    return 0;
}
```

Program 2.7.1: Example of C++ code using the ADOL-C library.

Similar functions are available for the evaluation of Jacobians, Hessians and higher order derivatives.

## 2.8 Some further topics in AD

In Bischof et al. (1993) a method is described to extract partial derivatives of arbitrary order by interpolating a number of univariate Taylor expansions. Further, in Christianson (1991) it is shown that univariate Taylor series can be propagated by a method very similar to reverse automatic differentiation. Since it is possible to obtain each partial derivative separately, the combination of these methods is ideally suited to exploit sparsity patterns of higher order derivatives. The methods described above have been implemented in the ADOL-C package for automatic differentiation, see Griewank et al. (1996). Chapter 10 of Griewank (2000) presents a summary of these types of techniques.

Automatic differentiation is closely related to a number of other techniques such as the backpropagation method for neural networks and adjoint data assimilation, as used in for instance meteorology. The relation between AD and backpropagation is discussed in Saarinen et al. (1991). Adjoint data assimilation and its relation to AD will be further explored in Chapter 6 of this thesis.

An important area of application of automatic differentiation is that of obtaining accurate roundoff error estimates. See Iri (1991) for a review. Automatic differentiation may also play an important part in the development of interval arithmetics. See Kulisch (1996) for a discussion on how the arithmetic capability and repertoire of computers should be expanded to make optimal use of the recent advances.

# Chapter 3

# Local sensitivity analysis based on automatically generated derivative information

This chapter presents an overview of local sensitivity analysis based on derivative information and of its implementation using automatic differentiation. We show how to obtain useful sensitivity information by means of first and higher-order derivatives of various types of differentiable input-output relationships. We present a detailed description of a derivative-based method for the propagation of uncertainty information and subsequent uncertainty decomposition. Several properties of this method are studied and its performance is compared to global decomposition methods. For a case study on the carbon contribution of forest phytomass this method is applied to investigate so-called upscaling sensitivity. As further examples of derivative-based methods for sensitivity analysis we treat the investigation of implicit relations and introduce precision sensitivity. The analysis types discussed can be implemented in a straightforward manner by means of automatic differentiation.

## 3.1 Introduction

This chapter presents an overview of local sensitivity analysis methods based on derivative information. These methods can benefit from implementation by means of automatic differentiation algorithms. In section 3.2, we first present a brief general overview of the field of sensitivity analysis.

As an illustration of the methods discussed in this chapter we use the so-called State Forest Account (SFA) Method of the Foresty Project at the International Institute for Applied Systems Analysis (IIASA) in Laxenburg, Austria. The SFA method quantifies the growing stock volume and carbon content of the Russian

forests. The results of this method are important for the assessment of the contribution of these forests to the global carbon cycle. In section 3.3 we discuss some technical details of the SFA method that will be needed for the examples later in the chapter.

In section 3.4 we describe local methods for propagation of uncertainty information through functions using their first and higher-order derivatives. In particular we focus on a local propagation method of variance-covariance that can also be used for the decomposition of variance into sources corresponding to the uncertainty of the input variables and their interactions. We apply this method to assess the sensitivity of the results from the SFA Method to upscaling.

In the remainder of the chapter we describe two more sensitivity methods based on derivative information. First, in section 3.5 we propose a method to quantify the effect of changes in the precision of input variables on the precision of the output variables. The precision sensitivity coefficients introduced there turn out to have an interesting interpretation in relation to the variance decomposition methods discussed earlier. Second, in section 3.6, we describe a method for dealing with implicit relations that can be used to quantify the effect of changes in non-parameter variables on parameter estimates obtained by nonlinear regression.

All derivative-based methods of this chapter have been implemented in a C++ library using the ADOL-C library (Griewank et al., 1996) for automatic differentiation.

## 3.2 A brief overview of sensitivity analysis

Sensitivity analysis concerns the investigation of the sensitivity of the results of a computation with respect to its input assumptions. Sensitivity analysis deals with 'what if' scenarios, e.g. with questions of the type 'What happens to my result if I change the value of this input variable by so much?' In this respect it is very widely applicable, and it is mathematically uncomplicated in the sense that one can compute results for different scenarios and simply compare the outcomes.

In practice however it may not be computationally feasible to consider many scenarios and it will be beneficial to use a systematic approach for both the choice of scenarios and the parsimonious use of computational resources. Various overviews of sensitivity analysis are available, see for instance Campolongo et al. (2000) for a short general overview, Helton and Davis (2000) for sampling-based methods and Rabitz et al. (1983) for local methods. Chan et al. (2000b) provide an overview of software for sensitivity analysis. In this section we present a brief overview of methods based on the characterization of a typical sensitivity analysis shown in Figure 3.1. Here the structure of sensitivity analysis is characterized as a process consisting of four steps.

In the first step the computation is identified for which the sensitivity analysis is to be performed. This means that one has to describe the output factors, i.e. those results of the computation in which one is interested, and their relation to the input factors. The input factors may be simple assumptions on the numerical

value of an input variable, but can also be of a more structural nature, such as an assumption regarding the validity of a particular probability distribution. Two questions of particular importance at this stage are (i) whether the input-output relation is differentiable, and (ii) whether it is possible to compute this derivative. The answers determine the feasibility of the use of the derivative-based methods described in this chapter. For cases that computations are expensive and the number of input variables is large, Morris (1991) proposes an approach to perform a preliminary screening to identify influential variables.

Next, the uncertainty of the input factors and their interactions must be assessed. Knowledge on the uncertainty with respect to the quantitative values of the input variables is usually represented by means of a (joint) probability distribution. The characterization of subjective uncertainty has been widely studied, see for instance Helton and Davis (2000), Cooke (1991) and Meyer and Booker (1991). Sometimes parameterization can be used to assign a probability distribution to a range of assumptions. At this stage one usually also decides to investigate either an entire space of input factors, or to focus on a local region around one particular reference value of the input factors. If one is interested in establishing the reliability of a particular computational result, it is generally sufficient to perform a local analysis. If the sensitivity analysis is performed to support, for instance, the choice of an experimental design, a more global analysis may be more appropriate.

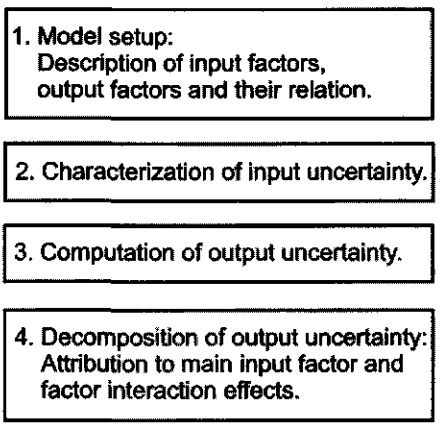



Figure 3.1: Characterization of the structure of a typical sensitivity analysis.

The next step is to map the uncertainty information on the input factors into uncertainty information on the output factors. A straightforward approach to this end is by means of sampling-based methods, see for instance Helton and Davis (2000). For each element of a sample the associated output is computed, and the combined outcomes for all elements are used to approximate the output uncertainty. This output uncertainty can be represented by, for instance, histograms, by estimated cumulative distribution functions or by means of box plots. These methods are computationally expensive, even though considerable improvements in efficiency can be achieved by using special sampling techniques such as Latin hypercube sampling (McKay et al., 1979). This chapter will mainly focus on methods based on the use of derivative information for the propagation of properties of an input probability distribution. For computations on large models it may also be efficient to first approximate the model by means of a so-called response surface and then perform the sensitivity analysis using this approximation, see Box and Draper (1987), and see Isukapalli et al. (2000) for the use of automatic

differentiation for this methodology. A review of sensitivity methods that exploit the special structure of systems of differential equations can be found in Rabitz et al. (1983).

In the last step, which is often termed sensitivity analysis in its own right, the output variation is decomposed into terms that correspond to the input sources of uncertainty. For the sampling-based methods the effect of various sources of input uncertainty can be investigated by the examination of simple scatterplots or by the computation of rank regression correlation coefficients, see Helton and Davis (2000). Further, there is a large variety of methods based on the decomposition of output variance, see Chan et al. (2000a) for an overview. These include the computation of Sobol' sensitivity indices, which may alternatively be computed by means of the FAST (Fourier Amplitude Sensitivity Test) method. Archer et al. (1997) have shown that the first order versions of these methods, which treat the input variables as deterministic, are equivalent to a decomposition of variance in which the input variables are interpreted as independent random variables. In this chapter we will also discuss a local method for the decomposition of variance by means of derivative information and compare the results to those obtained by a global method.

## 3.3 The State Forest Account Method

To illustrate the methods discussed in this chapter we consider a computation for establishing the carbon mass content of phytomass. The computation is used by the Forestry Project at IIASA as part of the assessment of the contribution of Russian forests to the global carbon cycle. For this purpose a database was developed with detailed figures on forest growing stock volume. With the aim of using additional information concerning the quality of the forest sites and their relative stocking density, the total area of Russia was divided into so-called ecoregions. See Shepashenko et al. (1998) for the basic principles by which these ecoregions are defined. For each ecoregion the dominant tree species are identified and for each species the database contains the growing stock volumes by age class. Most data are obtained from the Russian State Forest Account, which is assembled from forest inventories by forest management enterprises, natural reserves and national parks.

This database, and a number of other databases, on for example agricultural land-use and carbon emissions of the energy sector, together constitute the data sources for the IIASA study on the full carbon account of Russia (Nilsson et al., 2000). This study is of particular interest in its relation to the carbon accounting system that is proposed to be used for the Kyoto protocol. To make target reductions more feasible and cost-effective the protocol allows the industrialized countries to compensate fossil fuel emissions by deducting savings from the creation of biological sinks for carbon. These may include planting of new forests, reforesting clear-cut areas and cutting down unhealthy forests. Since the interaction between atmosphere and terrestrial biosphere is still poorly understood

(Schimel (1995), Goudriaan (1992)), however, the effects of this type of measure can currently only be assessed by performing a full carbon account, see Shvidenko and Nilsson (1998). This study shows that the reductions possible by the activities prescribed by the protocol fall well within the uncertainty range of the total flux balance. It is thus concluded that the effectiveness of land-use change measures and the effectiveness of emission reductions cannot currently be compared in a meaningful manner. This also means that countries are not able to verify their Kyoto targets at the country level.

In this chapter we do not deal directly with uncertainty questions regarding the full carbon account, but consider the sensitivity analysis of one constituent method of the full assessment procedure. This method transforms a database of forest growing stock volumes into a number of so-called phytomass (living biomass) fractions. The mass fractions distinguished are: wood of stems, wood of branches, foliage (leaves and needles), roots, understory (undergrowth and bushes) and ground vegetation. Both the distribution of the total phytomass over the mass fractions and the mass density associated with a fraction depend on the age of the stand, the quality of the forested area (measured by a site-index) and the relative stocking density of the stand. Next, the phytomass fractions are transformed into carbon content by a simple conversion factor. We now give a more detailed description of this computation, which we refer to as the State Forest Account (SFA) Method.

| | | | | Age classes | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Ecoregion | FIT | DS | Variables | $A_1$ | $A_2$ | ... | $A_q$ | $SI$ | $RS$ |
| $m$ | $r_m$ | $\rho_1$ | $S$ | $S_{11}$ | $S_{12}$ | ... | $S_{1q}$ | | |
| | | | $GS$ | $GS_{11}$ | $GS_{12}$ | ... | $GS_{1q}$ | $SI_{\rho_1}$ | $RS_{\rho_1}$ |
| | | $\rho_2$ | $S$ | $S_{21}$ | $S_{22}$ | ... | $S_{2q}$ | | |
| | | | $GS$ | $GS_{21}$ | $GS_{22}$ | ... | $GS_{2q}$ | $SI_{\rho_2}$ | $RS_{\rho_2}$ |

Table 3.1: Growing stock volume database entry for ecoregion $m$.

The structure of a database entry for one ecoregion is shown in Table 3.1. For each ecoregion $m$, dominant species $\rho$ and age class $A$, the area covered $S$ and corresponding growing stock volume $GS$ are listed. The forest inventory type (FIT) variable indicates the method by which the growing stock volume was obtained. Each dominant species has an associated average site-index $SI$ and average relative stocking $RS$ for the particular ecoregion. The site-index is proportional to the average height of a stand at a certain age, and is as such an indicator of site quality. The relative stocking is defined as the ratio between the basal area of a stand and the basal area of a fully stocked stand.

The phytomass fraction $M_{\text{fr}}$ (Tg) is related to the growing stock volume by

$$M_{\text{fr}} = f_{\text{fr}} \cdot GS, \tag{3.1}$$

where $GS$ is the (green) growing stock in $\text{hm}^3$ and $\text{f}_{\text{fr}}$ is the relative density ($\text{Tg}/\text{hm}^3$) of the corresponding phytomass fraction.

The relative density $f_{\text{fr}}$ depends on age $A$, site-index $SI$ and relative stocking $RS$. In Lakida et al. (1997) and Shepashenko et al. (1998) it is shown that $f_{\text{fr}}$ can be approximated using the following expression

$$f_{\text{fr}} = c_0 SI^{c_1} A^{(c_2 + c_3 RS + c_4 RS^2)}, \tag{3.2}$$

where the coefficients $c_0$ through $c_4$ are determined by nonlinear regression. See Lakida et al. (1997) for the details of the estimation process. The regression coefficients are determined for test plots with constant $SI$ and $RS$ values. For the entire ecoregion these values are of course not constant. The uncertainty caused by this 'upscaling' effect will be investigated in section 3.4.6.

As an example of an application of the State Forest Account Method we take pine (*Pinus sylvestris*), with characteristics of the SFA mixed and deciduous forest type, as our species of consideration. We compute the carbon contributions of two fractions, viz. of the stem and needles fractions. The phytomass of the two fractions is described by

$$M_s(A) = f_s(A, SI, RS)\, GS(A), \tag{3.3}$$
$$M_n(A) = f_n(A, SI, RS)\, GS(A), \tag{3.4}$$

where $M_s$ and $M_n$ are the stem and needles phytomass, and $f_s$ and $f_n$ are the stem and needles phytomass densities, respectively.

The regression coefficients that have been determined for the pine species are listed in Table 3.2 (taken from Shepashenko et al. (1998)).

| Phytomass fraction | $c_0$ | $c_1$ | $c_2$ | $c_3$ | $c_4$ |
|---|---|---|---|---|---|
| Stem | 0.3172 | 0.0445 | 0.1338 | -0.1824 | 0.0851 |
| Needles | 0.4146 | 0.7097 | -0.8140 | 0.1107 | -0.1257 |

Table 3.2: Regression coefficients in equation (3.2) for pine (mixed and deciduous forests).

The carbon content $C$ (Tg) of the two mass fractions now follows from

$$C = \sum_{A=1}^{q} (\delta_s M_s(A) + \delta_n M_n(A)), \tag{3.5}$$

where $\delta_s$ and $\delta_n$ are the dry mass to carbon mass conversion factors for the stem and needle fractions, respectively; $q$ is the number of age classes.

The uncertainty of the carbon content depends both on the accuracy of the data in the database and the validity of the structure of the computation. In the following we will investigate the sensitivity of the computation output with respect to the input data, and consider the decomposition of the output uncertainty with respect to various sources of input uncertainty. Special attention is also paid to the uncertainty caused by the upscaling, mentioned earlier, from test plots to ecoregions.

## 3.4 Propagation of uncertainty by means of derivative information

### 3.4.1 Propagation of perturbations

Consider a system response $R(x)$ with $n$ input variables $x = (x_1, \ldots, x_n)'$. The sensitivity coefficients of the response at input $x$ are defined by

$$s_i^1(x) = \frac{\partial R}{\partial x_i}(x), \quad i = 1, \ldots, n. \tag{3.6}$$

The coefficient $s_i^1(x)$ is a linear estimate of the number of units change in $R$ caused by a unit change in $x_i$. The sensitivity results can be made independent of the units of the response and input variables by using normalized sensitivity coefficients

$$\bar{s}_i^1(x) = \frac{\partial R}{\partial x_i}(x)\frac{x_i}{R}, \quad i = 1, \ldots, n. \tag{3.7}$$

Note that in a linear approximation we have

$$\bar{s}_i^1(x)\frac{\delta x_i}{x_i} = \frac{1}{R}\frac{\partial R}{\partial x_i}(x)\delta x_i = \frac{\delta R}{R}, \quad i = 1, \ldots, n. \tag{3.8}$$

It follows that the coefficient $\bar{s}_i^1(x)$ represents a linear estimate of the percentage change in $R$ as a result of a one percent change in $x_i$. Analogously, the sensitivity can be scaled by using the standard deviations of input and output variables, giving

$$\tilde{s}_i^1(x) = \frac{\partial R}{\partial x_i}(x)\frac{\mathrm{sd}(x_i)}{\mathrm{sd}(R)}, \quad i = 1, \ldots, n, \tag{3.9}$$

in which case the coefficient is a linear estimate of the percentage change in $R$ relative to its standard deviation resulting from a change in $x_i$ of one percent of its standard deviation.

Second order sensitivity coefficients are defined by

$$s_{ij}^2(x) = \frac{\partial^2 R}{\partial x_i \partial x_j}(x), \tag{3.10}$$

and,

$$\bar{s}_{ij}^2(x) = \frac{\partial^2 R}{\partial x_i \partial x_j}(x)\frac{x_i x_j}{R}, \quad \text{and} \quad \tilde{s}_{ij}^2(x) = \frac{\partial^2 R}{\partial x_i \partial x_j}(x)\frac{\mathrm{sd}(x_i)\mathrm{sd}(x_j)}{\mathrm{sd}(R)}, \tag{3.11}$$

for $i, j = 1, \ldots, n$. Sensitivity coefficients of order higher than two are defined analogously.

If the independent variables $x$ are perturbed by a vector $\delta x$, the sensitivity coefficients can be used to approximate the resulting change $\delta R$ in the response. We have, using a Taylor expansion of order two, that

$$\delta R = R(x + \delta x) - R(x) = \sum_{i=1}^{n} s_i^1(x)\delta x_i + \frac{1}{2}\sum_{i,j=1}^{n} s_{ij}^2(x)\delta x_i \delta x_j + \mathcal{O}(|\delta x|^3). \tag{3.12}$$

Using normalized sensitivity coefficients we obtain an expression for the relative output perturbation in terms of the relative input perturbations

$$\frac{\delta R}{R} = \sum_{i=1}^{n} \bar{s}_i^1(x) \left(\frac{\delta x_i}{x_i}\right) + \frac{1}{2} \sum_{i,j=1}^{n} \bar{s}_{ij}^2(x) \left(\frac{\delta x_i}{x_i}\right) \left(\frac{\delta x_j}{x_j}\right) + \mathcal{O}(|\delta x|^3). \qquad (3.13)$$

**The SFA Method example**

As an example of perturbation propagation, we consider computation of the carbon content contribution of the pine species for one ecoregion, described in section 3.3. Two computations are considered: for the first computation we assume $SI = 2$ and $RS = 0.3$; for the second computation we assume $SI = 4$ and $RS = 0.7$. We further take for both cases $\delta_s = 0.5$, $\delta_n = 0.45$ and $GS(A) = 1.5 \cdot 10^7 \mathrm{m}^3$ for $A = 1, \ldots, 10$. The resulting carbon content contributions are then 44 Tg C for case 1 and 54 Tg C for case 2. The corresponding normalized sensitivity coefficients are shown in Figure 3.2. The sensitivity coefficient shown for $c$ is the maximum

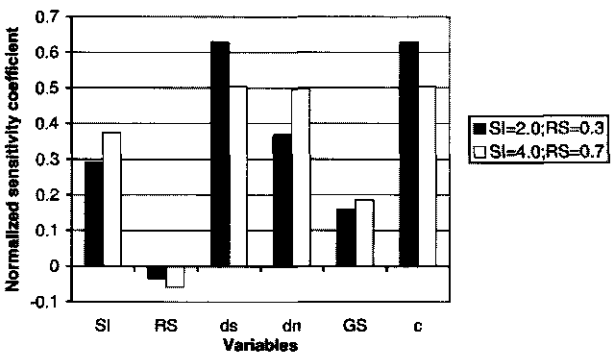



Figure 3.2: Normalized sensitivity coefficients for the SFA Method example. Meaning of the variable labels: ds = $\delta_s$, dn = $\delta_n$, GS: maximum value for the growing stock variables, and c: maximum value for the regression coefficients.

value of the coefficients corresponding to the regression coefficients $c_0$ through $c_4$, for the two phytomass fractions. The coefficient corresponding to $GS$ is the maximimum sensitivity coefficient value of the 10 age class sensitivity coefficients. If all standard deviations are taken to be a fixed fraction of the variable values, the absolute value of the sensitivity coefficients scaled by means of standard deviations are also a fixed fraction of the normalized sensitivity coefficients. Notice though that for variables with negative values these coefficients will be opposite in sign.

### 3.4.2 Propagation of expectation and variance-covariance

We now discuss propagation of uncertainty in the input variables into the expectation and variance-covariance of a response function by means of local derivative information. In Atherton et al. (1975) a linear approximation is used to propagate input variables with a normal distribution through systems of ordinary differential

equations. Higher-order variance-covariance propagation for normal input distributions is discussed in Ronen (1988).

Consider a system response $y = R(x)$ depending on $n$ independent variables $x = (x_1, \ldots, x_n)$. We assume that the input variables are random variables with a joint distribution for which the expectation and variance-covariance exist. The expectation and variance-covariance of the output variables are approximated by means of Taylor series expansions of these quantities. The expansion of the response function around the expected value of the input variables is given by

$$\begin{aligned} R(x) &= R(E[x]) + \sum_{i_1=1}^{n} \left(\frac{\partial R}{\partial x_{i_1}}\right)_{E[x]} \delta x_{i_1} + \frac{1}{2} \sum_{i_1,i_2=1}^{n} \left(\frac{\partial^2 R}{\partial x_{i_1} \partial x_{i_2}}\right)_{E[x]} \delta x_{i_1} \delta x_{i_2} \\ &+ \frac{1}{3!} \sum_{i_1,i_2,i_3=1}^{n} \left(\frac{\partial^3 R}{\partial x_{i_1} \partial x_{i_2} \partial x_{i_3}}\right)_{E[x]} \delta x_{i_1} \delta x_{i_2} \delta x_{i_3} + \ldots \\ &+ \frac{1}{N!} \sum_{i_1,\ldots,i_N=1}^{n} \left(\frac{\partial^N R}{\partial x_{i_1} \ldots \partial x_{i_N}}\right)_{E[x]} \delta x_{i_1} \delta x_{i_2} \ldots \delta x_{i_N} + \ldots, \end{aligned} \tag{3.14}$$

in which $\delta x_i = x_i - E[x_i]$.

For the expectation of the response function we then find, since $E[\delta x_i] = 0$,

$$\begin{aligned} E[R(x)] &= R(E[x]) + \frac{1}{2} \sum_{i_1,i_2=1}^{n} \left(\frac{\partial^2 R}{\partial x_{i_1} \partial x_{i_2}}\right)_{E[x]} E[\delta x_{i_1} \delta x_{i_2}] + \\ &\frac{1}{3!} \sum_{i_1,i_2,i_3=1}^{n} \left(\frac{\partial^3 R}{\partial x_{i_1} \partial x_{i_2} \partial x_{i_3}}\right)_{E[x]} E[\delta x_{i_1} \delta x_{i_2} \delta x_{i_3}] + \ldots + \\ &\frac{1}{N!} \sum_{i_1,\ldots,i_N=1}^{n} \left(\frac{\partial^N R}{\partial x_{i_1} \ldots \partial x_{i_N}}\right)_{E[x]} E[\delta x_{i_1} \ldots \delta x_{i_N}] + \ldots. \end{aligned} \tag{3.15}$$

It follows that the expectation of the response $E[R(x)]$ is equal to the sum of the response of the expectation of the input $R(E[x])$ and of additional higher-order correction terms. The correction terms are products of higher-order response derivatives and higher-order moments of the input distribution.

For the propagation of the input variance-covariance we first consider a scalar response function. We construct an $N$th order approximation of the response variance

$$\operatorname{var}(R(x)) = E[(R(x) - E[R(x)])^2]. \tag{3.16}$$

Since we have

$$\begin{aligned} &E[(\delta x_{i_1} \ldots \delta x_{i_k} - E[\delta x_{i_1} \ldots x_{i_k}])(\delta x_{j_1} \ldots \delta x_{j_l} - E[\delta x_{j_1} \ldots x_{j_l}])] = \\ &\quad (E[\delta x_{i_1} \ldots \delta x_{i_k} \delta x_{j_1} \ldots \delta x_{j_l}] - E[\delta x_{i_1} \ldots \delta x_{i_k}] E[\delta x_{j_1} \ldots \delta x_{j_l}]), \end{aligned} \tag{3.17}$$

it follows using (3.14), (3.15) and (3.16) that the $N$th order approximation of the response variance $\mathrm{var}_N(R)$ is given by

$$\mathrm{var}_N(R) = \sum_{k,l=1}^{N} \frac{1}{k!\,l!} \sum_{\substack{i_1,\ldots,i_k=1 \\ j_1,\ldots,j_l=1}}^{n} \left( \frac{\partial^k R}{\partial x_{i_1} \ldots \partial x_{i_k}} \frac{\partial^l R}{\partial x_{j_1} \ldots \partial x_{j_l}} \right)_{E[x]}$$

$$(E[\delta x_{i_1} \ldots \delta x_{i_k} \delta x_{j_1} \ldots \delta x_{j_l}] - E[\delta x_{i_1} \ldots \delta x_{i_k}] E[\delta x_{j_1} \ldots \delta x_{j_l}]). \quad (3.18)$$

Notice that some terms in this expression are zero, since $E[\delta x] = 0$. For $N = 1$ we have

$$\mathrm{var}_1(R(x)) = \sum_{i,j=1}^{n} \left( \frac{\partial R}{\partial x_i} \frac{\partial R}{\partial x_j} \right)_{E[x]} E[\delta x_i \delta x_j]. \quad (3.19)$$

By noting that the $(\frac{\partial R}{\partial x_i})_{E[x]}$ are the first order sensitivity coefficients $s_i(E[x])$ and that $E[\delta x_i \delta x_j]$ is the covariance $\mathrm{cov}(x_i, x_j)$ of input variables $x_i$ and $x_j$, we get

$$\mathrm{var}_1(R(x)) = \sum_{i,j=1}^{n} s_i^1(E[x])\, s_j^1(E[x])\, \mathrm{cov}(x_i, x_j) = s\, V\, s', \quad (3.20)$$

where $s = (s_1^1(E[x]), \ldots, s_n^1(E[x]))$ is the sensitivity (row)vector and $V$ is the variance-covariance matrix of the input variables. This formula is known as the *sandwich rule*. The expression depends on the input variable distribution only through its expectation and variance-covariance.

The second order approximation ($N = 2$) of the response variance (3.18) is given by

$$\mathrm{var}_2(R(x)) = \sum_{i,j=1}^{n} \left( \frac{\partial R}{\partial x_i} \frac{\partial R}{\partial x_j} \right)_{E[x]} E[\delta x_i \delta x_j]$$

$$+ \sum_{\substack{i_1,i_2=1 \\ j=1}}^{n} \left( \frac{\partial^2 R}{\partial x_{i_1} \partial x_{i_2}} \frac{\partial R}{\partial x_j} \right)_{E[x]} E[\delta x_{i_1} \delta x_{i_2} \delta x_j]$$

$$+ \frac{1}{4} \sum_{\substack{i_1,i_2=1 \\ j_1,j_2=1}}^{n} \left( \frac{\partial^2 R}{\partial x_{i_1} \partial x_{i_2}} \frac{\partial^2 R}{\partial x_{j_1} \partial x_{j_2}} \right)_{E[x]}$$

$$(E[\delta x_{i_1} \delta x_{i_2} \delta x_{j_1} \delta x_{j_2}] - E[\delta x_{i_1} \delta x_{i_2}] E[\delta x_{j_1} \delta x_{j_2}]). \quad (3.21)$$

Notice that for the second order approximation third and fourth order central moments are required. We will show further on that for normally distributed input variables the higher-order moments can be expressed in terms of the variance-covariance.

Now consider the case that the response $R(x)$ is multivariate:

$$R(x) = (R_1(x), \ldots, R_m(x))'. \quad (3.22)$$

The expressions for the approximations of the variance remain valid for each component individually. What remains to be investigated is the covariance of two components, $R_p(x)$ and $R_q(x)$ say,

$$\operatorname{cov}(R_p(x), R_q(x)) = E[(R_p(x) - E[R_p(x)])(R_q(x) - E[R_q(x)])]. \tag{3.23}$$

Using the same approach as for the variance of a scalar response, we obtain for the $N$th order approximation

$$\operatorname{cov}_N(R_p(x), R_q(x)) = \sum_{k,l=1}^{N} \frac{1}{k!\,l!} \sum_{\substack{i_1,\ldots,i_k=1 \\ j_1,\ldots,j_l=1}}^{n} \left( \frac{\partial^k R_p}{\partial x_{i_1} \ldots \partial x_{i_k}} \frac{\partial^l R_q}{\partial x_{j_1} \ldots \partial x_{j_l}} \right)_{E[x]}$$

$$(E[\delta x_{i_1} \ldots \delta x_{i_k} \delta x_{j_1} \ldots \delta x_{j_l}] - E[\delta x_{i_1} \ldots \delta x_{i_k}] E[\delta x_{j_1} \ldots \delta x_{j_l}]). \tag{3.24}$$

For $N = 1$ we have

$$\operatorname{cov}_1(R_p(x), R_q(x)) = \sum_{i,j=1}^{n} s^1_{pi}(E[x])\, s^1_{qj}(E[x]) \operatorname{cov}(x_i, x_j), \tag{3.25}$$

where $s^1_{pi}$ denotes the sensitivity coefficient of response component $p$ with respect to input variable $x_i$, for $p = 1, \ldots, m$ and $i = 1, \ldots, n$. It follows that the covariance matrix of the response is given by

$$\operatorname{cov}_1(R) = SVS', \tag{3.26}$$

where $S$ is the derivative of the response function with respect to the input variables at the input expectation. The second order approximation of the covariance of two response components is given by

$$\begin{aligned}
\operatorname{cov}_2(R_p(x), R_q(x)) = {} & \sum_{i,j=1}^{n} \left( \frac{\partial R_p}{\partial x_i} \frac{\partial R_q}{\partial x_j} \right)_{E[x]} E[\delta x_i \delta x_j] + \\
& \frac{1}{2} \sum_{\substack{i_1,i_2=1 \\ j=1}}^{n} \left( \frac{\partial^2 R_p}{\partial x_{i_1} \partial x_{i_2}} \frac{\partial R_q}{\partial x_j} \right)_{E[x]} E[\delta x_{i_1} \delta x_{i_2} \delta x_j] + \\
& \frac{1}{2} \sum_{\substack{i=1 \\ j_1,j_2=1}}^{n} \left( \frac{\partial R_p}{\partial x_i} \frac{\partial^2 R_q}{\partial x_{j_1} \partial x_{j_2}} \right)_{E[x]} E[\delta x_i \delta x_{j_1} \delta x_{j_2}] + \\
& \frac{1}{4} \sum_{\substack{i_1,i_2=1 \\ j_1,j_2=1}}^{n} \left( \frac{\partial^2 R}{\partial x_{i_1} \partial x_{i_2}} \frac{\partial^2 R}{\partial x_{j_1} \partial x_{j_2}} \right)_{E[x]} \\
& (E[\delta x_{i_1} \delta x_{i_2} \delta x_{j_1} \delta x_{j_2}] - E[\delta x_{i_1} \delta x_{i_2}] E[\delta x_{j_1} \delta x_{j_2}]).
\end{aligned} \tag{3.27}$$

As in (3.21) this second-order expressions depend on the third and fourth order moments of the input variable distributions. We now further reduce these expressions for the uniform and multivariate normal distributions.

### Uniformly distributed input variables

We first consider the special case of independent input variables which are uniformly distributed. The density function of variable $x_i$ has a constant value $1/(2r_i)$ on the interval $[E[x_i] - r_i, E[x_i] + r_i]$, and is zero outside this interval. Since the uniform density function is symmetric around its mean, the odd central moments are zero.

The $N$th order central moments for $N$ even, are given by

$$E[(x_i - E[x_i])^N] = \frac{r_i^N}{N+1} = \frac{3^{N/2}}{N+1}\mathrm{var}(x_i)^{N/2}. \tag{3.28}$$

By substitution in (3.21) and (3.27) we find

$$\begin{aligned}\mathrm{var}_2(R(x)) = &\sum_{i=1}^{n}\left((s_i^1)^2\mathrm{var}(x_i) + \frac{1}{5}(s_{ii}^2)^2\mathrm{var}(x_i)^2\right)\\ &+ \frac{1}{4}\sum_{\substack{i,j=1\\ i\neq j}}^{n}(s_{ij}^2)^2\mathrm{var}(x_i)\mathrm{var}(x_j),\end{aligned} \tag{3.29}$$

and

$$\begin{aligned}\mathrm{cov}_2(R_p(x), R_q(x)) = &\sum_{i=1}^{n}\left(s_{pi}^1 s_{qi}^1\mathrm{var}(x_i) + \frac{1}{5}s_{pii}^2 s_{qii}^2\mathrm{var}(x_i)^2\right)\\ &+ \frac{1}{4}\sum_{\substack{i,j=1\\ i\neq j}}^{n}s_{pij}^2 s_{qij}^2\mathrm{var}(x_i)\mathrm{var}(x_j),\end{aligned} \tag{3.30}$$

where the first and second order sensitivity coefficients are evaluated in the mean values of the input variables.

### Multivariate normally distributed input variables

For this case the joint density function of the input variables is given by

$$f(x) = \frac{1}{\sqrt{(2\pi)^n|V|}}\exp(-\frac{1}{2}(x-\mu)'V^{-1}(x-\mu)), \tag{3.31}$$

where $\mu = E[x]$ is the expectation vector of the independent variables and $V$ the variance-covariance matrix.

The higher-order central moments of a multivariate normal distribution can be expressed in terms of the second order moments, i.e. in terms of the variance-covariance of the input variables. Since the multivariate normal distribution is symmetric about its mean, the odd central moments are zero. Using the characteristic function of the multivariate normal distribution with zero mean,

$$\phi(t) = \exp(-\frac{1}{2}t'Vt), \tag{3.32}$$

we can compute the even central moments using

$$i^N E[\delta x_{i_1} \dots \delta x_{i_N}] = \frac{\partial^N \phi}{\partial t_{i_1} \dots \partial t_{i_N}}. \tag{3.33}$$

For $N = 4$ we then find

$$E[\delta x_{i_1} \dots \delta x_{i_4}] = V_{i_1 i_2} V_{i_3 i_4} + V_{i_1 i_3} V_{i_2 i_4} + V_{i_1 i_4} V_{i_2 i_3}. \tag{3.34}$$

Moments of order higher than four take similar expressions. The $N$th order moment consists of a sum of covariance products such that each term in the product consists of the covariance of two input variables corresponding to a pair taken from the index symbol set $\{i_1, \dots, i_N\}$. For example $V_{i_1 i_2}$ corresponds to the pair $(i_1, i_2)$. If all pairs corresponding to a product of covariances are disjoint and each element of the index symbol set is used exactly once in those pairs, then the product is said to correspond to a partition of the index symbol set in disjoint pairs. The sum leading to the $N$th order moment is taken such that on the one hand each product of covariances in the sum corresponds to a partition of the index symbol set in disjoint pairs, and on the other hand each partition of the index symbol set in disjoint pairs corresponds to exactly one product of covariances in the sum.

Under the multivariate normal assumption and using (3.34), the second order approximation of the variance of a response variable given by (3.21) then reduces to

$$\begin{aligned} \mathrm{var}_2(R(x)) &= \sum_{i,j=1}^{n} s_i^1 s_j^1 \mathrm{cov}(x_i, x_j) + \frac{1}{4} \sum_{i,j,k,l=1}^{n} s_{ij}^2 s_{kl}^2 \\ &\quad (\mathrm{cov}(x_i, x_k)\mathrm{cov}(x_j, x_l) + \mathrm{cov}(x_i, x_l)\mathrm{cov}(x_j, x_k)), \end{aligned} \tag{3.35}$$

where the first and second order sensitivity coefficients are evaluated at the input expectation $E[x]$. Analogously, the second order approximation of the covariance given by (3.27) reduces to

$$\begin{aligned} \mathrm{cov}_2(R_p(x), R_q(x)) &= \sum_{i,j=1}^{n} s_{pi}^1 s_{qj}^1 \mathrm{cov}(x_i, x_j) + \frac{1}{4} \sum_{i,j,k,l=1}^{n} s_{pij}^2 s_{qkl}^2 \\ &\quad (\mathrm{cov}(x_i, x_k)\mathrm{cov}(x_j, x_l) + \mathrm{cov}(x_i, x_l)\mathrm{cov}(x_j, x_k)). \end{aligned} \tag{3.36}$$

### 3.4.3 Validation of local approximations

Sampling-based methods are global in nature, i.e. they do not depend on local approximations of the response function. These methods are thus suitable for cases where the possible range of input variables is so large that the local approximations to the response function are not sufficiently accurate. Since, however, the sampling-based methods require a response function evaluation for every element in the sample, they are computationally expensive. Local sensitivity methods based on derivative information require a single function evaluation at a particular point of interest, and the evaluation of first and possibly higher-order derivatives at this point. As discussed in Chapter 2, the derivatives can be obtained efficiently by means of automatic differentiation. Especially for response functions with large associated computation times, a local derivative method leads to a substantial reduction in the compution time that is required to obtain the sensitivity information.

To evaluate the validity of the local approach, one must investigate the difference between the response function and its local approximation. As we have

$$\begin{aligned} \mathrm{cov}(R_p(x), R_q(x)) &= E[(R_p(x) - E[R_p(x)])(R_q(x) - E[R_q(x)])] \\ &= \int f_x(x)(R_p(x) - E[R_p(x)])(R_q(x) - E[R_q(x)])\, dx, \quad (3.37) \end{aligned}$$

where $f_x(x)$ denotes the probability density function of the input variables, the validity of the variance-covariance approximation relies on the assumption that the contribution corresponding to regions where the local response approximation is not accurate is small relative to the total variance-covariance. If there is reason to doubt this assumption, it is recommended to use global sensitivity analysis methods. This is the case if the input probability density function is relatively large in input variable regions corresponding to poor approximations, or also, if the response functions show large deviations from their means in such regions.

Typically, the response function is approximated well around its mean, and the Taylor expansion is less accurate in the tails of the input variable distributions. If the tails of the input variable distribution play an important role, as is often the case in risk analysis, sampling methods based on, for instance, Latin hypercube and importance sampling are recommended, see Helton and Davis (2000).

### 3.4.4 Decomposition of variance-covariance

Using the structure of the variance-covariance expressions we can investigate the contribution of each of the input variables to the total variance. This is particularly straightforward for the linear variance-covariance approximations given by (3.20) and (3.25). Each term in these expressions can be directly assigned to either an individual input variable, or the correlation of two input variables. It is also possible to construct a decomposition in which each term is assigned to exactly one of the input variables, both for the first order approximation and second

order approximation. This will be discussed further in the context of precision sensitivity coefficients in section 3.5. In section 3.4.5 the decompositions based on local approximations will be compared to decompositions based on sampling methods. The following results will be useful in that discussion.

The random variable $y = R(x)$ depends on the input random variables $x_i, i = 1, \ldots, n$. For each of the $x_i$ variables the variance of $y$ may be expressed as a sum of two terms by

$$\text{var}(y) = \text{var}_{x_i}(E[y|x_i]) + E_{x_i}[\text{var}(y|x_i)], \quad i = 1, \ldots, n. \tag{3.38}$$

The first component is called the variance of the conditional expectation (VCE), the second component is called the residual part. If $E[y|x_i]$ mimics $y$ well, i.e. if the VCE is close to the variance of $y$, then the corresponding input variable is influential. Also notice that if $y$ and $x_i$ are independent, the VCE is equal to zero. If $y$ depends on $x_i$ deterministically then the VCE is equal to $\text{var}(y)$. McKay (1995) used the decomposition of (3.38) to define the correlation ratio $\eta_i$ to measure the importance of $x_i$ by

$$\eta_i^2 = \frac{\text{var}_{x_i}(E[y|x_i])}{\text{var}(y)}. \tag{3.39}$$

Notice that (3.38) is general in the sense that it may also be used to define importance measures for *a set* of input variables.

If the input variables are independent Cox (1982) proves an extension of (3.38), giving the following decomposition of the output variance

$$\text{var}(y) = \sum_{i=1}^{n} V_i + \sum_{i<j}^{n} V_{ij} + \sum_{i<j<k}^{n} V_{ijk} + \ldots + V_{12\ldots n}, \tag{3.40}$$

where

$$V_{ijk\ldots} = \text{var}(Z_{ijk\ldots}), 1 \leq i < j < k < \ldots \leq n, \tag{3.41}$$

$$Z_i = E[y|x_i], \quad i = 1, \ldots, n, \tag{3.42}$$

$$Z_{ij} = E[(y - \sum_{p=1}^{n} Z_p)|x_i, x_j], 1 \leq i < j \leq n, \tag{3.43}$$

$$Z_{ijk} = E[(y - \sum_{p=1}^{n} Z_p - \sum_{p<q}^{n} Z_{pq})|x_i, x_j, x_k], 1 \leq i < j < k \leq n, \tag{3.44}$$

with higher-order $Z$ variables defined analogously. The variance is a sum of the VCE's of the input variables, and of terms representing variances corresponding to interactions between the input variables.

### 3.4.5 The SFA Method example (continued)

We again consider the computation of the carbon content contribution of the pine species for one ecoregion, described in section 3.3. We take $SI = 4.0$ and $RS = .7$, $\delta_s = 0.5$, $\delta_n = 0.45$ and $GS(a) = 1.5 \cdot 10^7 \mathrm{m}^3$ for $a = 1, \ldots, 10$ (case 2). The resulting carbon content contribution is 54 Tg C.

We first characterize the input uncertainty by assuming that the input variables are independent and normally distributed with expectation equal to the input variable value used in the computation, and standard deviation equal to 10% of the absolute value of the input value.

Using (3.20) and (3.35) we can then compute first and second order approximations of the resulting variance of the computed carbon content. The first and second order methods both give a standard deviation of 7.0 Tg, corresponding to a coefficient of variation of 13%.

The associated decomposition of variance with respect to the input variables is shown in Figure 3.3 (a). This decomposition is based on (3.20): the relative contribution of input variable $x_i$ to the total variance is equal to $s_i^2 \mathrm{var}(x_i)/\mathrm{var}(y) \cdot 100\%$.

As a first validation of these results we check the accuracy of the Taylor approximation by comparison of the response function to its approximation for each variable at 3 standard deviations from its mean. The largest resulting difference for the first order approximation is 19% (with differences much lower for most variables). For the second order approximation the largest difference is only 2%.

We further compare the resulting decomposition to a sample-based decomposition. Figure 3.3 (b) shows the decomposition obtained by using the VCE values and the corresponding correlation coefficients defined in equation (3.39) as computed by a simple random sampling method. Since in this case the input variables are independent, the variance can be decomposed according to (3.40). Most of the variance is explained by the VCE coefficients in that the higher-order interaction terms have a negligible contribution. The two decompositions of variance are nearly equal. In both cases the uncertainty in the regression coefficients is the most important cause of uncertainty in the carbon content.

Figure 3.3 (c) then shows a variance decomposition for the computation where the input variables are no longer assumed to be independent. We have assumed a correlation of 10% among both the regression coefficients, and among the growing stock variables. Notice that both types of correlation have a substantial influence on the resulting carbon content variance. In this case the standard deviation increased to 8.6 Tg corresponding to a coefficient of variation of 16%, compared to 13% for the independent case.

As an example of a propagation of a multivariate response function we computed the covariance of $M_s$ and $M_n$ for both independent and dependent input variables. For the case of independent input variables the coefficients of variation were 11 and 18% respectively, while the correlation was 0.06. For the dependent case the standard devations were 12 and 20% respectively, with a correlation of the two phytomasses of 0.3.

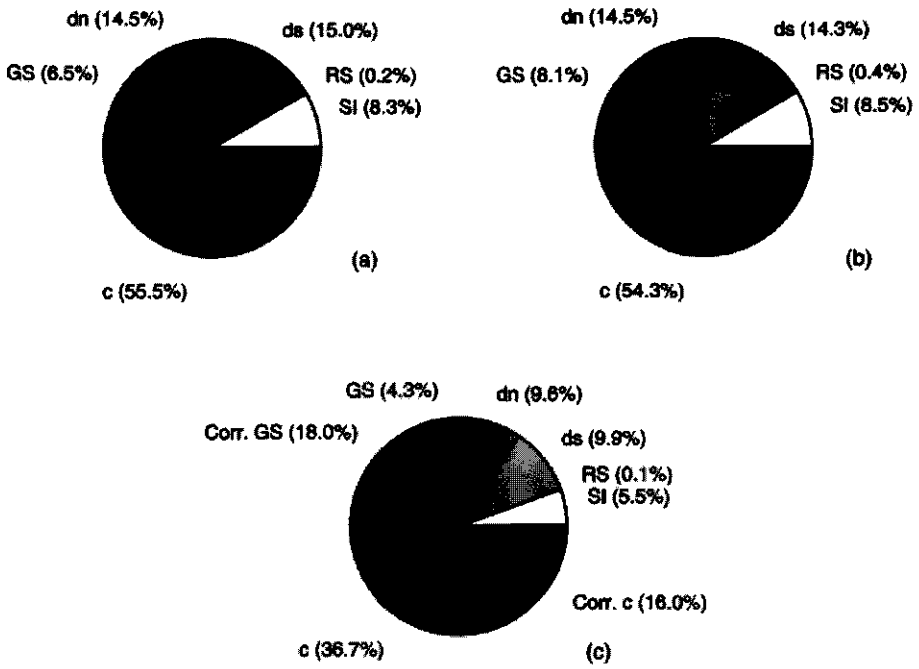



Figure 3.3: Decompositions of variance for the SFA Method example; (a) decomposition using local derivative information for independent input variables, (b) decomposition using a global decomposition method for independent input variables, and (c) decomposition using local derivative information for dependent input variables. Meaning of the variable labels: ds = $\delta_s$, dn = $\delta_n$, GS: value summed over the growing stock variables, c: value summed over the regression coefficients, Corr. c: contribution of regression coefficient correlations, and Corr. GS: contribution of growing stock variable correlations.

### 3.4.6 Uncertainty propagation and upscaling

Upscaling is the use of fitted relations in an area that is larger, and consequently less homogeneous, than the area for which the relation was fitted. The computation of the carbon content contribution of an entire ecoregion based on relations fitted on much smaller test plots is a typical example. As a final application of the uncertainty propagation methods, we show how they may be used to investigate this practice of upscaling.

Test plots with homogeneous characteristics with respect to site index and relative stocking were used to describe the relation between mass fraction density and plot characteristics, i.e. to obtain the regression coefficients $c_0$ through $c_4$ of both phytomass fractions. Subsequently, this same relation is used to compute the carbon contribution of an entire ecoregion for which there is considerable variation and uncertainty in the site index and relative stocking variables.

On the ecoregion level our knowledge and uncertainty with respect to the input variables is represented by means of a joint probability distribution. As shown, we can propagate the expectation and variance-covariance of this distribution through the computation to obtain the expectation and variance of the resulting phytomass carbon content. Here, however, we have treated the entire ecoregion as one large test plot for which the relation corresponding to the estimated regression coefficients must be valid. To investigate the sensitivity of the results with respect to this assumption, we propose the following procedure.

First assume that the ecoregion consists of $K$ subregions and that

$$C = \sum_{k=1}^{K} C_k, \tag{3.45}$$

where $C_k$ is the carbon contribution of subregion $k$. Our strategy will be to first characterize the uncertainty of the input variables at subregion level and subsequently use this information to compute the expected contribution and its variance of one subregion. Then, the expectation and variance of the total ecoregion carbon contribution follow from (3.45). The carbon contribution sensitivity as a result of the upscaling practice can then be evaluated by comparison of these estimates with estimates obtained by treating the entire ecoregion as a single test plot.

We can use the uncertainty representation for the entire ecoregion to obtain an uncertainty representation of the input variables at the subregion level. Since the most detailed information we have on the site index and relative stocking is provided by the probability distribution for these variables at the ecoregion level, we assume that also for each subregion these variables are realizations from these distributions, i.e. we assume that the $SI$ and $RS$ variables have the same distribution as on the ecoregion level. Since the regression coefficients are in fact obtained at test plot level, we can also take the same probability distribution for these variables at the subregion level as was used for the ecoregion level. To obtain a probability distribution of the growing stock volume for a subregion, we assume that the growing stock volume of the ecoregion $GS(A)$ is a sum of the growing stock volumes $GS_k(A)$ of the subregions, i.e.

$$GS(A) = \sum_{k=1}^{K} GS_k(A). \tag{3.46}$$

Since we have no further information at the subregion level we take the $GS_k(A)$ variables to have identical distributions. If we further assume that the number of plots is sufficiently large, such that the subregion growing stock volumes may be taken to be independent, then we can model the uncertainty of the growing stock volume as random variable for which $E[GS_k(A)] = E[GS(A)]/K$ and $\mathrm{var}(GS_k(A)) = \mathrm{var}(GS(A))/K$. The ecoregion level covariance values may similarly be divided by $K$ to obtain subregion covariance values. Note that the expectation of subregion growing stock volume is inversely proportional to $K$ whereas the standard deviation is inversely proportional to the square root of $K$. This means

that the subregion growing stock volumes become relatively more uncertain for an increasing number of subregions.

We have now fully characterized the input uncertainty distribution of the variables at subregion level, such that we can propagate this information to compute the expected carbon contribution of one subregion, and the associated contribution variance. Estimates of the expectation and variance of carbon content for the entire ecoregion then follow from (3.45), giving

$$E^{(K)}[C] = KE^{(K)}[C_{\text{sub}}], \quad \text{and} \quad \text{var}^{(K)}[C] = K\text{var}^{(K)}[C_{\text{sub}}], \tag{3.47}$$

where $E^{(K)}[C_{\text{sub}}]$ and $\text{var}^{(K)}[C_{\text{sub}}]$ are the expectation and variance of the carbon content obtained for a subregion, if $K$ subregions are assumed.

For the independent input variables we found that at ecoregion level the carbon content, computed by using the average values of the input variables, was 54 Tg. Using expression (3.15) we can then estimate the actual expectation of the carbon content. In this case the higher order terms are negligible (0.5 % of the response corresponding the average input values), so the expectation again equals 54 Tg. The variance equals 50 $\text{Tg}^2$, corresponding to a coefficient of variation of 13%.

We next consider the case that the ecoregion is subdivided into 10 subregions. The carbon content contribution corresponding to the average input values for one subregion was equal to 5.4 Tg. The higher order terms in (3.15) are again negligible (0.5% of the response corresponding to the average input values), so also the expectation of the carbon content equals 5.4 Tg. The variance equals 0.8 $\text{Tg}^2$, corresponding to a coefficient of variation of 16%. As expected because of the higher coefficient of variation for the growing stock variables, the results obtained for the subregion level have higher associated uncertainty than those obtained by treating the ecoregion as a single homogeneous region. Using (3.47), however, we find that for the complete ecoregion the expectation of the carbon content is equal to 54 Tg with a variance of only 8 $\text{Tg}^2$. This corresponds to a coefficient of variation of 5%. In general the coefficient of variation at the ecoregion level is equal to the coefficient of variation at the subregion level divided by the square root of $K$. We found that the coefficient of variation decreases for increasing $K$. In order to obtain a realistic impression of the carbon content uncertainty, the value of $K$ should be chosen in accordance with the true situation: the size of the subregions must be small enough to be homogeneous with respect to the variables $SI$ and $RS$, but not so small that their properties can no longer be considered independent of those of the neighboring subregions.

Since the complete ecoregion should in reality be considered as composed of smaller subregions, the obtained results are in fact expected to be more precise than indicated by the estimates obtained by treating the ecoregion as a single homogeneous region. We conclude that upscaling in this case is no reason of concern. This, of course, still presupposes the uncertainty assessment and average values of the input variables at the ecoregion level to be reasonably accurate.

## 3.5 Precision sensitivity

We now introduce a new type of sensitivity measure, which we shall refer to as *precision sensitivity*. The precision of a variable is taken to be measured by its coefficient of variation. Precision sensitivity coefficients can then be defined to measure the relative change (in percent) in output precision as a result of a one percent change[1] in the precision of an input variable. In section 3.4.2 we discussed the decomposition of output variance-covariance in terms of the influence of the uncertainty of the input variables. We will show that, for a linear approximation, the terms of such a decomposition can be interpreted as precision coefficients.

For the coefficients of variation we use the following notation

$$e_{R_p} = \mathrm{cv}(R_p) = \frac{\sigma_{R_p}}{R_p}, \quad \text{and} \quad e_{x_i} = \mathrm{cv}(x_i) = \frac{\sigma_{x_i}}{x_i}, \tag{3.48}$$

for $p = 1, \ldots, m$, $i = 1, \ldots, n$, where $\sigma_{x_i}$ is the standard deviation of $x_i$ and $\sigma_{R_p}$ is the standard deviation of $R_p$. We then define the (normalized) precision sensitivity coefficients by

$$\bar{s}_{pe_i} = \frac{\partial e_{R_p}}{\partial e_{x_i}} \frac{e_{x_i}}{e_{R_p}} = \frac{\partial \sigma_{R_p}}{\partial \sigma_{x_i}} \frac{\sigma_{x_i}}{\sigma_{R_p}}, \tag{3.49}$$

where the input variable $x_i$ is assumed to remain fixed. The derivatives are evaluated at the value of the input standard deviation (or coefficient of variation) that is taken to represent the input variable uncertainty.

A Taylor expansion may be used to evaluate the result of simultaneously changing the precision of a number of input variables

$$\frac{\delta e_{R_p}}{e_{R_p}} = \sum_{i=1}^{n} \bar{s}_{pe_i} \left( \frac{\delta e_{x_i}}{e_{x_i}} \right) + \mathcal{O}(|\delta e_x|^2). \tag{3.50}$$

As in the case of regular sensitivity coefficients these approximations can be improved by incorporating analogously defined higher-order precision sensitivity coefficients.

Taking

$$\mathrm{cov}(x_i, x_j) = \rho_{ij} \sigma_{x_i} \sigma_{x_j}, \tag{3.51}$$

where $\rho_{ij}$ is the correlation of input variables $x_i$ and $x_j$, and using (3.20), a first order approximation of $\sigma^2_{R_p}$ is given by

$$\sigma^2_{R_p} = \sum_{i,j=1}^{n} s^1_{pi} s^1_{pj} \rho_{ij} \sigma_{x_i} \sigma_{x_j}. \tag{3.52}$$

By taking the derivative with respect to the standard deviation of $x_i$, we get

$$\frac{\partial \sigma_{R_p}}{\partial \sigma_{x_i}} = \frac{1}{2\sigma_{R_p}} \frac{\partial \sigma^2_{R_p}}{\partial \sigma_{x_i}} = \frac{1}{\sigma_{R_p}} \sum_{j=1}^{n} s^1_{pi} s^1_{pj} \rho_{ij} \sigma_{x_j}. \tag{3.53}$$

[1] Both changes are relative to the original values of the precision, i.e. they are in percents, not in percent points.

For the case of normally distributed variables we can also use the second order approximation of $\sigma^2_{R_p}$ given by (3.35)

$$\sigma^2_{R_p} = \sum_{i,j=1}^{n} s^1_{pi} s^1_{pj} \rho_{ij} \sigma_{x_i} \sigma_{x_j} + \frac{1}{4} \sum_{i,j,k,l=1}^{n} s^2_{pij} s^2_{pkl} (\rho_{ik}\rho_{jl} + \rho_{il}\rho_{jk}) \sigma_{x_i} \sigma_{x_j} \sigma_{x_k} \sigma_{x_l}, \tag{3.54}$$

to obtain the derivative

$$\frac{\partial \sigma_{R_p}}{\partial \sigma_{x_i}} = \frac{1}{\sigma_{R_p}} \left( \sum_{j=1}^{n} s^1_{pi} s^1_{pj} \rho_{ij} \sigma_j + \frac{1}{2} \sum_{j,k,l=1}^{n} s^2_{pij} s^2_{pkl} (\rho_{ik}\rho_{jl} + \rho_{il}\rho_{jk}) \sigma_{x_j} \sigma_{x_k} \sigma_{x_l} \right). \tag{3.55}$$

Using (3.49) we then find for the first and second order accurate precision sensitivity coefficients

$$\bar{s}_{pe_i} = \frac{1}{\sigma^2_{R_p}} \sum_{j=1}^{n} s^1_{pi} s^1_{pj} \rho_{ij} \sigma_{x_i} \sigma_{x_j}, \tag{3.56}$$

and

$$\bar{s}_{pe_i} = \frac{1}{\sigma^2_{R_p}} \left( \sum_{j=1}^{n} s^1_{pi} s^1_{pj} \rho_{ij} \sigma_{x_i} \sigma_{x_j} + \frac{1}{2} \sum_{j,k,l=1}^{n} s^2_{pij} s^2_{pkl} (\rho_{ik}\rho_{jl} + \rho_{il}\rho_{jk}) \sigma_{x_i} \sigma_{x_j} \sigma_{x_k} \sigma_{x_l} \right), \tag{3.57}$$

respectively.

It is interesting to note that the precision sensitivity coefficient computed from a first order variance approximation is equal to the relative contribution of the associated input variable to the total variance. The first order approximation of the total variance is given by (3.52). This approximation consists of a sum of $n^2$ terms. The $n$ terms $(s^1_{pi})^2 \sigma_i^2, i = 1, \ldots, n$ may be directly assigned to their corresponding input variables. The terms $s^1_{pi} s^1_{pj} \rho_{ij} \sigma_i \sigma_j, i \neq j$ are symmetric in $i$ and $j$, such that the term $s^1_{pi} s^1_{pj} \rho_{ij} \sigma_{x_i} \sigma_{x_j}$ can be assigned to variable $x_i$ and the equal term $s^1_{pj} s^1_{pi} \rho_{ji} \sigma_{x_j} \sigma_{x_i}$ to variable $x_j$. The relative contribution of input variable $x_i$ to the total variance is then equal to

$$\frac{1}{\sigma^2_{R_p}} \sum_{j=1}^{n} s^1_{pi} s^1_{pj} \rho_{ij} \sigma_{x_i} \sigma_{x_j}, \tag{3.58}$$

which is exactly equal to the first order approximation of the precision sensivity coefficient, given in (3.56). A similar assignment, however, of the second order terms in (3.54) to the individual input variables is not equal to the second order approximation of the precision coefficient.

**The SFA Method example**
In section 3.4.5 we showed that the approximation of the total carbon contribution variance is dominated by the linear terms for both independent and dependent input variables. For the independent input variables the relative contributions to the total variance are thus equal to the precision sensitivity coefficients. It immediately follows from Figure 3.3 (a) that an increase in precision of 10 percent in for instance SI leads to an increase of precision in the total carbon contribution of 0.83 percent.

For the dependent case, the contributions of the correlation terms are divided over the precision sensitivity coefficients of the individual contributions of the regression coefficients and the growing stock variables.

## 3.6 Sensitivity analysis of implicit relations

In Rabitz et al. (1983) a so-called feature sensitivity analysis is described for the maximum of a function of a single variable, in the context of systems for chemical kinetics. Here, we present an extension of this type of sensitivity analysis for variables that are implicitly defined. The method is based on the implicit function theorem and allows for the computation of sensitivity coefficients of any order given sufficiently high differentiability.

The method is best understood by considering an example: we quantify the sensitivity of a parameter vector estimated by means of a maximum likelihood (or least-squares) method with respect to any variables that are used in the estimation process, but are not estimated themselves. Examples of such variables are the regressor variables, observation weighting coefficients and also the observations themselves. In Chapter 4 we will discuss the application of this type of sensitivity analysis in the field of statistical inference for maximum likelihood estimation.

Let $\hat{\theta}$ be a maximum likelihood or least-squares estimate, obtained by optimization of a function $S(\theta, x)$ in $\theta$ for a certain value $\tilde{x}$ of $x$, such that

$$\frac{\partial}{\partial \theta_j} S(\hat{\theta}, \tilde{x}) = 0, \quad j = 1, \ldots, p, \tag{3.59}$$

where $p$ is the size of the parameter vector. The vector $\tilde{x}$, of size $M$, is defined such that it consists of variables with respect to which the sensitivity of the parameters is to be investigated. It may consist of any variables used in the estimation process that are not parameters to be estimated (and with respect to which the function $S$ is sufficiently differentiable). In fact, one may even consider fixing a number of the parameters and computing the sensitivity of the remaining parameters with

respect to such fixed parameters. This course will, however, not be further pursued here.

If the second derivative of $S$ with respect to the parameters exists and is non-singular, then by the implicit function theorem there exists a function $\theta = g(x)$ in the neighbourhood of $\tilde{x}$ which satisfies $\hat{\theta} = g(\tilde{x})$. If $S$ is sufficiently smooth with respect to $\theta$ and $x$, we have the following relations for the derivatives of $g$ with respect to $x$ in $\tilde{x}$:

$$\sum_{i=1}^{p} \frac{\partial^2 S}{\partial\theta_i \partial\theta_j} \frac{\partial g_i}{\partial x_{k_1}} = -\frac{\partial^2 S}{\partial\theta_j \partial x_{k_1}}, \quad k_1 = 1, \ldots, M; \quad j = 1, \ldots, p, \quad (3.60)$$

and, with Leibniz' rule,

$$\sum_{i=1}^{p} \frac{\partial^2 S}{\partial\theta_i \partial\theta_j} \frac{\partial^N g_i}{\partial x_{k_1} \ldots \partial x_{k_N}} = -\Big[\frac{\partial^{(N+1)} S}{\partial\theta_j \partial x_{k_1} \ldots \partial x_{k_N}} + \sum_{l=1}^{N-1} \sum_{i=1}^{p} \Big[\binom{N-1}{l} \frac{\partial^{(l+2)} S}{\partial\theta_j \partial\theta_i \partial x_{k_1} \ldots \partial x_{k_l}} \frac{\partial^{(N-l)} g_i}{\partial x_{k_{(l+1)}} \ldots \partial x_{k_N}}\Big]\Big], \quad (3.61)$$

for $N = 2, 3, \ldots$ and $k_1, \ldots, k_l = 1, \ldots, M$; $j = 1, \ldots, p$.

Since it is assumed that the second derivative of $S$ with respect to $\theta$ is nonsingular, these equations can be used to compute derivatives of $g$ to arbitrary order in a recursive manner, given sufficiently high differentiability. Notice that if $\hat{\theta}$ corresponds to an optimum of $S$, then in most cases the second derivative of $S$ will be either positive or negative definite, so the assumption of nonsingularity will be met.

The derivatives of $g$ can now be used as sensitivity coefficients to quantify the sensitivity of $\theta$ with respect to $x$. Taylor expansion of $g$ around $\tilde{x}$ gives

$$\theta_i = g_i(x) = g_i(\tilde{x}) + \sum_{k_1=1}^{M} \frac{\partial g_i}{\partial x_{k_1}} \delta x_{k_1} + \frac{1}{2} \sum_{k_1,k_2=1}^{M} \frac{\partial^2 g_i}{\partial x_{k_1} \partial x_{k_2}} \delta x_{k_1} \delta x_{k_2} + \mathcal{O}(|\delta x|^3), \quad (3.62)$$

where $\delta x = x - \tilde{x}$. If we take $\delta\theta = \theta - \hat{\theta}$, then

$$\delta\theta_i = \sum_{k_1=1}^{M} \frac{\partial g_i}{\partial x_{k_1}} \delta x_{k_1} + \frac{1}{2} \sum_{k_1,k_2=1}^{M} \frac{\partial^2 g_i}{\partial x_{k_1} \partial x_{k_2}} \delta x_{k_1} \delta x_{k_2} + \mathcal{O}(|\delta x|^3). \quad (3.63)$$

The sensitivity coefficients in this expression can also be normalized to obtain an equation, in terms of relative changes, analogous to (3.13) of section 3.4.1.

## 3.7 Concluding remarks

In this chapter an overview has been presented of methods for sensitivity analysis based on derivative information. The derivative information is used to construct a local approximation for the relation between the input and output variables.

Global sensitivity analysis allows the study of models in a wide range of the values of their parameters. For large models, however, computation of global sensitivity information may become prohibitively expensive. In that case, the types of local sensitivity analysis described in this chapter may provide a feasible alternative for obtaining the required sensitivity information.

Next to methods for standard propagation of perturbations, we have presented a method for the propagation of expectation and variance-covariance through a response function in particular detail. We discussed its derivation, its validation, and showed that it may be used to obtain a decomposition of variance with respect to the input sources of uncertainty. We further introduced precision sensitivity coefficients that provide insight into the influence of the input sources of uncertainty on the output precision. It was shown that the components into which the variance is decomposed (based on a linear approximation) have an alternative interpretation as marginal precision coefficients. Finally, unlike many global methods, the propagation method is able to deal naturally with cases where the input variables are dependent.

The performance of the methods based on local approximation was compared to that of a global decomposition method using simple random sampling. A fair comparison with respect to runtime performance between the methods is difficult since the performance of the sampling method can be improved by using a more intelligent sampling strategy. Nonetheless, on a Pentium II 350 MHz, 64 Mb memory machine the decomposition method based on local approximation using automatic differentiation took less than a second to compute the decomposition, whereas the global method took at least one hour to obtain similar results. Even then, the results of repeated runs showed that convergence of the values obtained by the global method could still improve. Again, using Latin hypercube sampling or a sampling method designed specifically for the purpose of computing the variance of conditional expectations, these computation times can be significantly reduced, but one should note the example considered here is still only a very small sensitivity problem.

We used the propagation method to study the State Forest Account Method. By considering the carbon content contribution of the pine species in a hypothetical ecoregion, we showed that certain regression coefficients, relating forest stand characteristics to phytomass fraction density, are an important source of uncertainty in the carbon contribution. We further showed by means of the uncertainty propagation method that even though the regression coefficients are determined at test plots with homogeneous characteristics, the use of the fitted relation at the inhomogeneous ecoregion level is not a matter of great concern.

As a further example of derivative-based methods for sensitivity analysis we discussed the investigation of implicit relations. Higher-order approximations for the sensitivity of parameters, estimated by means of for instance a maximum likelihood method, were given with respect to secondary variables in the estimation process.

In all these cases the derivatives may be obtained by means of automatic differentiation. As discussed in Chapter 2 several implementation types are available to

this end. Also the efficiency of automatically generated derivatives in comparison to numerically approximated derivatives is discussed there. Using the operator overloading implementation type of automatic differentiation, analysis routines that use the derivatives of a user-defined function can be implemented without reference to the actual content of this function. This strategy was followed in the design of the C++ libary in which the derivative-based methods of this chapter have been implemented. A number of technical details concerning the implementation of the routines for, among others, sensitivity analysis, will be discussed in more detail in Chapter 7.

# Chapter 4

# Statistical inference and model selection for nonlinear models

In this chapter we discuss methods for nonlinear parameter estimation, statistical inference and model selection. First, estimation and statistical inference are introduced from a likelihood perspective. Strict likelihood maximization is in general not desirable and methods are presented to select models with an appropriate balance between support from the data and structural simplicity. Next, we discuss the influence of model nonlinearity on these issues and consider derivative-based methods for the construction of accurate inferential information in the presence of model nonlinearity. The derivative-based methods can be implemented by automatic differentiation, in that way increasing the feasibility of statistical inference for complex nonlinear models.

## 4.1 Introduction

In this chapter we deal with the issue of extracting information from data by means of (parameterized) mathematical probability models. We do so in two parts. In the first part general principles of likelihood-based parameter estimation are discussed. In the second part we work out these principles for normal nonlinear regression problems. There we pay particular attention to methods for dealing with nonlinearity of the models. We consider several derivative-based methods for measuring the effects of nonlinearity and for the construction of accurate inferential information in the presence of model nonlinearity. Our aim is to provide the background for the development of routines for the analysis of parameter estimation procedures by methods that can be implemented by automatic differentiation.

We start by introducing a number of modeling concepts. By using models, observations are transformed into estimates of variables, *parameters*, that we can-

not, or do not want to, measure directly. Parameter estimation has a variety of purposes, but generally one is interested in *statistical inference*, i.e. using data to answer substantive questions. Welsh (1996) provides an introduction to statistical inference paradigms and to typical questions that may be addressed. See also Robins and Wasserman (2000) for a recent discussion on a number of issues regarding the foundations of statistical inference.

In this chapter a *model* is defined as a set of probability density[1] functions for a random vector of observations, indexed by a vector of parameters. Each density function is assumed to be a candidate for describing a real world situation that one attempts to model. Each parameter vector thus corresponds in a sense to a *hypothesis* about reality.

The use of data for the investigation of *relative support* for the different candidates is central to statistical inference. Perhaps the single most important notion in this respect is that of *likelihood*. The likelihood for a candidate parameter vector, or hypothesis, is the probability density value of the data given that the hypothesis is true, i.e. it is the value of the probability density function, corresponding to the parameter vector, at the realization of the observation vector. By means of the likelihood ratio these values can be used directly to measure relative support. In section 4.2 we discuss the relations between likelihood and several types of statistical inference in more detail.

The usefulness of parameter estimation for extraction of information from data will to an important extent depend on the appropriateness of choices with respect to the structure of the probability models and the design of the study. For example, which should be the regressor or independent variables; which variables are those to be modeled; and which replication structure do we use? What is the appropriate complexity of the model relationships?

In this chapter we will mainly focus on the relationship between parameters and observed variables. We use the term *model structure* if we want to emphasize the mathematical definition of the parameterized family of density functions that constitute a model. The model structure quantifies how the probability density of an observation vector is assumed to vary with the different values of the parameters. In general, some of these parameters will be of direct interest, as they are related to the purpose of the modeling effort, whereas other parameters are introduced out of necessity. The values associated with the latter nuisance parameters are required to describe the relationship between the parameters of interest and the density function of the observation vector.

The model structure should be informative with respect to the parameters, i.e. the parameters to be estimated should have a clear effect on the probability distribution of the observed variables. The evaluation of this level of informativeness is an important model design issue. At the same time, mathematical relationships between parameters and observed variables should, based on our scientific understanding, be sufficiently accurate in their representation of the real world situation

[1]Although most of the theory presented in this chapter also holds for discrete random variables, or probability measures in general, we will use terminology and notation for continuous random variables throughout.

in order to be able to serve as a basis for inference.

With respect to model structure complexity it is important to realize that when modeling phenomena in reality, it is generally possible to describe patterns in increasingly higher levels of detail. Modeling of more subtle effects will, however, also put higher demands on the data quality (and quantity). In practice, quality of available data will often not be determined by statistical considerations alone and often be limited due to physical or economic constraints. This means that a very detailed model, even if it is an accurate representation of reality, may often not be the best model to use and that models of lower structural complexity can lead to more useful parameter estimates: the model structure must be chosen in a manner that trades off underfitting and overfitting. In section 4.3, model selection methods are discussed that can aid in the choice of a model structure complexity that is appropriate in relation to the quality of the available data.

In section 4.4 the influence of model structure nonlinearity on both statistical inference and model selection is further worked out. We focus our attention on an important special case of maximum likelihood estimation: the normal nonlinear regression problem. Nonlinear regression is used in virtually all fields of science for the fitting of models to data and a large body of literature is available on the subject; Seber and Wild (1989) provide a particularly valuable account. We first provide some background on the interaction between model nonlinearity and parameter estimation and inference. To this end we exploit the special structure of the problem to present a clear geometrical interpretation of this interaction.

We then go on to discuss derivative-based methods for both quantifying the extent of nonlinearity and the construction of accurate inferential information in the presence of nonlinearity. It will be shown that the methods for sensitivity analysis discussed in Chapter 3 can also contribute to this end.

## 4.2 Likelihood in statistical inference

Consider a model, i.e. a parameterized family of probability density functions $g$ for a random vector of observed variables $Y$, denoted by

$$\{g(y|\theta)|\theta \in \Theta\}, \tag{4.1}$$

where the parameters $\theta$ are taken from a parameter space $\Theta$. An observation vector realization $y$ leads to a likelihood function $\mathcal{L}_y(\theta) = g(y|\theta)$ that provides a likelihood value for each of the parameter vector values. The maximum likelihood estimate $\hat{\theta}(y)$ maximizes the likelihood function.

Likelihood plays a central role in statistical inference, see for instance Reid (2000) for an overview. In the likelihood paradigm of statistical inference support for hypotheses must always be relative. The law of likelihood asserts that the amount of evidential support that data provide for one hypothesis relative to another, is quantified by the likelihood ratio, see for example Hacking (1965).

This can be formulated as:

If hypothesis $A$ implies that the probability density for a random variable $Y$ to take the value $y$ is $p_A(y)$, while hypothesis $B$ implies that the probability density is $p_B(y)$, then the observation $Y = y$ provides evidence supporting $A$ over $B$ if and only if $p_A(y) > p_B(y)$. The likelihood ratio, $p_A(y)/p_B(y)$ measures the strength of that evidence.

See Royall (1997) for a detailed defense of this approach. For a discussion of the likelihood principle, which asserts that two experiments that yield proportional likelihood functions should yield identical inferences, and its relation to other principles such conditionality and sufficiency, we also refer to Birnbaum (1962).

Based on the law of likelihood, the likelihood ratio $\mathcal{L}_y(\hat{\theta})/\mathcal{L}_y(\theta)$ can be used to construct likelihood regions of parameter values that have a certain level of support relative to the maximum likelihood estimate. Various authors, such as Fisher (1956), Kass and Raftery (1995) and Royall (2000), have made useful suggestions on how to relate likelihood ratio values to varying degrees of plausibility. Royall (2000), for example, uses an approach based on comparison to so-called canonical experiments. Likelihood ratios close to 1 represent only weak evidence for one hypothesis over another, whereas likelihood ratios around 8 represent fairly strong evidence. Likelihood ratios higher than 32 represent very strong evidence favoring for instance a maximum likelihood estimate over applicable alternative parameter values.

It is important to note that likelihood is defined only for simple hypotheses, i.e. for a single parameter or density function at a time. For dealing with composite hypotheses and nuisance parameters, several special types of likelihood have been suggested, for example profile and modified profile likelihood, and conditional and marginal likelihood. See Mukerjee and Reid (1999) for an overview.

In the likelihood paradigm it is recognized that if a meaningful prior probability density function is available (see for instance Edwards (1972)), the likelihood is the factor by which the prior probability is changed by the observation of the data. This follows from Bayes' theorem, by which we have

$$p(\theta|y) \propto \mathcal{L}_y(\theta)\pi(\theta), \tag{4.2}$$

where $\pi(\theta)$ is the prior probability density function. Note that the posterior probability density $p(\theta|y)$ depends on the data only through the likelihood function. A similar relation holds for the posterior odds which are equal to the prior odds multiplied by the likelihood ratio

$$\frac{p(\theta_1|y)}{p(\theta_2|y)} = \frac{\mathcal{L}_y(\theta_1)}{\mathcal{L}_y(\theta_2)}\frac{\pi(\theta_1)}{\pi(\theta_2)}. \tag{4.3}$$

In Bayesian statistics probability is used as a fundamental measure of uncertainty, and also there likelihood is used to transform prior probability into posterior probability by means of (4.2). As such, Bayesian methods are particularly suitable

for dealing with a priori beliefs and for answering inference questions regarding what to belief. This does however rely on the ability to specify a suitable prior probability density. Kass and Wasserman (1996) discuss difficulties related to the construction of priors, in particular of the noninformative type. Bernardo and Smith (1994) give a review of the foundations of Bayesian inference. For detailed general accounts on Bayesian inference we refer to Box and Tiao (1992) and Gelman et al. (1995).

The frequentist approach to inference is based on the investigation of statistics, i.e. of random variables with distributions that depend only on the 'true' distribution of the observation vector and not on any unknown parameters. Generally this true distribution is unknown, but the model can be used to derive a distribution of the statistic that is valid at least whenever the true distribution (or density) is an element of the model family. Unlike inference respecting the likelihood principle, frequentist inference is generally not just based on the observed data, but also on what might have been observed. Even though the methods thus require assumptions regarding the true distribution, this approach can nonetheless provide valuable information in case the assumptions are indeed warranted.

Frequentist probability statements are often based on the distribution of the likelihood function and derived quantities such as the maximum likelihood estimator. Barndorff-Nielsen's approximation (Barndorff-Nielsen, 1983) shows that the distribution of the maximum likelihood estimator is obtained to a very high order of approximation directly from the likelihood function. Also the model selection criteria that are discussed in the next section, are based on the distribution of the likelihood function. It will be shown there, that the criteria may aid in the selection of an appropriate model structure by providing a 'rate of exchange' between data support and model simplicity.

## 4.3 Model selection

We assume that we have a collection of $R$ alternative models for the same data, denoted by

$$\{g^r(y|\theta_r)|\theta_r \in \Theta_r\}_r \equiv \left\{g^r_{\theta_r}|\theta_r \in \Theta_r\right\}_r, \quad r = 1, \ldots, R. \tag{4.4}$$

Note that the dimension of the parameter space may differ between the families of density functions, i.e. we take $\dim \Theta_r = p_r$. We shall further, tentatively, assume that the observed data are a realization of a random vector $Y$ (with $n$ components) for which a probability density function exists. This reflects the idea that the observation vector is variable in the following sense: if the data could be collected more than once, one would not obtain identical results each time. The density function will be denoted by $f(y)$ and we will refer to it as the *true* density function.

The true density function will in general be unknown, but some of its characteristics may still be influenced at the design stage. For example, apart from the

choice of which variables to observe in the first place, it may be also be possible to choose a certain experimental design and replication structure.

Now consider the models of (4.4). Each model has an associated maximum likelihood estimator $\hat{\theta}_r(Y)$. The models may differ in their complexity, i.e. in the level of detail in which they describe the process that leads to the data. For models of higher complexity, one generally expects to get higher maximum likelihood values $\mathcal{L}(\hat{\theta}_r(y)) = g(y|\hat{\theta}_r(y))$.

In Figure 4.1 (a) an example is shown of a true density function, together with a particular realization of the associated random vector. For a model structure that is too complex relative to the available data, the resulting estimated density function will be determined to a large extent by the peculiarities of the particular realization. This is shown in 4.1 (b). Due to the high level of freedom in the model, a high likelihood value can be obtained; the density tends to concentrate around the realization. This unstable behavior is referred to as *overfitting*. If, however, the model structure is based on a description that is low in its level of detail, the family of density functions may not contain a good approximation to the true density. This is called *underfitting*, see 4.1 (c). In this case the mode of the density function cannot get close to the observed realization. This commonly leads to the spreading out of the density function to increase the likelihood value, thus overestimating the variance of the observations.

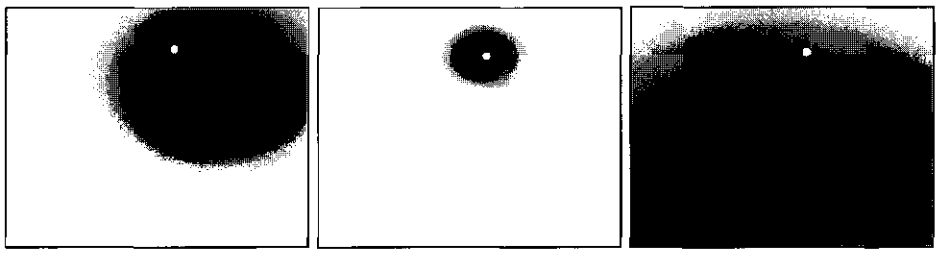
These examples show that we are interested in designing models for which the density function associated with the maximum likelihood estimate is expected to be close to the true density function, rather than in models for which likelihood maximization just centers the estimated density function around the data. In the following we will discuss model selection methods that attempt to estimate how close the maximum likelihood density is expected to be to the true density function. As a measure of the distance between the density functions, we will use the Kullback-Leibler (KL) discrepancy. In section 4.3.3 we show that minimization of the expected Kullback-Leibler discrepancy relative to the true density is equivalent to maximization of the expected support for the estimated parameters based on the likelihood with respect to observations independent of the data used to obtain the parameter estimate.

### 4.3.1 Selection by means of the expected KL-discrepancy

The KL-discrepancy between two density functions $f(y)$ and $g(y)$ is defined by

$$I(f,g) = \int f(y) \log \frac{f(y)}{g(y)}\, dy. \tag{4.5}$$

We have that $I(f,g) \geq 0$ with equality if and only $f(y) = g(y)$ (a.e.). The KL-discrepancy can be interpreted as a relative entropy; see Appendix B for a discussion of the underlying information-theoretic principles. It is shown there that the discrepancy should be interpreted as a distance to a particular fixed probability density function; in our case this will be the true density function $f(y)$.



(a) True density (b) Overfitting (c) Underfitting

Figure 4.1: Three figures show density functions on a sample space, depicted here as two-dimensional. In (a) the true density function and a realization of the random observation vector (indicated by the white dot) are shown. Resulting estimated density functions in case of overfitting and underfitting are shown in (b) and (c), respectively. The grayscale of the points in sample space is proportional to the probability density function.

Consider a single model, denoted by $\{g_\theta | \theta \in \Theta\}$. We assume that a unique probability density function $g_{\theta_0}$ exists which has the smallest KL-discrepancy with the true density, i.e.

$$\theta_0 = \arg\min I(f, g_\theta). \tag{4.6}$$

The density function $g_{\theta_0}$ is referred to as the best approximating density, and $I(f, g_{\theta_0})$ as the *discrepancy due to approximation*, or also, as the *model bias*.

Here we will consider the case that maximum likelihood estimation is used to obtain a parameter vector and thereby an associated density function. See Linhart and Zucchini (1986) for a model selection framework using general estimators.

The maximum likelihood estimator is a random variable depending on the observation vector, and the density function $g_{\hat{\theta}}$ will in general not be equal to $g_{\theta_0}$. This, in turn, means that $I(f, g_{\hat{\theta}})$ is also a random variable. In White (1982) it is shown that asymptotically, i.e. for large sample sizes, the expectation of $\hat{\theta}$ is equal to $\theta_0$ and the expectation of $I(f, g_{\hat{\theta}})$ is equal to $I(f, g_{\theta_0})$. Expressions for the asymptotic variances of $\hat{\theta}$ and $I(f, g_{\hat{\theta}})$ can be found in Burnham and Anderson (1998). To quantify the difference between $g_{\hat{\theta}}$ and $g_{\theta_0}$, Linhart and Zucchini (1986) introduce the *discrepancy due to estimation* as $I(g_{\theta_0}, g_{\hat{\theta}})$. This discrepancy is also referred to as the *model variance*.

The total discrepancy $I(f, g_{\hat{\theta}})$ is a combined result of the discrepancy due to approximation and the discrepancy due to estimation. In general these effects act in opposition to each other: if the discrepancy due to approximation decreases for a more complex model structure, the discrepancy due to estimation generally increases.

For model selection, we intend to compare estimates of the expected KL-discrepancy for each of the $R$ models of (4.4). We select the model for which

the expectation of $I(f, g^r_{\hat{\theta}_r})$ is as small as possible, i.e. we want to minimize

$$\mathrm{E}_X[I(f, g^r_{\hat{\theta}_r(x)})] = \mathrm{E}_X[\int f(y) \log(\frac{f(y)}{g^r(y|\hat{\theta}_r(x))})\, dy], \tag{4.7}$$

over $r = 1, \ldots, R$. Here and in the following both $X$ and $Y$ are (independent) random vectors with true density function $f$; $x$ and $y$ denote their respective realizations. The notation $E_X$ indicates that the expectation is taken with respect to the density function of the random vector $X$.

### 4.3.2 The Akaike information criterion (AIC)

Estimators of the model-dependent part of the expected KL-discrepancy are called criteria. Usually they are defined in such a way that they must be minimized. Here we provide an overview of the properties of a number of useful criteria.

Rewriting (4.7), and no longer explicitly indicating the dependency on $r$, we have

$$\mathrm{E}_X[I(f, g_{\hat{\theta}(x)})] = \int f(y) \log(f(y))\, dy - \mathrm{E}_X[\int f(y) \log(g(y|\hat{\theta}(x)))\, dy]. \tag{4.8}$$

Notice that the first term in this expression does not depend on the model to be selected. Hence for now we choose to maximize the target

$$T = \mathrm{E}_X[\int f(y) \log(g(y|\hat{\theta}(x)))\, dy] = \mathrm{E}_X[\mathrm{E}_Y[\log(g(y|\hat{\theta}(x)))]]. \tag{4.9}$$

It can be shown that, asymptotically (i.e. for $n \to \infty$),

$$T \approx \mathrm{E}_X[\log(g(x|\hat{\theta}(x)))] - \mathrm{tr}[J(\theta_0)(I(\theta_0))^{-1}], \tag{4.10}$$

where tr is the matrix trace operator, and

$$I(\theta) = \mathrm{E}_Y[-\frac{\partial^2}{\partial \theta^2} \log(g(y|\theta))], \tag{4.11}$$

and

$$J(\theta) = \mathrm{E}_Y[(\frac{\partial}{\partial \theta} \log(g(y|\theta)))(\frac{\partial}{\partial \theta} \log(g(y|\theta)))']. \tag{4.12}$$

See for instance Burnham and Anderson (1998) or White (1994) for a derivation of this result. The parameter vector $\theta_0$ corresponds to the best approximating density as defined in section 4.3.1. So criteria based on estimators for $T$ should have the following structural form

$$\hat{T} = \log(g(x|\hat{\theta}(x))) - \hat{\mathrm{tr}}[J(\theta_0)(I(\theta_0))^{-1}], \tag{4.13}$$

where $\hat{\mathrm{tr}}$ represents an estimator for the entire trace term. The Takeuchi information criterion (TIC), has the structural form

$$\mathrm{TIC} = -2 \log(g(x|\hat{\theta}(x)) + 2\, \mathrm{tr}[\hat{J}(\theta_0)(\hat{I}(\theta_0))^{-1}] \tag{4.14}$$

and is an example of a criterion based on this target. Burnham and Anderson (1998) discuss several possible estimators for $\mathrm{tr}[J(\theta_0)(I(\theta_0))^{-1}]$. Experimental results suggest, however, that estimators of the trace term are often highly variable, even for large $n$.

For this reason the use of the robust Akaike selection criterion is generally recommended. The estimator for the trace term used in this criterion is based on the result that if the true density corresponds to a particular element of the density function family (also in this case referenced by $\theta_0$), we have that $J(\theta_0) = I(\theta_0)$. This can be shown as follows. Since $\int g(y|\theta)\,dy = 1$, we have

$$\int \frac{\partial g(y|\theta)}{\partial \theta}\,dy = 0. \tag{4.15}$$

If we differentiate this expression again with respect to $\theta$, and use

$$\frac{\partial \log(g(y|\theta))}{\partial \theta} = \frac{1}{g(y|\theta)}\frac{\partial g(y|\theta)}{\partial \theta}, \tag{4.16}$$

we find that for every $\theta$

$$\mathrm{E}_g[(\frac{\partial}{\partial \theta}\log(g(y|\theta)))(\frac{\partial}{\partial \theta}\log(g(y|\theta)))'] = \mathrm{E}_g[-\frac{\partial^2 \log(g(y|\theta))}{\partial \theta^2}], \tag{4.17}$$

where $E_g$ denotes the expectation over a random variable $Y$ assuming that it has probability density function $g_\theta(y)$. Using $f = g_{\theta_0}$ we thereby also have,

$$J(\theta_0) = I(\theta_0). \tag{4.18}$$

Substitution of (4.18) in (4.13) now gives

$$\hat{T} = \log(g(x|\hat{\theta}(x))) - p. \tag{4.19}$$

where $p$ is the dimension of the parameter space $\Theta$. The maximization of this target is equivalent to the minimization of the Akaike information criterion (AIC) given by

$$\mathrm{AIC} = -2\log(g(x|\hat{\theta}(x))) + 2p. \tag{4.20}$$

Since the AIC depends on the observation vector realization, the AIC is the realization of a random variable depending on the random observation vector. The model with the smallest AIC realization value is thus not necessarily also the model with the smallest expected KL-discrepancy. Burnham and Anderson (1998) suggest the following interpretation of AIC values. If $\Delta_r$ is the difference in AIC value with the model that obtained the smallest AIC value, then for any model with $\Delta_r \leq 2$ there is no credible evidence that this model should be ruled out as being the model with lowest expect KL-discrepancy. For $2 < \Delta_r \leq 4$, $4 < \Delta_r \leq 7$ and $\Delta_r > 7$ there is weak, definite and strong evidence, respectively, that the model is not the 'KL-best model'. These recommendations should, however, only be used if one can reasonably assume that the assumptions leading to the AIC,

i.e. small model misspecification, large sample size and independent observations, are satisfied.

For nonlinear regression problems an improved estimator is available which is particularly useful for the case that the asymptotic assumption is not valid, i.e if the number of parameters $p$ to be estimated is relatively large compared to the number of observations $n$. This criterion will be discussed in section 4.4.

In Stone (1977) it is shown that model selection based on the Akaike information criterion is asymptotically equivalent to model selection by cross-validation. Model selection by cross-validation is discussed extensively in Stone (1974). Estimation of the expected KL-discrepancy by means of cross-validation methods is further discussed in Linhart and Zucchini (1986). Estimation by means of cross-validation is conceptually simple, but is computationally expensive, particularly for large sample sizes or models with large numbers of parameters.

### 4.3.3 Likelihood and the KL-discrepancy

Model selection by means of the KL-discrepancy is based on minimization of

$$\begin{aligned} \mathrm{E}_X[I(f, g_{\hat{\theta}}(x))] &= E_X[\int f(y) \log(\frac{f(y)}{g(y|\hat{\theta}(x))})\, dy] \\ &= \mathrm{E}_X[\mathrm{E}_Y[\log(\frac{f(y)}{g(y|\hat{\theta}(x))})]], \end{aligned} \tag{4.21}$$

over the available model families. The two expectations in the lower expression of (4.21) can be understood by considering both as averages over draws from the true distribution. First we take a realization $x$ using the true density function, leading to $\hat{\theta}(x)$ as the corresponding maximum likelihood estimate and $g(y|\hat{\theta}(x))$ as the estimated density function. Now if $y$ is also drawn using the true density independent of $x$, the log-likelihood ratio $\log(f(y)/g(y|\hat{\theta}(x)))$ quantifies the support provided by this realization $y$ for the estimated density function $g(y|\hat{\theta}(x)))$ relative to the true density function $f(y)$. Kullback-Leibler model selection thus selects the model which *on the average* provides the highest expected support by an independent observation for the estimated density relative to the true density. The independence of the observation prevents overfitting.

As,

$$\mathrm{E}_X[\mathrm{E}_Y[\log(\frac{f(y)}{g(y|\hat{\theta}(x))})]] = \mathrm{E}_Y[\log(f(y))] - \mathrm{E}_X[\mathrm{E}_Y[\log(g(y|\hat{\theta}(x)))]] \tag{4.22}$$

minimization of the expected Kullback-Leibler discrepancy is also equivalent to maximization of

$$\mathrm{E}_X[\mathrm{E}_Y[\log(g(y|\hat{\theta}(x)))]] = \mathrm{E}_X[\mathrm{E}_Y[\log(\mathcal{L}_y(\hat{\theta}(x)))]]. \tag{4.23}$$

This quantity can be understood in manner analogous to the quantity of (4.21). It is the average likelihood for the estimated density, where again the estimated

density is determined independently of the realization used to evaluate the likelihood.

The selection criteria discussed so far in this chapter, e.g. (4.14) and (4.20), consist of two terms. The first term, proportional to the negative of the maximum log-likelihood $-\log(\mathcal{L}_x(\hat{\theta}(x)))$, is a measure of the lack of model fit. The second term can be considered as a penalty for the complexity of the model. The criteria can thus be interpreted, heuristically, to determine a rate of exchange between support, measured by the log-likelihood, and the simplicity of the model. Other approaches to model selection, see for instance Schwarz (1978) for a Bayesian approach, may lead to different rates of exchange.

## 4.4 Normal nonlinear regression

### 4.4.1 Introduction

We now consider an important case of maximum likelihood estimation in more detail: the *normal nonlinear regression problem*. As indicated in the introduction, our main aim is to provide a general overview of derivative-based methods for statistical inference in the presence of nonlinearity. The special structure of this problem allows us to present a clear geometric interpretation of such methods.

In Appendix C various formulations of the nonlinear regression problem are discussed, leading to the following standard model

$$\left\{ g(y|\theta) = (2\pi\sigma^2)^{-\frac{n}{2}} \exp\left(\frac{-\|y - f(\theta)\|^2}{2\sigma^2}\right) \mid \theta \in \Theta, \sigma > 0 \right\} \tag{4.24}$$

i.e. we take a family of multivariate normal density functions parameterized by a vector $\theta$ and a variance parameter $\sigma^2$. The associated random vectors are denoted by

$$Y = f(\theta) + \varepsilon, \quad \varepsilon \sim N(0, \sigma^2 I_n). \tag{4.25}$$

Note that the design dependence is suppressed and the multivariate covariance structure is particularly simple. Both issues are discussed in Appendix C. Also note that two important changes of notation are introduced. First, $f$ no longer denotes the true density function. Second, the parameter vector $\theta$ no longer represents all parameters, since there is now an additional parameter, $\sigma$, that has its own name. We still consider $\theta$ as an element from a $p$-dimenional parameter space $\Theta$.

We assume that the function $f : \Theta \to S$ is injective and maps the parameter space $\Theta$ into a $p$-dimensional manifold $\mathcal{F}$ in the $n$-dimensional sample space $S$:

$$\mathcal{F} = \left\{ f(\theta) \in S | \theta \in \Theta \right\}. \tag{4.26}$$

The manifold $\mathcal{F}$ is called the *expectation manifold* as for each $\theta$, $f(\theta)$ is the expectation of the random vector corresponding to $\theta$. The nonlinearity of the problem can be characterized in terms of the curvature of this manifold and its parameterization. This is discussed in sections 4.4.2 and 4.4.4.

### Estimation

The likelihood and log-likelihood functions for an observation vector realization $y$ are given by

$$\mathcal{L}_y(\theta, \sigma) = (2\pi\sigma^2)^{-\frac{n}{2}} \exp(\frac{-\|y - f(\theta)\|^2}{2\sigma^2}), \tag{4.27}$$

and

$$\log \mathcal{L}_y(\theta, \sigma) = -\frac{n}{2}\log(2\pi\sigma^2) - \frac{\|y - f(\theta)\|^2}{2\sigma^2}. \tag{4.28}$$

respectively. The maximum likelihood estimators $\hat{\theta}(Y)$ and $\hat{\sigma}(Y)$ satisfy

$$\hat{\theta}(y) = \arg\min_{\theta} \|y - f(\theta)\|^2, \quad \text{and,} \quad \hat{\sigma}^2(y) = \frac{\|y - f(\hat{\theta}(y))\|^2}{n}, \tag{4.29}$$

for each realization $y$. The maximum likelihood estimator for $\sigma$ is biased; an unbiased estimator for $\sigma$ equals $(\|y - f(\hat{\theta}(y))\|^2/(n-p))^{1/2}$. The maximum likelihood estimator $\hat{\theta}(Y)$ is equal to the least-squares estimator as it minimizes

$$S(\theta) = \|y - f(\theta)\|^2, \tag{4.30}$$

i.e. $\hat{\theta}(y)$ minimizes the distance between $f(\theta)$ on the expectation manifold and the realization $y$ of the observation vector. It follows that $f(\hat{\theta}(y))$ is the projection of the realization $y$ on the expectation manifold. If $f$ is linear, say $f(\theta) = X\theta$, the maximum likelihood estimator for $\theta$ is $\hat{\theta} = (X'X)^{-1}X'y$. We observe that also in the nonlinear case $f(\hat{\theta})$ is the projection of $y$ on the tangent plane at $f(\hat{\theta})$, i.e.

$$\hat{\theta}(y) = (D\hat{f}'D\hat{f})^{-1}D\hat{f}'y, \tag{4.31}$$

where $D\hat{f} = D^1 f(\hat{\theta}(y))$ (see Appendix A for the notation). However, $\hat{\theta}$ generally cannot be obtained directly from this expression as in the linear case, and must be computed by (iterative) minimization of $\|y - f(\theta)\|^2$ instead.

### Inference

In section 4.2 likelihood-type inference was discussed based on relative evidential support quantified by the likelihood ratio $\mathcal{L}_y(\hat{\theta})/\mathcal{L}_y(\theta)$. The construction of likelihood regions for nonlinear models is discussed in section 4.4.3.

With regard to frequentist-type inference, we first list some main results for the case of a linear model $f(\theta) = X\theta$. A $100(1-\alpha)\%$ confidence region is a random region in parameter space the realization of which depends on the outcome of the observation vector. Given that the observations have a distribution corresponding to true values of $\theta^*$ and $\sigma^{*2}$, the random region covers these (fixed) values with probability $100(1-\alpha)\%$. Two examples of $100(1-\alpha)\%$ confidence regions, are

- for individual parameters $\theta_i, i = 1, \ldots p$, also known as Wald confidence intervals:

$$\hat{\theta}_i \pm t_{n-p}^{\alpha/2}\sqrt{\frac{S(\hat{\theta})}{n-p}\hat{c}_{ii}} \tag{4.32}$$

where $\hat{c}_{ii}$ is the $i$-th diagonal element of the matrix $(X'X)^{-1}$ and $t_{n-p}^{\alpha/2}$ is the upper $\alpha/2$-quantile of the $t$-distribution, and,

- for all parameters (except $\sigma$) simultaneously:

$$\left\{\theta|(\theta-\hat{\theta})'X'X(\theta-\hat{\theta}) \leq \frac{p}{n-p}S(\hat{\theta})F_{p,n-p}^{\alpha}\right\} = \left\{\theta|S(\theta) \leq cS(\hat{\theta})\right\}, \quad (4.33)$$

where

$$c = 1 + \frac{p}{n-p}F_{p,n-p}^{\alpha} \quad (4.34)$$

and where $F_{p,n-p}^{\alpha}$ is the upper $\alpha$-quantile of the $F_{p,n-p}^{\alpha}$ distribution.

If we assume that the observation vector has a density function corresponding to $\theta = \theta^*$ and $\sigma = \sigma^*$, the distribution of $\hat{\theta}$ follows directly from the distribution of $y$, since $\hat{\theta} = (X'X)^{-1}X'y$. We then have $\hat{\theta} \sim N(\theta^*, \sigma^{*2}(X'X)^{-1})$ and the maximum likelihood estimator is unbiased, since $\mathrm{E}[\hat{\theta}] = \theta^*$.

For nonlinear functions $f$, inference is usually based on linearization of $f$ around the maximum likelihood estimate: $f(\theta) \approx f(\hat{\theta}) + D\hat{f}(\theta - \hat{\theta})$. Asymptotically (i.e. for $n \to \infty$) we have

$$\hat{\theta} \sim N(\theta^*, \sigma^{*2}((D^1f(\theta^*))'(D^1f)(\theta^*))^{-1}). \quad (4.35)$$

For finite $n$, however, we have in general that (i) $\hat{\theta}$ is not normally distributed, (ii) $\hat{\theta}$ is biased, i.e. $\mathrm{E}[\hat{\theta}] \neq \theta^*$, and (iii) $\mathrm{var}(\hat{\theta}, \hat{\theta}) \neq \sigma^{*2}((D^1f(\theta^*))'(D^1f(\theta^*)))^{-1}$. In section 4.4.5 corrections for the moments of the maximum likelihood estimator are discussed. In Donaldson and Schnabel (1987) it is shown that also the confidence regions based on a linear approximation of the expectation manifold often do not have the desired coverage properties. In section 4.4.3 solutions to this problem are discussed.

### Model selection

For the case that a model has a normal error structure as in (4.24), i.e. for normal nonlinear regression problems, an estimator for the expected Kullback-Leibler discrepancy can be derived that is more accurate than the AIC. This corrected Akaike information criterion (AICc) was derived by Hurvich and Tsai (1989) and is given by

$$\mathrm{AICc} = \mathrm{AIC} + \frac{2p(p+1)}{n-p-1}, \quad (4.36)$$

where $p$ is here the total number of parameters including $\sigma$. The correction term provides a small sample correction for the AIC, i.e for the case that the number of parameters $p$ to be estimated is relatively large compared to the number of observations $n$. Note that, as the correction term is not stochastic, the improved accuracy is obtained without any increase in variance of the estimator. The derivation of the AICc does rely on a linearization of the expectation manifold. As, however, both the expected Kullback-Leibler expectation and the AICc

are invariant under reparameterizations, the only nonlinearity relevant to this issue is the so-called intrinsic nonlinearity (see 4.4.2) which is itself invariant under reparameterizations.

### 4.4.2 Nonlinearity measures

To quantify model nonlinearity, Bates and Watts (1980) proposed the *intrinsic* and *parameter-effects* curvature measures. To obtain definitions for the two types of curvature, consider the nonlinearity of $f$ at an arbitrary point $\theta_0 \in \Theta$. Using the notation of Appendix A, the Taylor expansion of $f$ around $\theta_0$ is given by

$$f_i(\theta) = f_i(\theta_0) + \sum_{j_1} (D^1_{j_1} f_i)\delta\theta_{j_1} + \frac{1}{2}\sum_{j_1,j_2} (D^2_{j_1 j_2} f_i)\delta\theta_{j_1}\delta\theta_{j_2} + \mathcal{O}(|\delta\theta|^3), \qquad (4.37)$$

where $\delta\theta = \theta - \theta_0$ and the derivatives are evaluated at $\theta_0$.

The validity of the tangent space approximation used to obtain linear inference estimates depends on the magnitude of the quadratic term (neglecting, for now, terms of order three and higher) relative to that of the linear term. The vector in sample space corresponding to the quadratic term may be decomposed into a component in the tangent space at $\theta_0$ and a component in the orthogonal complement of this tangent space, see Figure 4.2.

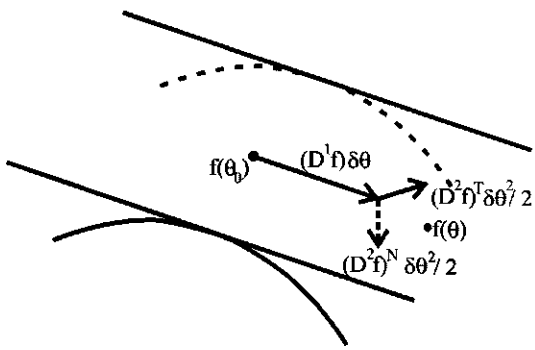



Figure 4.2: Decomposition of the second order Taylor term into a component in tangent space and a component in the orthogonal complement of the tangent space.

We thus have

$$(D^2 f)(\theta_0)\delta\theta^2 = (D^2 f)^T(\theta_0)\delta\theta^2 + (D^2 f)^N(\theta_0)\delta\theta^2 \qquad (4.38)$$

where $(D^2 f)^T$ and $(D^2 f)^N$ are obtained by (left) multiplication of $D^2 f$ with the appropriate projection matrices (see again appendix A). The component in the tangent space leads to the parameter-effects curvature, and the component in the orthogonal complement to the intrinsic curvature.

The nonlinearity measures defined by Bates and Watts (1980) are now

$$K_{\delta\theta}^{T} = \frac{\|(D^2 f)^T(\theta_0)\delta\theta^2\|}{\|(D^1 f)(\theta_0)\delta\theta\|^2}, \quad \text{and} \quad K_{\delta\theta}^{N} = \frac{\|(D^2 f)^N(\theta_0)\delta\theta^2\|}{\|D^1 f(\theta_0)\delta\theta\|^2}, \tag{4.39}$$

for the parameter-effects curvature and the intrinsic curvature, respectively, both in the direction $\delta\theta$. (Seber and Wild, 1989, Appendix B5) show that the intrinsic curvature is invariant under (differentiable) reparameterizations of the expectation manifold.

To facilitate comparison, the nonlinearity measures are commonly scaled to dimensionless quantities by dividing by a characteristic length in the problem. Bates and Watts (1980) use a characteristic length based on the radius of a confidence sphere obtained by mapping the linear approximation to the $100(1-\alpha)\%$ confidence ellipsoid on the expectation manifold. These scaled measures and other quantities derived from the intrinsic and parameter-effects curvatures used to quantify nonlinearity are further discussed in (Seber and Wild, 1989, Chapter 4). We shall take the practical approach of measuring nonlinearity mainly by its effect on various statistical procedures of statistical inference, such as by the computation of parameter bias, variance-covariance and likelihood regions.

### 4.4.3 Inference regions and nonlinearity

We show how the two types of curvature complicate the construction of inference regions. As an example of an inference region we take a likelihood-based region; similar considerations apply to confidence regions. A likelihood region consists of parameter values that satify the property that the ratio $\mathcal{L}_y(\theta,\sigma)/\mathcal{L}_y(\hat{\theta}(y),\hat{\sigma}(y))$ is smaller than a given constant, say $0 < c < 1$. The importance of such regions was discussed in section 4.2.

In the following, we will only compare parameter combinations for which $\sigma$ is fixed at the maximum likelihood estimate $\hat{\sigma}(y) = S(\hat{\theta}(y))/n$. This leads to a type of profile likelihood region, see for instance Mukerjee and Reid (1999). To obtain the profile likelihood region, we must then construct the likelihood contour consisting of parameters for which the likelihood ratio is exactly equal to $c$. From (4.27), (4.28) and (4.30) it follows that the likelihood contour of parameter values that satisfy $\mathcal{L}_y(\theta,\hat{\sigma}(y))/\mathcal{L}_y(\hat{\theta}(y),\hat{\sigma}(y)) = c$, can also be defined by

$$\left\{\theta | S(\theta) - S(\hat{\theta}) = \delta^2\right\}, \tag{4.40}$$

where $\delta^2 = -2\hat{\sigma}^2 \log c$. Since $S(\theta) = \|y - f(\theta)\|^2$, this also means that the points on the expectation manifold corresponding to the likelihood contour all have the same distance $\eta = (\delta^2 + S(\hat{\theta}))^{\frac{1}{2}}$ to the observation vector realization $y$. For a linear model $S(\theta) - S(\hat{\theta}) = \|f(\theta) - f(\hat{\theta})\|^2$, so in that case the points of the expectation manifold with distance $\eta$ to $y$ are on a sphere with centre $f(\hat{\theta})$ and radius $\delta$.

From Figure 4.2 we recall that a point on the expectation manifold can be

approximated as a sum of vectors, namely

$$f(\theta) = f(\hat{\theta}) + (D^1\hat{f})\delta\theta + \frac{1}{2}(D^2\hat{f})^T(\delta\theta)^2 + \frac{1}{2}(D^2\hat{f})^N(\delta\theta)^2 + \mathcal{O}(|\delta\theta|^3), \quad (4.41)$$

where $\delta\theta = \theta - \hat{\theta}$. For the notation we refer to Appendix A. Starting from $f(\hat{\theta})$ we first add the linear term $(D^1\hat{f})\delta$ and next the terms $\frac{1}{2}(D^2\hat{f})^T(\delta\theta)^2$ and $\frac{1}{2}(D^2\hat{f})^N(\delta\theta)^2$ responsible for the parameter-effects and intrinsic curvature, respectively. Note that $(D^1\hat{f})\delta\theta + \frac{1}{2}(D^2\hat{f})^T(\delta\theta)^2$ lies in the tangent space and that $\frac{1}{2}(D^2\hat{f})^N(\delta\theta)^2$ is perpendicular to the tangent space.

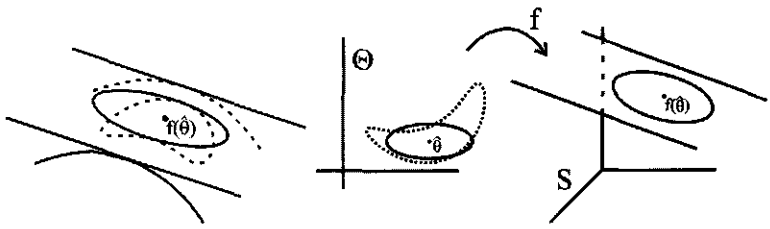
The construction of the likelihood contour in parameter space can now be visualized to proceed in two stages. First consider Figure 4.3 (a). It shows the expectation manifold and the tangent space at $f(\hat{\theta})$. The dashed contour indicates the points on the expectation manifold with distance $\eta$ to the realized observation vector $y$. Since $f$ maps the points of the likelihood contour in parameter space into this contour, we will refer to it as the *expectation contour*.

Due to intrinsic curvature, the tangent space at $f(\hat{\theta})$ deviates from the expectation manifold and the expectation contour is no longer a (sub)sphere as in the linear case. We can, however, project the expectation contour on the tangent space. From (4.41) it follows that by projection the term $\frac{1}{2}(D^2\hat{f})^N(\delta\theta)^2$ perpendicular to the tangent space vanishes. By projection we thus get the solid contour of Figure 4.3 (a), called the *tangent contour*, which is related to the likelihood contour in parameter space by the map $f(\hat{\theta}) + (D^1\hat{f})\delta\theta + \frac{1}{2}(D^2\hat{f})^T(\delta\theta)^2$. In Appendix D it is shown that the tangent contour is an ellipsoid in the tangent space.

The second stage, shown in Figure 4.3 (b), is to construct the likelihood contour in parameter space from the tangent contour. This construction is complicated by the presence of the term $\frac{1}{2}(D^2\hat{f})^T\delta\theta^2$ in the map that relates the two contours. If there were no parameter-effects curvature the two contours would be related by a linear map; Appendix D shows the likelihood contour is then an ellipsoid, depicted in Figure 4.3 (b) as the solid contour, which is easy to construct.

For linear maps, straight parallel equispaced lines in parameter space are mapped onto straight parallel equispaced lines in the tangent space. For nonlinear maps this is generally not the case and the parameter-effects curvature quantifies to which extent a regular grid is deformed when mapped onto the tangent space. If the parameter-effects curvature is not zero, then the likelihood contour may have a more complicated shape, indicated by the dashed contour in Figure 4.3 (b), and is more difficult to compute. This problem can be solved by using a reparameterization of the expectation manifold to reduce the parameter-effects curvature. In Appendix E several strategies for choosing reparameterizations are discussed (the curvature arrays used in the computations are described below).

To summarize, first an ellipsoid tangent contour that accounts for intrinsic curvature can be constructed. This construction does not require assumptions with respect to parameter-effects curvature. Next, the associated likelihood contour in parameter space is determined by means of a linear approximation. The extent to which this linear approximation is valid is quantified by the parameter-




(a) Intrinsic curvature (b) Parameter-effects curvature

Figure 4.3: Distortion of likelihood contours due to two types of curvature. In (a) the dashed contour, called the expectation contour, indicates points on the expectation manifold with a certain distance to the realized observation vector $y$. Due to intrinsic curvature the expectation manifold deviates from the tangent space at $f(\hat{\theta})$. The solid contour, called the tangent contour, is the projection of the expectation contour on the tangent space. In (b) the relation between the tangent contour and the likelihood contour is depicted. The solid contour corresponds to a case with no parameter-effects curvature, the dashed contour to a case with parameter-effects curvature.

effects curvature. If this curvature is large the expectation manifold must first be reparameterized.

In Donaldson and Schnabel (1987) it is shown that model nonlinearity also complicates the construction of confidence or likelihood intervals for individual parameters. In Cook and Weisberg (1990) an alternative is presented for the standard Wald confidence intervals; Clarke (1987a) discusses the computation of such intervals. This method, however, requires reparameterization of the expectation manifold. As the choice of a suitable reparameterization is difficult for large models, we will use an efficient method by Clarke (1987b) based on marginal curvature, that does not rely on reparameterization. This method and its application will be discussed in more detail in Chapter 5.

### 4.4.4 Curvature arrays

Curvature arrays, first introduced in Bates and Watts (1980), are used in most methods aimed at the construction of accurate inferential information in the presence on nonlinearity. To obtain expressions for the curvature arrays we must restructure $f$ into a function $\tilde{f}$ such that its tangent plane at the maximum likelihood estimate aligns with the first $p$ coordinate axes of a rotated sample space. The projections of the second order derivative of $\tilde{f}$ on the tangent plane and its orthogonal complement will be the parameter-effects and intrinsic curvature arrays, respectively.

First, a QR decomposition provides a basis for both the tangent space and its

orthogonal complement at $f(\theta_0)$. We use

$$(D^1 f(\theta_0)) = QR = \begin{pmatrix} Q_t & Q_c \end{pmatrix} \begin{pmatrix} R_t \\ 0 \end{pmatrix}, \tag{4.42}$$

where the columns of the $n \times p$ matrix $Q_t$ are the basis vectors of the tangent space, and the columns of the $n \times (n-p)$ matrix $Q_c$ are the basis vectors of the orthogonal complement of the tangent space. The $p \times p$ matrix $R_t$ is upper-triangular and non-singular.

We then use this decomposition to define the function $\tilde{f}$ by

$$\tilde{f}(\phi) = Q' f(K\phi), \tag{4.43}$$

where

$$K = R_t^{-1}. \tag{4.44}$$

The relation between $f(\theta)$ and $\tilde{f}(\phi)$ is shown in Figure 4.4.

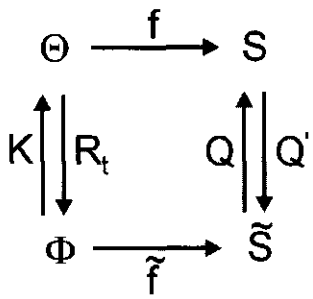



Figure 4.4: Structure of the reparameterization and rotation of the expectation manifold.

As $Q'$ is the inverse of $Q$, multiplication by $Q'$ can be interpreted as a rotation by which the basis vectors of the tangent space are mapped to the first $p$ unit vectors and the basis vectors of the orthogonal complement of the tangent space are mapped to the last $n-p$ basis vectors of the transformed sample space $\tilde{S}$ (see Figure 4.4).

The derivative of $\tilde{f}$ at $\phi_0 = R_t \theta_0$ is given by

$$(D^1 \tilde{f})(\phi_0) = Q'(D^1 f)(\theta_0)K = Q'(Q_t R_t)K = Q'Q_t = \begin{pmatrix} I_p \\ 0_{(n-p)\times p} \end{pmatrix}. \tag{4.45}$$

This means that, in $\tilde{S}$, projection on the tangent space consists of taking the first $p$ components and setting the remaining components to zero. For the second order derivative, we find

$$D^2_{k_1 k_2} \tilde{f}_i = \sum_{i_1, j_1, j_2} Q_{i_1 i} (D^2_{j_1 j_2} f_{i_1}) K_{j_1 k_1} K_{j_2 k_2}. \tag{4.46}$$

Here we have used the component notation described in Appendix A; the components correspond to the second derivatives of $\tilde{f}_i$ with respect to $\phi_{k_1}$ and $\phi_{k_2}$, where $i = 1, \ldots, n$ and $k_1, k_2 = 1, \ldots, p$. The second order derivatives of $\tilde{f}$ projected on the tangent space and its orthogonal complement are called the parameter-effects curvature array $A^T$ and intrinsic curvature array $A^N$, respectively. We thus have

$$A^T_{ik_1k_2} = (D^2_{k_1k_2}\tilde{f}_i)^T = \sum_{i_1,j_1,j_2} Q_{ti_1i}(D^2_{j_1j_2}f_{i_1})K_{j_1k_1}K_{j_2k_2}, \tag{4.47}$$

and,

$$A^N_{ik_1k_2} = (D^2_{k_1k_2}\tilde{f}_i)^N = \sum_{i_1,j_1,j_2} Q_{ci_1i}(D^2_{j_1j_2}f_{i_1})K_{j_1k_1}K_{j_2k_2}. \tag{4.48}$$

using $Q_t$ and $Q_c$ defined in (4.42). Higher order curvature arrays are similarly defined as projections of the higher order derivatives of $\tilde{f}$. For instance, we will use

$$A^T_{ik_1k_2k_3} = (D^3_{k_1k_2k_3}\tilde{f}_i)^T = \sum_{i_1,j_1,j_2,j_3} Q_{ti_1i}(D^3_{j_1j_2j_3}f_{i_1})K_{j_1k_1}K_{j_2k_2}K_{j_3k_3}. \tag{4.49}$$

The order of the array can be identified by the number of indices.

The curvature measures can be expressed in terms of the curvature arrays. As the curvature measures do not depend on the length of the direction vector, we may use $\delta\theta = K\delta\phi$ with $\|\delta\phi\| = 1$. We then have, using invariance of vector norms under orthogonal transformations, that

$$\begin{aligned} K^T_{\delta\theta} &= \frac{\|(D^2f)^T\delta\theta^2\|}{\|(D^1f)\delta\theta\|^2} = \frac{\|Q_tQ_t'\,D^2f\,(K\delta\phi)^2\|}{\|(D^1f)K\delta\phi\|^2} \\ &= \frac{\|Q_tA^T(\delta\phi)^2\|}{\|\delta\phi\|} = \|A^T\delta\phi^2\|, \end{aligned} \tag{4.50}$$

and, analogously,

$$K^N_{\delta\theta} = \|A^N\delta\phi^2\|. \tag{4.51}$$

The left and right multiplication operations for higher order arrays are defined in Appendix A.

The curvature arrays may be computed directly from (4.47) and (4.48). This requires the computation of the first and second order derivatives of $f$, a QR-decomposition of the first derivative, and computation of the inverse of $R_t$. In Bates et al. (1983) an algorithm is described that efficiently computes the two curvature arrays using a symmetric storage scheme. Draper and Smith (1981) discuss a geometric interpretation of the individual coefficients of the parameter-effects curvature array.

### 4.4.5 Moments of maximum likelihood estimators

We consider the computation of the moments of the maximum likelihood estimator for a normal nonlinear regression model, given by (4.24), for the case that the true

density function is a model element referenced by $\theta = \theta^*$ and $\sigma = \sigma^*$. We thus use

$$Y = f(\theta^*) + \varepsilon, \qquad \varepsilon \sim \mathrm{N}(0, \sigma^{*2} I_n). \tag{4.52}$$

for *known* $\theta^*$ and $\sigma^*$.

The maximum likelihood estimator $\hat{\theta}$ is a random variable depending on $\varepsilon$. In the following we derive two properties of its distribution. Similar derivations are given in Clarke (1980). First the *estimation bias*

$$\mathrm{E}[\hat{\beta}] = \mathrm{E}[\hat{\theta} - \theta^*] = \mathrm{E}[\hat{\theta}] - \theta^*, \tag{4.53}$$

and second the *estimation variance-covariance*

$$\mathrm{var}(\hat{\theta}, \hat{\theta}) = \mathrm{E}[(\hat{\theta} - \mathrm{E}\hat{\theta})(\hat{\theta} - \mathrm{E}\hat{\theta})']. \tag{4.54}$$

The mean square error (MSE) is given by

$$\mathrm{MSE} = \mathrm{E}[(\hat{\theta} - \theta^*)'(\hat{\theta} - \theta^*)] = \mathrm{tr}(\mathrm{var}(\hat{\theta}, \hat{\theta})) + \mathrm{E}[\hat{\beta}]'\mathrm{E}[\hat{\beta}]. \tag{4.55}$$

The estimation bias, variance-covariance and MSE may be used in the design stage of a statistical setup. For example, the relation between the number of observations and the MSE can be investigated by simulation, to determine a setup that is expected to be sufficiently accurate. If one assumes that the density associated with a certain parameter value corresponds to the true density of the observations, then one can compute the bias, variance-covariance and MSE of the maximum likelihood estimator. Note that these quantities can be computed for any values of the parameters. This is illustrated by a simple example in Figure 4.5. In the figure the likelihood function for a particular realization is shown as well. The properties of the estimator distribution are also often computed after data have been collected, to gain an idea of the accuracy of the estimator based on the assumption that the density corresponding to the estimated parameters is the true density of the observations.

### Expansion of the maximum likelihood estimator in residuals

To compute the estimation bias and variance-covariance we want to approximate the relation between the maximum likelihood estimator $\hat{\theta}$ and $\varepsilon$. Since the aim is to express the corrections to the linear case in terms of the intrinsic and parameter-effects curvature arrays, we again use reparameterization of the parameter space and rotation of sample space as described by (4.43) in section 4.4.4.

We thus take

$$\theta = K\phi, \quad \text{or} \quad \theta_j = \sum_k K_{jk}\phi_k, \tag{4.56}$$

and consider the rotated residual vector $\tilde{\varepsilon} = Q'\varepsilon$. Note that the rotated residuals have the same multivariate normal distribution as the original residuals. Further, recall that multiplication by $Q'$ represents a rotation by which the basis vectors

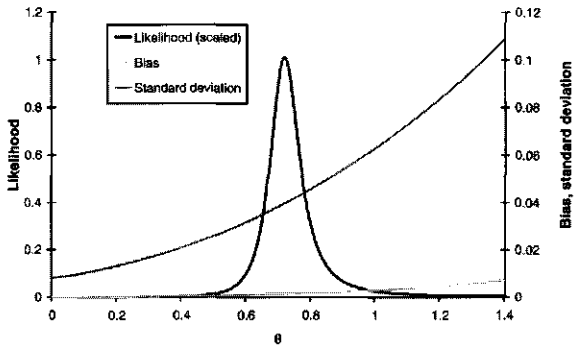



Figure 4.5: Likelihood, bias and standard deviation functions for a simple nonlinear regression problem. Model: $y_i = 10\exp(-x_i\theta) + \varepsilon_i, \varepsilon_i \sim N(0,\sigma^2), i = 1,2,3$. Observed values: $x_1 = 1, y_1 = 5; x_2 = 2, y_2 = 2; x_3 = 3, y_3 = 1.5$. All graphs are plotted for a value of $\sigma = \hat{\sigma}$, where $\hat{\sigma}$ is the maximum likelihood estimate corresponding to the observed values. Note however that the graphs can be obtained for any value of $\sigma > 0$.

of the tangent space are mapped to the first $p$ unit vectors and the basis vectors of the orthogonal complement of the tangent space are mapped to the last $n-p$ basis vectors of the transformed sample space $\tilde{S}$. We split the rotated residuals into two parts, one corresponding to the rotated tangent space and one to its rotated orthogonal complement

$$\tilde{\varepsilon} = \begin{pmatrix} \varepsilon_t \\ \varepsilon_c \end{pmatrix} = \begin{pmatrix} Q_t'\varepsilon \\ Q_c'\varepsilon \end{pmatrix}. \tag{4.57}$$

The bias of the maximum likelihood estimator $\hat{\phi}$ is expressed as a power series in $\varepsilon_t$ and $\varepsilon_c$. The bias is denoted by

$$\mathrm{E}[\hat{\gamma}] = \mathrm{E}[\hat{\phi} - \phi^*] = \mathrm{E}[\hat{\phi}] - \phi^*, \tag{4.58}$$

where $\phi^* = K^{-1}\theta^*$.

If we take the power series expansion of $\hat{\gamma}$ in $\varepsilon_t$ and $\varepsilon_c$ given by

$$\begin{aligned}\hat{\gamma}_k = &\sum_{i_1}(c^{10}_{ki_1}\varepsilon_{ti_1} + c^{01}_{ki_1}\varepsilon_{ci_1}) + \sum_{i_1,i_2}(c^{20}_{ki_1i_2}\varepsilon_{ti_1}\varepsilon_{ti_2} + c^{11}_{ki_1i_2}\varepsilon_{ti_1}\varepsilon_{ci_2} \\ &+ c^{02}_{ki_1i_2}\varepsilon_{ci_1}\varepsilon_{ci_2}) + \sum_{i_1,i_2,i_3}(c^{30}_{ki_1i_2i_3}\varepsilon_{ti_1}\varepsilon_{ti_2}\varepsilon_{ti_3} + c^{21}_{ki_1i_2i_3}\varepsilon_{ti_1}\varepsilon_{ti_2}\varepsilon_{ci_3} \\ &+ c^{12}_{ki_1i_2i_3}\varepsilon_{ti_1}\varepsilon_{ci_2}\varepsilon_{ci_3} + c^{03}_{ki_1i_2i_3}\varepsilon_{ci_1}\varepsilon_{ci_2}\varepsilon_{ci_3}) + \mathcal{O}(|\varepsilon|^4),\end{aligned} \tag{4.59}$$

then the coefficients can be expressed in terms of the intrinsic and parameter-effects curvature arrays, see Table 4.1. All arrays are evaluated at the known true values of the parameters. The expressions for the power series coefficients are derived in Appendix F.

| | |
|---|---|
| $c^{10}_{ki_1}$ | $\delta_{ki_1}$ |
| $c^{01}_{ki_1}$ | $0$ |
| $c^{20}_{ki_1i_2}$ | $-\frac{1}{2}A^T_{ki_1i_2}$ |
| $c^{11}_{ki_1i_2}$ | $A^N_{i_2ki_1}$ |
| $c^{02}_{ki_1i_2}$ | $0$ |
| $c^{30}_{ki_1i_2i_3}$ | $\sum_{k_1}\left[\frac{1}{2}A^T_{ki_1k_1}A^T_{k_1i_2i_3}-\frac{1}{2}A^N_{k_1i_1i_2}A^N_{k_1ki_3}\right]-\frac{1}{6}A^T_{ki_1i_2i_3}$ |
| $c^{21}_{ki_1i_2i_3}$ | $\sum_{k_1}\left[-A^T_{ki_1k_1}A^N_{i_3k_1i_2}-\frac{1}{2}A^T_{k_1i_1i_2}A^N_{i_3kk_1}-A^T_{k_1ki_1}A^N_{i_3k_1i_2}\right]$ $-\frac{1}{6}A^T_{ki_1i_2i_3}+\frac{1}{2}A^N_{i_3ki_1i_2}$ |
| $c^{12}_{ki_1i_2i_3}$ | $\sum_{k_1}A^N_{i_2k_1i_1}A^N_{i_3kk_1}$ |
| $c^{03}_{ki_1i_2i_3}$ | $0$ |

Table 4.1: Expressions for $c^{10}_{ki_1},\ldots,c^{03}_{ki_1i_2i_3}$.

### Bias of the maximum likelihood estimator

Since $\varepsilon_t$ and $\varepsilon_c$ are components of the rotated residuals they have a multivariate normal distribution with independent components of zero mean and variance $\sigma^{*2}$. The multivariate normal distribution is symmetric about its mean. Therefore, its odd central moments are equal to zero. By taking the expectation of the power series (4.59) for $\hat{\gamma}$, we thus find

$$\mathrm{E}[\hat{\gamma}_k]=\sum_{i_1,i_2}(c^{20}_{ki_1i_2}\mathrm{E}[\varepsilon_{ti_1}\varepsilon_{ti_2}]+c^{11}_{ki_1i_2}\mathrm{E}[\varepsilon_{ti_1}\varepsilon_{ci_2}]), \tag{4.60}$$

and from the distribution of the rotated residuals we have

$$\mathrm{E}[\varepsilon_{ti_1}\varepsilon_{ti_2}]=\sigma^{*2}\delta_{i_1i_2}\quad\text{and}\quad\mathrm{E}[\varepsilon_{ti_1}\varepsilon_{ci_2}]=0, \tag{4.61}$$

where $\delta_{i_1i_2}=1$, if $i_1=i_2$, and $\delta_{i_1i_2}=0$ otherwise. So, using Table 4.1, we find

$$\mathrm{E}[\hat{\gamma}_k]=-\frac{1}{2}\sigma^{*2}\sum_{i_1}A^T_{ki_1i_1}. \tag{4.62}$$

From (4.56) it then follows that bias in the original parameters is given by

$$\mathrm{E}[\hat{\beta}_j]=E[\hat{\theta}_j]-\theta^*_j=\sum_k K_{jk}\mathrm{E}[\hat{\gamma}_k]=-\frac{1}{2}\sigma^{*2}\sum_{k,i_1}K_{jk}A^T_{ki_1i_1}. \tag{4.63}$$

Note that the bias only depends on the parameter-effects curvature.

### Variance-covariance of the maximum likelihood estimator

The variance-covariance of the maximum likelihood estimator follows from

$$\begin{aligned}\operatorname{cov}(\hat{\theta}_{j_1}, \hat{\theta}_{j_2}) &= \mathrm{E}[\hat{\theta}_{j_1}\hat{\theta}_{j_2}] - \mathrm{E}\hat{\theta}_{j_1}\mathrm{E}\hat{\theta}_{j_2} \\ &= \sum_{k_1,k_2} K_{j_1k_1}K_{j_2k_2}(\mathbf{E}[\hat{\gamma}_{k_1}\hat{\gamma}_{k_2}] - \mathrm{E}\hat{\gamma}_{k_1}\mathrm{E}\hat{\gamma}_{k_2}).\end{aligned} \tag{4.64}$$

Considering terms up to fourth order, we first find

$$\mathrm{E}\hat{\gamma}_{k_1}\mathrm{E}\hat{\gamma}_{k_2} = \frac{1}{4}\sigma^{*4}\sum_{i_1,i_2} A^T_{k_1i_1i_1}A^T_{k_2i_2i_2}. \tag{4.65}$$

by using the expression (4.62) derived for the estimation bias. To obtain $\mathrm{E}[\hat{\gamma}_{k_1}\hat{\gamma}_{k_2}]$ first the power series expansions for $\hat{\gamma}_{k_1}$ and $\hat{\gamma}_{k_2}$ from (4.59) are multiplied and again only terms to order four are retained. Then taking the expectation term by term and using the results of Chapter 3 (section 3.4.2) to obtain the fourth order moments of the residuals, leads to the approximation for $\mathrm{E}[\hat{\gamma}_{k_1}\hat{\gamma}_{k_2}]$. The resulting variance-covariance can be expressed as a sum of the linear approximation to the variance-covariance and three additional correction terms due to model nonlinearity

$$\operatorname{cov}(\hat{\theta}_{j_1}, \hat{\theta}_{j_2}) = \sum_{k_1,k_2}\left[K_{j_1k_1}K_{j_2k_2}(\sigma^{*2}\delta_{k_1k_2} + \sigma^{*4}(C^1_{k_1k_2} + C^2_{k_1k_2} + C^3_{k_1k_2}))\right], \tag{4.66}$$

where

$$\begin{aligned}C^1_{k_1k_2} = \sum_{i_1,i_2}(&\frac{1}{2}A^T_{k_1i_1i_2}A^T_{k_2i_2i_1} + A^T_{i_2i_1k_1}A^T_{k_2i_1i_2} + A^T_{k_1i_1i_2}A^T_{i_2i_1k_2} \\ &+ \frac{1}{2}A^T_{k_2k_1i_2}A^T_{i_2i_1i_1} + \frac{1}{2}A^T_{k_1k_2i_2}A^T_{i_2i_1i_1}),\end{aligned} \tag{4.67}$$

$$C^2_{k_1k_2} = \sum_{i_1,i_2}(A^N_{i_2k_1i_1}A^N_{i_2k_2i_1} - A^N_{i_2i_1i_1}A^N_{i_2k_1k_2}), \tag{4.68}$$

$$C^3_{k_1k_2} = -\frac{1}{2}\sum_{i_1}(A^T_{k_1k_2i_1i_1} + A^T_{k_2k_1i_1i_1}). \tag{4.69}$$

Since $((D^1f)'(D^1f))^{-1} = (R_t'Q_t'Q_tR_t)^{-1} = (R_t'R_t)^{-1} = KK'$, it follows that the first term in (4.66) indeed corresponds to the variance-covariance matrix based on a linear approximation of the expectation manifold. Also note that the $C^2$ term is invariant under reparameterizations, so this term cannot be reduced by reparameterization.

### Maximum likelihood estimator moments under model misspecification

We present a method to approximate the moments of the maximum likelihood estimator in the case of model misspecification. It provides a generalization of the

method discussed in the previous section for the case that the probability density function of the observation vector is not an element of the model family.

An overview of maximum likelihood estimation for misspecified models is given in White (1982). Here several asymptotic results are presented. More general studies on the influence of model specification on statistical inference can be found in Cook (1986) and Gustafson (2001).

The method is conceptually simple: first a Taylor series expansion of the estimator with respect to the residuals is constructed by means of the implicit function theorem, and next the moments of the distribution of the residuals are propagated through this expression. Both elements of this approach were already described in Chapter 3.

Suppose that the true density function has expectation $\mu = \mathrm{E}[Y]$ and that $Y = \mu + \epsilon$, where $\varepsilon$ has expectation zero. We use the fact that the maximum likelihood estimate $\hat{\theta}(\mu)$ corresponding to the realization $y = \mu$ minimizes $\|\mu - f(\theta)\|^2$ and satisfies

$$\frac{\partial}{\partial\theta}\|y - f(\theta)\|^2_{\theta=\hat{\theta}(\mu),y=\mu} = \frac{\partial}{\partial\theta}S(\theta,y)_{\theta=\hat{\theta}(\mu),y=\mu} = 0, \tag{4.70}$$

where we have taken $S(\theta,y) = \|y - f(\theta)\|^2$. A Taylor expansion of $h(y) = \hat{\theta}(y)$ with respect to $\varepsilon$ now follows from the implicit function theorem applied to (4.70) around $(\hat{\theta}(\mu),\mu)$. Using the results from equations (3.60) and (3.61) (Chapter 3, section 3.6), we have

$$\begin{aligned}\hat{\theta}_i(y) = h_i(y) = \hat{\theta}_i(\mu) &+ \sum_{k_1=1}^{n}\frac{\partial h_i}{\partial y_{k_1}}\varepsilon_{k_1} \\ &+ \frac{1}{2}\sum_{k_1,k_2=1}^{n}\frac{\partial^2 h_i}{\partial y_{k_1}\partial y_{k_2}}\varepsilon_{k_1}\varepsilon_{k_2} + \mathcal{O}(|\varepsilon|^3),\end{aligned} \tag{4.71}$$

where $\varepsilon = y - \mathrm{E}[Y] = y - \mu$, and

$$\sum_{i=1}^{p}\frac{\partial^2 S}{\partial\theta_i\partial\theta_j}\frac{\partial h_i}{\partial y_{k_1}} = -\frac{\partial^2 S}{\partial\theta_j\partial y_{k_1}}, \tag{4.72}$$

and,

$$\sum_{i=1}^{p}\frac{\partial^2 S}{\partial\theta_i\partial\theta_j}\frac{\partial^2 h_i}{\partial y_{k_1}\partial y_{k_2}} = -\Big[\frac{\partial^3 S}{\partial\theta_j\partial y_{k_1}\partial y_{k_2}} + \sum_{i=1}^{p}\frac{\partial^3 S}{\partial\theta_j\partial\theta_i\partial y_{k_1}}\frac{\partial h_i}{\partial y_{k_2}}\Big]. \tag{4.73}$$

Here and further on, derivatives with respect to $\theta$ are evaluated in $\hat{\theta}(\mu)$, derivatives with respect to $y$ are evaluated in $\mu$. Note that this procedure can also be used for general likelihood or log-likelihood functions as these functions satisfy expressions similar to (4.70).

The derivatives of $S$ can be expressed in terms of derivatives of $f$ by

$$\frac{\partial^2 S}{\partial\theta_i\partial\theta_j} = 2\sum_{l=1}^{n}((D_i^1 f_l)(D_j^1 f_l) - (\mu_l - f_l(\hat{\theta}(\mu)))D_{ij}^2 f_l), \tag{4.74}$$

$$\frac{\partial^2 S}{\partial\theta_i\partial y_k} = -2(D_i^1 f_k), \tag{4.75}$$

$$\frac{\partial^3 S}{\partial\theta_i\partial y_{k_1}\partial y_{k_2}} = 0, \tag{4.76}$$

$$\frac{\partial^3 S}{\partial\theta_i\partial\theta_j\partial y_k} = -2(D_{ij}^2 f_k). \tag{4.77}$$

The second derivative of $S$ with respect to the parameters, given by (4.74), plays an important role in the entire procedure, since its inverse is required in the computation of the derivatives of $h$, as can be seen from (4.72) and (4.73). If this derivative is nonsingular, we can compute the derivatives of $h$, and construct a Taylor expansion for $\hat{\theta}$ in terms of $\varepsilon$ from (4.71). If $\varepsilon$ has a multivariate normal distribution, the expectation and variance-covariance can then be obtained from the propagation formulas (3.15) and (3.24) (section 3.4.2).

If the true density function is an element of the model family given by (4.24) with parameters $\theta = \theta^*$, then $\mu = f(\theta^*)$ and $\hat{\theta}(\mu) = \theta^*$, because $\mu$ lies on the expectation manifold. In this case the second derivative of $S$ with respect to the parameters reduces to $(D^1 f)'(D^1 f)$, because the term $(\mu - f(\hat{\theta}(\mu)))$ vanishes. We note that if the above procedure is applied to $\tilde{S}(\phi, \tilde{y}) = \|\tilde{y} - \tilde{f}(\phi)\|^2$, where $\tilde{y} = Q'y$, for the case that the true density function is an element of the model family, the expressions for the moments of the maximum likelihood estimator can be constructed in terms of the curvature arrays.

We conclude that the method described here can be used for the computation of accurate approximations for estimation bias and estimation variance-covariance, for multivariate normal density functions of the observation vector both inside and outside the model family.

## 4.5 Concluding remarks

In this chapter we have presented a detailed treatment of methods for statistical inference for nonlinear regression models. We have done so with the aim of providing the statistical background for the development of a system for the analysis of parameter estimation procedures by derivative-based methods.

The methods discussed allow for a thorough investigation of the effects of nonlinearity on statistical inference. The likelihood theory presented in the first part of the chapter provides the various methods with a clear interpretation. This theory also leads to the interpretation of model selection criteria as representing a rate of exchange between data support and model simplicity, thus providing a tool for the investigation of the appropriateness of model structure in relation to data

quality. The Akaike and corrected Akaike criteria are useful in this respect if models are only mildly misspecified. This does not have to pose a serious restriction, since large misspecification can usually be screened out by diagnostic checks, for instance by the investigation of the estimated residuals.

The methods discussed generally require only the computation of the maximum likelihood estimate and first and higher-order derivatives of model functions at this estimate. No additional evaluations of the model functions for different values of the parameters or other variables are needed. Computation of the required derivatives by means of automatic differentiation increases the feasibility of using nonlinear statistical inference for models with a large number of parameters. An implementation is provided by the MAP software library which will be described in Chapter 7. Using model functions supplied by the user, the library handles the computation of the inferential and model selection information.

The practical use of the theory presented here will be demonstrated in Chapter 5 in a case study on the stock assessment of North Sea herring.

# Appendix A – Notation

Along the usual notation for matrix and vector operations, we often use a notation in which equalities are written out by component. This notation generalizes to arbitrary order, facilitates the use of the product and chain rules of differentiation, and also leads to expressions that are easy to implement in computer code. The components are referenced by subscripts. Index ranges and summation indices are usually not indicated as they easily follow from the dimension of the variables they apply to.

For derivatives we use the notations $D^1 f$, $D^2 f$ to denote the full derivative arrays

$$(D^1 f)_{ij} = \frac{\partial f_i}{\partial \theta_j}, \quad \text{and} \quad (D^2 f)_{ij_1 j_2} = \frac{\partial f_i}{\partial \theta_{j_1} \partial \theta_{j_2}}. \tag{4.78}$$

Higher order derivative arrays are defined in a similar manner. For the components we also use $D^1_j f_i$ and $D^2_{j_1 j_2} f_i$. For arrays $A$ with more than two indices, two further operations are defined. Left multiplication of $A$ by a matrix B is defined by

$$(BA)_{jkl} = \sum_i B_{ji} A_{ikl}, \tag{4.79}$$

i.e. as a multiplication by summation over the first index of the array. This operation corresponds to the square-bracket multiplication defined in Bates and Watts (1980). Right multiplication with vectors $c_1$ through $c_n$ is defined by

$$(Ac_1 \dots c_n)_i = \sum_{i_1,\dots,i_n} A_{ii_1\dots i_n} c_{1i_1} \cdots c_{ni_n}. \tag{4.80}$$

We use only arrays that are symmetric with respect to all indices except the first one, e.g. we always have $A_{ij_1j_2} = A_{ij_2j_1}$. This means that right multiplication

does not depend on the order of the vectors $c_1, \ldots, c_n$. Multiplication by less than $n$ vectors naturally leads to product arrays of more than one index. We will further use $Ac^2$ for $Acc$.

# Appendix B – Entropy and Kullback-Leibler discrepancy

Discrepancies are used to quantify the dissimilarity between probability distributions. A well-known and generally applicable discrepancy, based on the likelihood ratio, is the Kullback-Leibler discrepancy. For a general discussion of discrepancies we refer to Linhart and Zucchini (1986).

Following Cover and Thomas (1991), we will introduce the Kullback-Leibler discrepancy, also referred to as Kullback-Leibler information, here as a relative entropy and provide some background on the connection between its definition and information theory. We refer to Soofi (2000) for a general overview of information-theoretic methods in statistics.

The *entropy* of a discrete random variable $X$ is defined by

$$H(X) = -\sum_{x \in \mathcal{X}} p(x) \log p(x) = E_p \log \frac{1}{p(X)}, \tag{4.81}$$

where $\mathcal{X}$ is the set of possible outcomes of $X$. For a logarithm of base 2, the entropy is expressed in bits. The entropy of a random variable is a measure of the uncertainty associated with the random variable. One can show, for instance, that the entropy is a lower bound on the average length (in bits) of the shortest description of the outcomes of the random variable. Shannon and Huffman codes provide descriptions with an average length that is within one bit of the entropy. These matters are discussed in more detail in Cover and Thomas (1991, chapter 5).

For a fixed number of possible outcomes, the entropy is highest for the uniform distribution, i.e. we have $H(X) \leq \log |\mathcal{X}|$, with equality if and only if $X$ has a uniform distribution over $\mathcal{X}$. Here $|\mathcal{X}|$ denotes the number of possible outcomes of $X$.

The *relative entropy* or *Kullback-Leibler information* $I(p, q)$ of the probability mass function $p$ with respect to the probability mass function $q$ is defined by

$$I(p, q) = \sum_x p(x) \log \frac{p(x)}{q(x)} = E_p \log \frac{p(X)}{q(X)}. \tag{4.82}$$

It can be shown that $I(p, q) \geq 0$ with equality if and only if $p(x) = q(x)$ for all $x \in \mathcal{X}$. The relative entropy is a measure of the distance between two distributions. It is not a distance in the usual mathematical sense of a metric, but rather a measure of the inefficiency of assuming that the mass function is $q$ when the true mass function is $p$. This can be illustrated by means of a Shannon code. For such a code every outcome of $X$ has an associated codeword length of $\lceil \log \frac{1}{p_i} \rceil$,

where $\lceil x \rceil$ is the smallest integer $\geq x$. This means that outcomes with high probability are assigned short codewords, and outcomes with low probability long codewords. A Shannon code for $p(x)$ provides an average description length within one bit of $H(p)$. If a code associated with another mass function $q(x)$ is used for outcomes of the true distribution, the expected description length will increase as for instance the outcomes of highest probability may no longer be corresponding to the codewords of shortest length. It can be shown that the average description length will be within one bit of $H(p) + I(p, q)$. This thus provides an insightful interpretation of the Kullback-Leibler discrepancy: it is not a distance measure between two arbitrary mass function, but of mass functions $q(x)$ relative to one 'true' mass function $p(x)$ based on the inefficiency of using $q(x)$ instead of $p(x)$.

The *differential entropy* $h(X)$ of a continuous random variable $X$ with density function $f(x)$ is defined as

$$h(X) = -\int_S f(x) \log f(x)\, dx, \tag{4.83}$$

where $S$ is the support set of the random variable, i.e. the set where $f(x) > 0$.

The so-called asymptotic equipartition property provides an interpretation of differential entropy. The differential entropy is the logarithm of the equivalent side length of the smallest set that contains most of the probability. We refer to Cover and Thomas (1991, chapter 9) for the precise definitions of the terms in this statement. The differential entropy is thus a measure for the dispersion of the random variable: low entropy implies that the random variable is confined to a small effective volume and high entropy indicates that the random variable is widely dispersed. Since the differential entropy is the logarithm of a length it may also take negative values. The differential entropy satisfies $h(X + c) = h(X)$, i.e. translation does not change the differential entropy, and $h(cX) = h(X) + \log(|c|)$ for all $c$.

Further insight in the differential entropy can be gained by considering quantized random variables of $X$. A quantized random variable $X^\Delta$ is obtained by dividing the range of $X$ into bins of length (or volume for random vectors) $\Delta$ . Within each bin there exists a value $x_i$ such that

$$f(x_i)\Delta = \int_{i\Delta}^{(i+1)\Delta} f(x)\, dx. \tag{4.84}$$

The quantized random variable is then defined as taking values $x_i$ with probability $p_i = f(x_i)\Delta$. The entropy of the quantized random variable is equal to

$$H(X^\Delta) = -\sum_{-\infty}^{\infty} p_i \log p_i = \log \Delta - \sum_{-\infty}^{\infty} \Delta f(x_i) \log f(x_i). \tag{4.85}$$

It follows that if the density $f(x)$ is Riemann integrable, then

$$H(X^\Delta) + \log \Delta \to h(X), \quad \text{for } \Delta \to 0. \tag{4.86}$$

This means that the $n$-bit quantization of a continuous variable has an entropy of approximately $h(X) + n$

As a final result we mention that the normal density maximizes the entropy over all densities with the same covariance, i.e. if $X$ is a random vector taking values from $\mathbb{R}^n$ with zero mean and covariance $K = \mathrm{E}[XX']$, then $h(X) \leq \frac{1}{2}\log(2\pi e)^n|K|$. Equality occurs if and only if $X \sim N(0, K)$.

*Relative entropy* or *Kullback-Leibler discrepancy* between two densities $f$ and $g$ is now defined by

$$I(f,g) = \int f(x) \log \frac{f(x)}{g(x)}\, dx. \tag{4.87}$$

Note that $I(f,g)$ is finite only if the support set of $f$ is contained in the support set of $g$. We have $I(f,g) \geq 0$ with equality if and only if $f = g$ (almost everywhere (a.e.)). The relative entropy again provides a measure for the difference in average description length when a code for a density function $g$ is used for outcomes with density $f$.

# Appendix C – Nonlinear regression problems

As a model for $n_1$ observations $(x_i, y_i), i = 1, \ldots, n_1$ we first consider,

$$y_i = f(x_i, \theta) + \varepsilon_i, \quad i = 1, \ldots, n_1, \tag{4.88}$$

where the $y_i$, $f(x_i, \theta)$ and $\varepsilon_i$ are all vectors with $n_2$ elements and the $\varepsilon_i = (\varepsilon_{i1}', \ldots, \varepsilon_{in_2}')'$ are usually assumed to be i.i.d $N_{n_2}(0, \Sigma)$, i.e. are assumed to be independently and identically distributed with multivariate normal distribution with zero expectation and variance-covariance matrix $\Sigma$. The vectors $x_i$ are regressor vectors, representing variables that are used to describe the relation between the observations $y$ and the parameters $\theta$, and which are assumed to be known. Written out in components we have

$$y_{ij} = f_j(x_i, \theta) + \varepsilon_{ij}, \quad i = 1, \ldots, n_1, \quad j = 1, \ldots, n_2. \tag{4.89}$$

The vectors $y_i, i = 1, \ldots, n_1$, can be grouped in one large vector $y = (y_1', \ldots, y_{n_1}')'$. We then have

$$y = f(\mathcal{D}, \theta) + \varepsilon, \quad \varepsilon \sim N_n(0, V), \tag{4.90}$$

where $n = n_1 n_2$, and

$$f(\mathcal{D}, \theta) = \Big(f_1(x_1, \theta), \ldots, f_{n_2}(x_1, \theta), \ldots, f_1(x_{n_1}, \theta), \ldots, f_{n_2}(x_{n_1}, \theta)\Big)'. \tag{4.91}$$

The random vector $\varepsilon$ is defined in a manner analagous to $y$, and $V$ is the variance-covariance matrix characterizing the multivariate normal distribution of $\varepsilon$. In this chapter the dependence of $f$ on the design $\mathcal{D} = (x_1', \ldots, x_{n_1}')'$ is suppressed as it is assumed to be fixed. For probability models with a random design we refer to Stefanski (2000) and Fuller (1987).

Usually it is assumed that the covariance structure is fixed apart from an unknown scaling factor $\sigma$. It then follows that the data and approximating random vectors can be transformed into

$$y = f(\mathcal{D}, \theta) + \varepsilon, \quad \varepsilon \sim N_n(0, \sigma^2 I_n). \tag{4.92}$$

A transformation to this effect, based on a Cholesky decomposition of the variance-covariance matrix, is discussed in Appendix A of Chapter 5. Problems where the entire covariance matrix must be estimated are discussed in Seber and Wild (1989, chapter 11). The family of density functions associated with the random vectors of (4.92) constitutes the model we will use for the nonlinear regression problems of section 4.4.

## Appendix D – Construction of likelihood contours

In Hamilton et al. (1982) coordinates $\tau$ are introduced by $\tau = Q_t'(f(\theta) - f(\hat{\theta}))$. The vector $Q_t\tau$ is then the projection of $f(\theta) - f(\hat{\theta})$ on the tangent space. By using $\tau$ to reparameterize the expectation manifold, it is shown that the likelihood contour can be approximated by $\tau'(I_p - B)\tau = \delta^2$ where $B$ depends on the intrinsic curvature. This means that the tangent contour is an ellipsoid in tangent space.

If parameter-effects curvature is small, the determination of the likelihood contour in parameter space is straightforward. We use $t = Q_t\tau$, so if $t$ is in the contour, we have $Q_t't = \tau$, giving $t'Q_t(I_p - B)Q_t't = \delta^2$. If the parameter-effects curvature is small, we can use $t = D^1 f(\theta - \hat{\theta})$, so

$$(\theta - \hat{\theta})' R_t'(I_p - B)R_t(\theta - \hat{\theta}) = \delta^2. \tag{4.93}$$

## Appendix E – Transformations for the reduction of parameter-effects curvature

The methods for obtaining simultaneous inference regions discussed in section 4.4.3 require the parameter-effects curvature to be sufficiently small. Here we discuss reparameterizations to reduce this parameter-effects curvature. The computational procedure to construct confidence intervals or (profile) likelihood contours of individual parameters described in Clarke (1987a), also relies on reparameterizations of the type discussed here.

Consider a reparameterization in which the new parameters $\psi$ are a differentiable transformation of the $\theta$ denoted by

$$\psi = g(\theta), \tag{4.94}$$

with inverse transformation,

$$\theta = h(\psi). \tag{4.95}$$

Draper and Smith (1981) show that the parameter-effects curvature array $\tilde{A}^T$ with respect to the new parameters is then given by

$$\tilde{A}^T_{ik_1k_2} = A^T_{ik_1k_2} - \sum_{i_1,j_1,j_2} R_{ti i_1} T_{i_1 j_1 j_2} K_{j_1 k_1} K_{j_2 k_2}. \tag{4.96}$$

For $T$ two different expressions can be found, either in terms of $g$,

$$T_{i_1 j_1 j_2} = \sum_l ((D^1 g)^{-1})_{i_1 l} D^2_{j_1 j_2} g_l, \tag{4.97}$$

or, in terms of its inverse $h$,

$$T_{i_1 j_1 j_2} = -\sum_{l_1,l_2} D^2_{l_1 l_2} h_{i_1} ((D^1 h)^{-1})_{l_1 j_1} ((D^1 h)^{-1})_{l_2 j_2}. \tag{4.98}$$

This results allows us to test any reparameterization on its effect on the parameter-effects curvature. This can be done very efficiently as it is not necessary to compute first or second order derivatives of $f$ with respect to the new parameters. In case the reparameterization is used to obtain more reliable confidence regions or likelihood contours, a transformation to meaningful parameters is to be preferred, as the obtained region can then be used directly. Alternatively, one may use a transformation for which the inverse is available, such that the region obtained for the new parameters can be mapped to a corresponding region in the orginal parameter space.

In Draper and Smith (1981) the expected-value transformations of Ross (1970, 1978) are discussed. These, however, are not suitable for models with more than a few parameters as the inverse then generally does not exist or is difficult to obtain.

In Tsai (1988) power transformations of the form $\psi_i = \theta_i^{\gamma_i}$ (if $\gamma_i \neq 0$), and $\psi_i = \ln(\theta_i)$ (if $\gamma_i = 0$) are suggested. The parameter-effects curvature may then be minimized by choosing suitable $\gamma_i$, for $i = 1, \ldots, p$. Their results indicate good performance with respect to the computation of confidence regions.

# Appendix F – Taylor expansion of the maximum likelihood estimator in residuals

If we take

$$g(\phi) = f(K\phi), \tag{4.99}$$

then the maximum likelihood estimator $\hat{\phi}$ is characterized by

$$\frac{d}{d\phi} \|y - g(\phi)\|^2 (\hat{\phi}) = 0, \tag{4.100}$$

In the notation of Appendix A, this can be written as

$$\sum_i (y_i - g_i(\hat{\phi})) D^1_k g_i(\hat{\phi}) = 0. \tag{4.101}$$

To obtain the difference $\hat{\gamma} = \hat{\phi} - \phi^*$ as a function of the random variable $\varepsilon$, both terms of the product are expanded using Taylor series expansions around $\phi^*$, giving

$$\begin{aligned}\sum_i \Big[ (\varepsilon_i - \sum_{k_1} D^1_{k_1} g_i(\phi^*)\hat{\gamma}_{k_1} - \frac{1}{2}\sum_{k_1,k_2} D^2_{k_1k_2} g_i(\phi^*)\hat{\gamma}_{k_1}\hat{\gamma}_{k_2} \\ - \frac{1}{6}\sum_{k_1,k_2,k_3} D^3_{k_1k_2k_3} g_i(\phi^*)\hat{\gamma}_{k_1}\hat{\gamma}_{k_2}\hat{\gamma}_{k_3} + \mathcal{O}(|\hat{\gamma}|^4)) \cdot (D^1_k g_i(\phi^*) \\ + \sum_{l_1} D^2_{kl_1} g_i(\phi^*)\hat{\gamma}_{l_1} + \frac{1}{2}\sum_{l_1,l_2} D^3_{kl_1l_2} g_i(\phi^*)\hat{\gamma}_{l_1}\hat{\gamma}_{l_2} + \mathcal{O}(|\hat{\gamma}|^3))\Big] = 0.\end{aligned} \tag{4.102}$$

From here all derivatives are evaluated in $\phi^*$ and we use $D^1_k g_i = Q_{tik}$ based on the QR-decomposition (4.42). Since $Q_t'Q_t = I$, or $\sum_i Q_{ti k_1} Q_{tik} = \delta_{k_1 k}$, we have

$$\begin{aligned}\hat{\gamma}_k = \sum_i \Big[ (\varepsilon_i - \frac{1}{2}\sum_{k_1,k_2} D^2_{k_1k_2} g_i \hat{\gamma}_{k_1}\hat{\gamma}_{k_2} - \frac{1}{6}\sum_{k_1,k_2,k_3} D^3_{k_1k_2k_3} g_i \hat{\gamma}_{k_1}\hat{\gamma}_{k_2}\hat{\gamma}_{k_3} \\ + \mathcal{O}(|\hat{\gamma}|^4)) \cdot (Q_{tik} + \sum_{l_1} D^2_{kl_1} g_i \hat{\gamma}_{l_1} + \frac{1}{2}\sum_{l_1,l_2} D^3_{kl_1l_2} g_i \hat{\gamma}_{l_1}\hat{\gamma}_{l_2} + \mathcal{O}(|\hat{\gamma}|^3)) \\ - (\sum_{k_1} Q_{tik_1}\hat{\gamma}_{k_1}) \cdot (\sum_{l_1} D^2_{kl_1} g_i \hat{\gamma}_{l_1} + \frac{1}{2}\sum_{l_1,l_2} D^3_{kl_1l_2} g_i \hat{\gamma}_{l_1}\hat{\gamma}_{l_2} + \mathcal{O}(|\hat{\gamma}|^3))\Big].\end{aligned} \tag{4.103}$$

Now we rewrite this results using the rotated residuals

$$\varepsilon_{tk} = \sum_i Q_{tik}\varepsilon_i \quad \text{and} \qquad \varepsilon_{ck} = \sum_i Q_{cik}\varepsilon_i, \tag{4.104}$$

and

$$\varepsilon_i = \sum_{k_1,i_1} (Q_{tik_1}Q_{ti_1k_1} + Q_{cik_1}Q_{ci_1k_1})\varepsilon_{i_1} = \sum_{k_1} (Q_{tik_1}\varepsilon_{tk_1} + Q_{cik_1}\varepsilon_{ck_1}).$$

We then have

$$\begin{aligned}\hat{\gamma}_k = &\varepsilon_{tk} + \sum_i \Big[ -\frac{1}{2}\sum_{k_1,k_2} D^2_{k_1k_2} g_i \hat{\gamma}_{k_1}\hat{\gamma}_{k_2} - \frac{1}{6}\sum_{k_1,k_2,k_3} D^3_{k_1k_2k_3} g_i \hat{\gamma}_{k_1}\hat{\gamma}_{k_2}\hat{\gamma}_{k_3} \\ &+ \mathcal{O}(|\hat{\gamma}|^4)) \cdot (Q_{tik} + \sum_{l_1} D^2_{kl_1} g_i \hat{\gamma}_{l_1} + \frac{1}{2}\sum_{l_1,l_2} D^3_{kl_1l_2} g_i \hat{\gamma}_{l_1}\hat{\gamma}_{l_2} + \mathcal{O}(|\hat{\gamma}|^3)) \\ &+ (\sum_{k_1} Q_{cik_1}\varepsilon_{ck_1} - Q_{tik_1}(\hat{\gamma}_{k_1} - \varepsilon_{tk_1})) \cdot (\sum_{l_1} D^2_{kl_1} g_i \hat{\gamma}_{l_1} \\ &+ \frac{1}{2}\sum_{l_1,l_2} D^3_{kl_1l_2} g_i \hat{\gamma}_{l_1}\hat{\gamma}_{l_2} + \mathcal{O}(|\hat{\gamma}|^3))\Big].\end{aligned} \tag{4.105}$$

This expression can now be used to derive a power series expansion of $\hat{\gamma}$ in $\varepsilon_t$ and $\varepsilon_c$: substitution of the power series (4.59) in (4.105), and equating its two sides lead to the power series coefficients listed in Table 4.1.

# Chapter 5

# Analysis of local model structure: an application to North Sea herring stock assessment[1]

We consider a maximum likelihood estimation procedure used for the stock assessment of North Sea herring. We describe an efficient method for the evaluation of model nonlinearity effects on statistical inference. We further discuss three types of analysis to investigate the appropriateness of the model structure used in the maximum likelihood estimation: (i) analysis of model complexity by means of selection criteria; (ii) analysis of propagation of uncertainty; and (iii) sensitivity analysis of implicit relations, in particular of maximum likelihood estimates with respect to non-estimated variables. The analysis types have the common property that they require only the maximum likelihood estimate and derivatives of model functions evaluated at the maximum likelihood estimate making them suitable for the analysis of large models.

## 5.1 Introduction

Fish stock assessment aims to provide fishery management with information on the stock history, the current state of the stock and its expected future development under different scenarios. Often statistical models are used in order to extract relevant information from available data.

In this chapter we focus on the use of maximum likelihood estimation, and in particular nonlinear regression, for fish stock assessment. Many stock assessment methods currently in use belong to this category, most notably perhaps the statistical catch-at-age methods following ideas by, among others, Doubleday (1976),

[1]This chapter is based on a preprint (co-author Martin Pastoors, RIVO-DLO).

Fournier and Archibald (1982) and Deriso et al. (1985). See Megrey (1989) for an overview of age-structured models. Schnute (1994) presents a general framework for the construction of likelihood functions for sequential fisheries models that also allows the incorporation of process error.

In maximum likelihood estimation a vector of observations is treated as the realization of a random vector. A model consists of a set of density functions indexed by a parameter vector such that each density function is a candidate for the description of the density of the observation vector. The density of the realization of the observation as a function of the parameters is the likelihood function. The notion of likelihood is central to statistical inference, see for example Reid (2000) for an overview. One of its uses is the direct evaluation of relative support for candidates by means of the likelihood ratio (Royall, 1997). Another is its use in Bayesian statistics where the likelihood function transforms prior probability into posterior probability.

In this chapter we discuss a number of methods for the analysis of the model structure describing the relation between parameters and observations. First, in section 5.2.2, we discuss likelihood-based parameter estimation and subsequent statistical inference for nonlinear models. We pay particular attention to the treatment of the influence of model structure nonlinearity on the quality of the inference results.

Next, we discuss three types of analysis that can be used to gain additional insight in the quality of the chosen model structure. The analysis types share the property that they require only the maximum likelihood estimate and derivatives of model functions at the maximum likelihood estimate: no additional evaluations of the model functions for different values of the parameters or other variables are needed. As such, the methods are local in nature, which makes them cheap to perform, particularly if implemented by means of automatic differentiation algorithms (Huiskes, 2001). Of course, the fact that the analysis is local may also limit its validity, and sometimes a more elaborate analysis will be necessary.

The three analysis types are:

1. Use of model selection criteria to investigate the appropriateness of the complexity of the model structure in relation to the available data. Strict likelihood maximization may lead to 'overfitting'. Model selection criteria can be interpreted to provide a 'rate of exchange' between support by the data and model simplicity. Ludwig and Walters (1989), Fournier (1989) and Richards and Schnute (1998) all show the importance of a well-chosen complexity level of the stock dynamics model for obtaining useful parameter estimates. An introduction to the use of selection criteria in a fisheries analysis context is given in Helu et al. (2000). See section 5.2.3.

2. Sensitivity analysis of implicit relations, in particular of the maximum likelihood estimate with respect to variables not estimated, such as variables assumed to be known (e.g. natural mortality), observations and observation weighting coefficients. See section 5.2.4.

3. Propagation of uncertainty information, in particular for variables depending on the estimated parameters. The method described in section 5.2.4 is an extension of the delta method applied in Kimura (1986).

All methods were applied in an analysis of the ICA stock assessment procedure (Patterson and Melvin, 1996) which we shall describe next.

## 5.2 Methods

### 5.2.1 Characterization of the ICA estimation problem

We have re-implemented the Integrated Catch-Age Analysis (ICA) assessment procedure currently in use by the International Council for Exploration of the Sea (ICES) for, among others, the North Sea herring stock. ICA is based on a procedure for analyzing time series of catch-at-age data and research vessel survey information as originally proposed by Deriso et al. (1985). The model and its statistical properties are further discussed in Kimura (1986) and Gudmundsson (1986). Its main aim is to reconstruct the fishery population dynamics by means of commercial catch data and of tuning indices, which are either of age-structured or of spawning stock biomass type.

A VPA-type procedure is used to obtain initial estimates of population abundance and fishing mortality. For the most recent years a more detailed analysis is performed by means of a nonlinear regression model of the fisheries catch data using the 'separability' assumption for the fishing mortality. Details of the entire estimation procedure can be found in Patterson and Melvin (1996). Here we list the symbols (Table 5.1) and equations (Table 5.2) relevant to our present purpose: the characterization of a typical ICA nonlinear regression estimation.

The maximum likelihood estimate corresponding to log-normally distributed residuals is the solution of the minimization problem given by (5.5). Population size and catches are modelled using the familiar cohort dynamics equations and separability assumption (5.1), (5.2) and (5.3) as described in Pope and Shepherd (1982).

The age-structured and spawning stock biomass indices are assumed to be of proportionate type, corresponding to expressions (5.6) and (5.7). The parameters estimated in the regression procedure are:

- Selection coefficients $s_a$ except for the reference age and the last true age (not a plus group) which are assumed to have a selection coefficient equal to one.
- Fishing mortality coefficients $f_y$ at reference age for each year in which the separable constraint (5.2) is assumed.
- Population sizes for all ages in the last year, and population sizes for the last true age in the years for which the separable constraint (5.2) is used.
- Proportianality parameters $q_{A,a}$ and $q_B$ for the tuning indices.

| Symbol | Meaning |
|---|---|
| $N_{a,y}$ | Population of age $a$ in year $y$ (in numbers) |
| $C_{a,y}$ | Total catch of age $a$ in year $y$ (in numbers) |
| $\tilde{C}_{a,y}$ | Observed total catch of age $a$ in year $y$ (in numbers) |
| $I_{A,a,y}$ | Age-structured index $A$ for age $a$ in year $y$ |
| $\tilde{I}_{A,a,y}$ | Observed age-structured index $A$ for age $a$ in year $y$ |
| $I_{B,y}$ | Spawning stock biomass index $B$ in year $y$ |
| $\tilde{I}_{B,y}$ | Observed spawning stock biomass index $B$ in year $y$ |
| $s_a$ | Selection coefficient for age $a$ |
| $f_y$ | Instantaneous fishing mortality rate at reference age in year $y$ |
| $F_{a,y}$ | Instantaneous fishing mortality rate at age $a$ in year $y$ |
| $M_{a,y}$ | Instantaneous natural mortality rate at age $a$ in year $y$ |
| $O_a$ | Maturity coefficient for age $a$ |
| $W_{a,y}$ | Weight in stock at age $a$ in year $y$ |
| $PF$ | Proportion of fishing mortality before spawning |
| $PM$ | Proportion of natural mortality before spawning |
| $PZ_A$ | Proportion of total mortality observation of age-structured index $A$ |
| $SSB_y$ | Spawning stock biomass in year $y$ |
| $\lambda_{a,y}$ | Weighting coefficient of observed catch for age $a$ in year $y$ |
| $\lambda_{A,a,y}$ | Weighting coefficient of age-structured index $A$ for age $a$ in year $y$ |
| $\lambda_{B,y}$ | Weighting coefficient of spawning stock biomass index $B$ in year $y$ |

Table 5.1: Integrated Catch-Age Analysis: Symbols.

- Optionally, parameters for a Beverton and Holt stock-recruit relationship.

The data sets, that together constitute the vector of observations used to estimate the parameters, are listed in Table 5.3.

### 5.2.2 Statistical inference for nonlinear regression problems

The estimation problem presented in the previous section can be formulated as maximum likelihood estimation problem, and in fact as a normal nonlinear regression of the standard type

$$y = f(\theta) + \varepsilon, \quad \varepsilon \sim N(0, \sigma^2 I_n), \tag{5.8}$$

i.e. the components of the error vector $\varepsilon$ are assumed to be independent and normally distributed with expectation zero and variance $\sigma^2$. The parameter vector $\theta = (\theta_1, \ldots, \theta_p)'$ has $p$ components; $n$ denotes the number of components of the observation vector. Appendix A shows how estimation problems with an error structure represented by a known (relative) variance-covariance matrix are transformed to the the standard problem (5.8).

**Population dynamics**

$$N_{a,y} = N_{a+1,y+1} \exp(F_{a,y} + M_{a,y}) \tag{5.1}$$

$$F_{a,y} = s_a f_y \tag{5.2}$$

$$C_{a,y} = \frac{F_{a,y}}{F_{a,y} + M_{a,y}} N_{a,y}(1 - \exp(-F_{a,y} - M_{a,y})) \tag{5.3}$$

$$SSB_y = \sum_a N_{a,y} W_{a,y} O_a \exp(-F_{a,y} PF - M_{a,y} PM) \tag{5.4}$$

**Parameter estimation**

Minimization of:

$$\sum_{a,y} \lambda_{a,y}(\ln(C(a,y)) - \ln(\tilde{C}(a,y)))^2 + \sum_{A,a,y} \lambda_{A,a,y}(\ln(I_A(a,y)) - \ln(\tilde{I}_A(a,y)))^2$$
$$+ \sum_{B,y} \lambda_{B,y}(\ln(I_B(y)) - \ln(\tilde{I}_B(y)))^2 \tag{5.5}$$

**Index types**

$$\text{Age-structured} : \quad I_{A,a,y} = q_{A,a} N_{a,y}(1 - \exp(-(F_{a,y} + M_{a,y}) PZ_A)) \tag{5.6}$$

$$\text{Spawning stock biomass} : \quad I_{B,y} = q_B SSB_y \tag{5.7}$$

Table 5.2: Integrated Catch-Age Analysis: Equations.

The maximum likelihood estimate $\hat{\theta}(y)$ corresponding to a realization $y$ of the observation vector, minimizes

$$S(\theta) = \|y - f(\theta)\|^2 = \sum_{i=1}^{n} (y_i - f_i(\theta))^2. \tag{5.9}$$

The function $f$ is nonlinear in the parameters $\theta$. For statistical inference usually a linear approximation of $f$ around the maximum likelihood estimate is used.

Here we will consider the influence of model nonlinearity on the quality of the statistical inference in more detail. We focus on two types of statistical inference:

1. The computation of the bias and variance-covariance of the maximum likelihood estimator $\hat{\theta}$ based on the assumption that the data is generated by the density function corresponding to the maximum likelihood estimate.
2. The computation of likelihood intervals for individual parameters based on a method presented in Clarke (1987b).

| Observations | Type | Age range | Year range |
|---|---|---|---|
| Catch | AS | 0-9(+) | 75-94 |
| IBTS (adult) | AS | 2-5(+) | 83-95 |
| IBTS (1ring) | AS | 1 | 79-95 |
| Acoustic | AS | 2-9(+) | 89-94 |
| MIK | AS | 0 | 78-95 |
| MLAI | SSB | - | 83-92 |
| Recruitment | - | - | 75-94 |

Table 5.3: Age and year ranges of catch and tuning indices observations. IBTS: International Bottom Trawl Survey; MIK: recruitment index; MLAI: Multiplicative Larvae Abundance Index. Index types: age-structured (AS) and spawning stock biomass (SSB). A (+) in the age range indicates that the highest age group is treated as a plus group.

We refer to Chapter 4 for a general overview of the interaction between model nonlinearity and statistical inference.

### Estimation bias and variance-covariance

If the observations $y$ have a distribution given by (5.8) for values $\theta = \theta^*$ and $\sigma = \sigma^*$, then asymptotically (i.e. for $n \to \infty$), we have that $\hat{\theta}$ converges in distribution to a normal random variable with expectation $E[\hat{\theta}] = \theta^*$ and variance-covariance matrix $\text{var}(\hat{\theta}, \hat{\theta}) = \sigma^{*2}(Df'Df)^{-1}$. Here $Df$ is the derivative of $f$ with respect to $\theta$ evaluated at $\theta^*$. These results indicate that asymptotically, the use of a linear approximation to obtain the moments of the maximum likelihood estimator is valid.

For finite $n$, however, we will generally have that $\hat{\theta}$ is biased, i.e. $E[\hat{\theta}] \neq \theta^*$, and $\text{var}(\hat{\theta}, \hat{\theta}) \neq \sigma^{*2}(Df'Df)^{-1}$. Also, the maximum likelihood estimator will not be normally distributed.

The computation of approximations to the bias $\beta = E[\hat{\theta}] - \theta^*$ and corrections for the estimation variance-covariance matrix by means of higher order derivatives is described in section 4.4.5 of Chapter 4.

### Likelihood intervals

The construction of simultaneous confidence regions in the presence of model nonlinearity is discussed in Hamilton et al. (1982) and Draper and Smith (1981). These methods require the choice of a reparameterization of the model to reduce the so-called parameter-effects curvature. This is a tedious and not always so easy job for models with large numbers of parameters. Instead, we will focus on an approach presented in Clarke (1987b). The procedure described there allows for the evaluation of the effect of model nonlinearity on likelihood intervals for an individual parameter. It also has a built-in warning system that indicates when a more elaborate analysis is required.

A likelihood interval for a scalar parameter $\theta_i$ is defined by

$$\{\theta_i | S(\hat{\theta}(\theta_i)) - S(\hat{\theta}) \leq c^2\hat{\sigma}^2\}, \quad i = 1, \ldots, p, \tag{5.10}$$

where $\hat{\theta}(\theta_i)$ minimizes $S(\theta)$ with respect to $\theta$ with the value of $\theta_i$ fixed, and $\hat{\sigma}$ is an estimate for $\sigma$. The quantity $S(\hat{\theta}(\theta_i))/\hat{\sigma}^2$ is referred to as the profile log-likelihood for $\theta_i$.

The likelihood interval given by (5.10) can be interpreted directly as a profile likelihood contour, see for instance Royall (1997). If $c$ is chosen equal to $pF_\alpha(p, \nu)$, where $F_\alpha(p, \nu)$ denotes the critical $F$ value with $p$ and $\nu$ degrees of freedom and tail area probability $\alpha$, the likelihood interval may serve as a 100(1-$\alpha$)% confidence interval.

If the regression model is linear in $\theta$, the lower and upper limits of the likelihood interval are given by $\hat{\theta}_i \pm \text{sd}(\hat{\theta}_i)c$. Here sd denotes the asymptotic estimation standard deviation for the maximum likelihood estimate. Its value can be computed from the estimation variance-covariance matrix discussed earlier. An expression for $\text{sd}(\hat{\theta}_i)$ is also given in Appendix B.

In Clarke (1987b) it is shown that more accurate approximations for the likelihood interval limits are given by

$$\hat{\theta}_i - \text{sd}(\hat{\theta}_i)c(1 + \frac{1}{2}\Gamma_i\hat{\sigma}c), \quad \text{and} \quad \hat{\theta}_i + \text{sd}(\hat{\theta}_i)c(1 - \frac{1}{2}\Gamma_i\hat{\sigma}c), \quad i = 1, \ldots, p, \tag{5.11}$$

for the lower and upper limits, respectively. The variables $\Gamma_i$ depend on first and second order derivatives of $f$ with respect to $\theta$. The computation of $\Gamma$ is described in Appendix B.

The derivation of this result is based on a power series expansion of the limits of the likelihood interval in $c\hat{\sigma}$. In this case terms of degree 3 and higher were neglected. Based on the properties of this power series, Clarke (1987b) suggests the use of the following rule of thumb for assessing the significance of $\Gamma_i$.

- If $\|\frac{1}{2}\Gamma_i\hat{\sigma}c\| \leq 0.1$, curvature effects may be ignored and the usual linear analysis will suffice.
- If $.1 < \|\frac{1}{2}\Gamma_i\hat{\sigma}c\| \leq \frac{1}{3}$ nonlinearity effects have to be compensated for and this can be realized by taking the interval limits given by (5.11).
- If $\|\frac{1}{2}\Gamma_i\hat{\sigma}c\| > \frac{1}{3}$ nonlinearity effects are so serious that a more thorough analysis along the lines of Hamilton et al. (1982), Draper and Smith (1981) and Clarke (1987a) is required.

For the ICA estimation we computed the estimation bias and variance-covariance matrix, and the likelihood intervals, both by means of our C++ library for model analysis by automatic differentiation (Huiskes, 2001). We further compared the outcomes to the results obtained by using traditional inferential methods based on linearization.

### 5.2.3 Model selection criteria

By means of maximum likelihood estimation, the density function is chosen from a set of candidate density functions that maximizes the density value of the given realization of the observation vector. This leads to the unfortunate situation that the chosen density function generally will be influenced by the peculiarities of a particular realization: and this tendency to 'overfit' will be greater for models of higher complexity.

In general, the best one can hope to do is to choose that model for which maximum likelihood estimation *on the average* leads to a density function that is as close as possible to the true density function. This true density function is that probability density function from which we tentatively assume the data to be generated. In Chapter 4 it was shown that if the Kullback-Leibler discrepancy is used to measure the distance to this true density, then minimization of this discrepancy is equivalent to the maximization of the expected density value, based on the density function associated with the maximum likelihood estimate. In this case, however, the expected density value is taken for observations that are *independent* of the observation used to determine the maximum likelihood estimate. So, theoretically, maximization of this expectation instead of ordinary likelihood, would prevent overfitting. In practice though we cannot just compute the expectation associated with each model: only estimators can be derived.

An important class of selection criteria consists of estimators for the model-dependent part of the Kullback-Leibler discrepancy. The selection criterion that is most appropriate for the estimation of this quantity in the case of nonlinear regression, in particular for the case that the number of parameters is relatively large in comparison to the number of observations, is the *corrected* Akaike selection criterion (AICc) (Hurvich and Tsai, 1989). This criterion is generally less biased than the uncorrected Akaike criterion (AIC), and since the correction is a constant term, AICc has equal variance. The AICc is given by

$$\begin{aligned} \text{AICc} &= \text{AIC} + \frac{2k(k+1)}{n-k-1} = -2\log(\mathcal{L}_y(\hat{\theta},\hat{\sigma})) + 2k + \frac{2k(k+1)}{n-k-1} \\ &= n\log(2\pi S(\hat{\theta})/n) + n + 2k + \frac{2k(k+1)}{n-k-1}, \end{aligned} \tag{5.12}$$

where $\mathcal{L}_y(\hat{\theta},\hat{\sigma})$ is the likelihood value at the maximum likelihood estimate, $k$ is the number of estimated parameters (including $\sigma$, so $k = p+1$), and $n$ is the number of observations; log is the natural logarithm. Burnham and Anderson (1998) provide a general introduction to Kullback-Leibler type selection criteria.

We also use Schwarz's Bayesian information criterion (BIC) (Schwarz, 1978), given by

$$\text{BIC} = -2\log(\mathcal{L}_y(\hat{\theta},\hat{\sigma})) + k\log n = n\log(2\pi S(\hat{\theta})/n) + n + k\log n. \tag{5.13}$$

For a comparison of the AIC and BIC criteria we refer to Helu et al. (2000). Both criteria consist of a term representing lack-of-fit and a term that can be considered

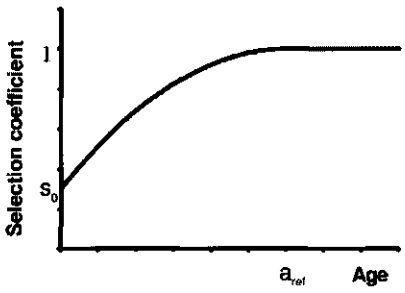



Figure 5.1: Parameterized selection coefficient at age relationship.

as a penalty for complexity. As their computation only requires the maximum likelihood value, they are generally cheaper to use than selection methods based on cross-validation (see Linhart and Zucchini (1986) for model selection by cross-validation).

We use the two criteria to study two modifications of the original ICA stock dynamics model. In the original ICA model the age-structured tuning indices use a proportionality parameter for every age group for which data is available. For the acoustic tuning index this leads to 8 proportionality parameters, which seems to be a rather large number for modeling the relation between the tuning index observations and the population sizes. As a first experiment we therefore investigate the effect of replacing the proportionality parameters $q_a$ for the different age groups by a single proportionality parameter $q$. As a second experiment we consider a parameterization of the age selection coefficients. Instead of using selection coefficients for each age group, we use a selection curve that depends on only two parameters. The selection curve is given by

$$s(a) = \begin{cases} 1 + \frac{s_0 - 1}{a_{\text{ref}}^2}(a - a_{\text{ref}})^2 & \text{if } a < a_{\text{ref}} \\ 1 & \text{if } a \geq a_{\text{ref}} \end{cases} \tag{5.14}$$

where the selection coefficient at age zero $s_0$, and the reference age $a_{\text{ref}}$ are the new parameters to be estimated (see Figure 5.1).

### 5.2.4 Local sensitivity analysis

For differentiable input-output relations, local sensitivity of output variables with respect to input variable perturbations can be quantified by means of Taylor expansions. We start out by discussing explicit input-output relations, and then move on to implicit relations. Also a local method for propagation of uncertainty is discussed.

### Explicit relations

Consider a response $R(x)$ with $n$ input variables $x = (x_1, \ldots, x_n)'$. We define sensitivity coefficients of the response at input $x$ by

$$s_i^1(x) = \frac{\partial R}{\partial x_i}(x), \quad i = 1, \ldots, n. \tag{5.15}$$

The coefficient $s_i^1(x)$ is a linear estimate of the number of units change in $R$ caused by a unit change in $x_i$. The sensitivity coefficients can be made independent of the units of the response and input variables by using normalized sensitivity coefficients

$$\bar{s}_i^1(x) = \frac{\partial R}{\partial x_i}(x)\frac{x_i}{R}, \quad i = 1, \ldots, n. \tag{5.16}$$

The coefficient $\bar{s}_i^1(x)$ represents a linear estimate of the change in $R$ as a result of a one percent change in $x_i$. Second order sensitivity coefficients are analogously defined by

$$s_{ij}^2(x) = \frac{\partial^2 R}{\partial x_i \partial x_j}(x), \quad \text{and} \quad \bar{s}_{ij}^2(x) = \frac{\partial^2 R}{\partial x_i \partial x_j}(x)\frac{x_i x_j}{R}, \tag{5.17}$$

for $i, j = 1, \ldots, n$. For a relative output perturbation $\delta R/R$ resulting from relative input perturbations $\delta x_i/x_i$, $i = 1, \ldots, n$, we now have

$$\frac{\delta R}{R} = \sum_{i=1}^{n} \bar{s}_i^1(x)\left(\frac{\delta x_i}{x_i}\right) + \frac{1}{2}\sum_{i,j=1}^{n} \bar{s}_{ij}^2(x)\left(\frac{\delta x_i}{x_i}\right)\left(\frac{\delta x_j}{x_j}\right) + \mathcal{O}(|\delta x|^3). \tag{5.18}$$

The sensitivity coefficients defined here are also used in expressions for the propagation of uncertainty information discussed below. In section 5.3.3 we give a simple example of the use of sensitivity coefficients for a predicted spawning stock biomass.

### Implicit relations

We focus on the sensitivity analysis of the implicit relation defining the maximum likelihood estimator, given by

$$\frac{d}{d\theta} S(\hat{\theta}, \tilde{x}) = 0. \tag{5.19}$$

$S$ may be given by (5.9) or by any other likelihood, log-likelihood or objective function which has been optimized in order to obtain the parameter estimate $\hat{\theta}$. The vector $\tilde{x}$ consists of the non-estimated variables for which a sensitivity analysis is desired. Examples of such variables are the observations, regressor variables and the observation weighting coefficients.

If the second order derivative of $S$ with respect to the parameters exists and is invertible, then by the implicit function theorem there exists a function $\theta = g(x)$

in the neighborhood of $\tilde{x}$ that satisfies $\hat{\theta} = g(\tilde{x})$ and for which the derivative is given by

$$\frac{d}{dx}g(\tilde{x}) = -\left(\frac{d^2}{d\theta^2}S(\hat{\theta},\tilde{x})\right)^{-1}\frac{d^2}{dxd\theta}S(\hat{\theta},\tilde{x}). \tag{5.20}$$

This linear map can be used to approximate the change $\delta\theta$ in the parameter estimate $\hat{\theta}$ resulting from a change $\delta x$ in $\tilde{x}$:

$$\delta\theta = \frac{d}{dx}g(\tilde{x})\delta x. \tag{5.21}$$

In Chapter 3 (section 3.6) a higher-order extension of this result is presented.

We apply this method to evaluate the sensitivity of the ICA parameter estimates with respect to the natural mortality level and the catch weighting coefficients.

### Propagation of uncertainty information

Consider now a vector response $R(x)$, consisting of $m$ output variables depending on $n$ input variables $x = (x_1, \ldots, x_n)'$. We are interested in obtaining the expectation and variance-covariance matrix of the response vector, given a known expectation and variance-covariance matrix of the input variables. Here we list the results for the case that the input variables have a multivariate normal distribution with expectation $\mu$ and variance-covariance matrix $V$. General expressions, valid for arbitrary probability distributions of the input variables, are listed in Appendix B. Note that for distributions other than the normal distribution, also higher order moments are required. The derivations of the expressions are given in Chapter 3 (section 3.4.2).

For the expectation of response component $p$, we have

$$E[R_p(x)] = R_p(\mu) + \frac{1}{2}\sum_{i,j=1}^{n} s^2_{pij}V_{ij}, \tag{5.22}$$

for $p = 1, \ldots, m$. Since the third order central moments of the multivariate normal distribution are zero, the terms neglected here are of order four and higher. For element $(p, q)$ of the response variance-covariance matrix, we similarly have

$$\mathrm{cov}(R_p(x), R_q(x)) = \sum_{i,j=1}^{n} s^1_{pi}s^1_{qj}V_{ij} + \frac{1}{4}\sum_{i,j,k,l=1}^{n} s^2_{pij}s^2_{qkl}(V_{ik}V_{jl} + V_{il}V_{jk}), \tag{5.23}$$

for $p, q = 1, \ldots, m$. In these expressions $s^1_{pi}$ and $s^2_{pij}$ are the first and second order sensitivity coefficients of response component $p$, respectively. All sensitivity coefficients are evaluated in $\mu$. Note that $V_{ij} = \rho_{ij}\sigma_i\sigma_j$ where $\sigma_i$ is the standard deviation of input variable $i$ and $\rho_{ij}$ is the correlation between input variables $i$ and $j$ (which is equal to 1 if $i = j$).

The linear part of this expression is the well-known 'sandwich' rule for the approximation of covariance. Each of the terms of the linear approximation can be

attributed to either an input variable or a correlation between input variables. We will consider this in more detail in section 5.3.3 where we continue the investigation of the spawning stock biomass prediction mentioned earlier.

## 5.3 Results

### 5.3.1 Nonlinearity effects in the ICA estimation inference

Table 5.4 lists the parameter estimates for the selection coefficients and fishing mortality at reference age, the coefficient of variation (which is equal to the standard deviation divided by the parameter estimate) and the measures of nonlinearity discussed in section 5.2.2.

Note that a number of the parameters have a value of $\|\frac{1}{2}\Gamma\hat{\sigma}c\|$ greater than 0.1 making it useful to correct the likelihood intervals for nonlinearity effects. None of the parameters have a value of $\|\frac{1}{2}\Gamma\hat{\sigma}c\|$ greater than 0.33.

In Figure 5.2 the parameter estimates with associated likelihood intervals are shown. The error bars on the left are obtained by a linear approximation; the bars on the right by the method of Clarke (1987b).

| Parameter | Estimate | CV (%) | $\|\frac{1}{2}\Gamma\hat{\sigma}c\|$ | Bias (%) | Correction Sd (%) |
|---|---|---|---|---|---|
| $s_0$ | 0.26 | 19 | 0.20 | 1.5 | 5.0 |
| $s_1$ | 0.61 | 19 | 0.17 | 1.3 | 3.6 |
| $s_2$ | 0.81 | 12 | 0.13 | 0.78 | 2.1 |
| $s_3$ | 0.76 | 11 | 0.12 | 0.65 | 1.9 |
| $s_5$ | 0.95 | 8.3 | 0.10 | 0.28 | 1.2 |
| $s_6$ | 1.0 | 7.7 | 0.070 | 0.12 | 0.34 |
| $s_7$ | 0.99 | 13 | 0.12 | 0.72 | 2.0 |
| $f_{89}$ | 0.61 | 8.7 | 0.056 | 0.079 | 0.3 |
| $f_{90}$ | 0.48 | 8.6 | 0.056 | 0.079 | 0.33 |
| $f_{91}$ | 0.47 | 8.3 | 0.057 | 0.079 | 0.36 |
| $f_{92}$ | 0.66 | 8.1 | 0.048 | 0.060 | 0.19 |
| $f_{93}$ | 0.76 | 9.1 | 0.049 | 0.051 | 0.077 |
| $f_{94}$ | 0.58 | 13 | 0.11 | 0.25 | 1.8 |

Table 5.4: Parameter estimates for the selection coefficients and fishing mortality coefficients at reference age, with associated coefficient of variation (CV) and measures of nonlinearity.

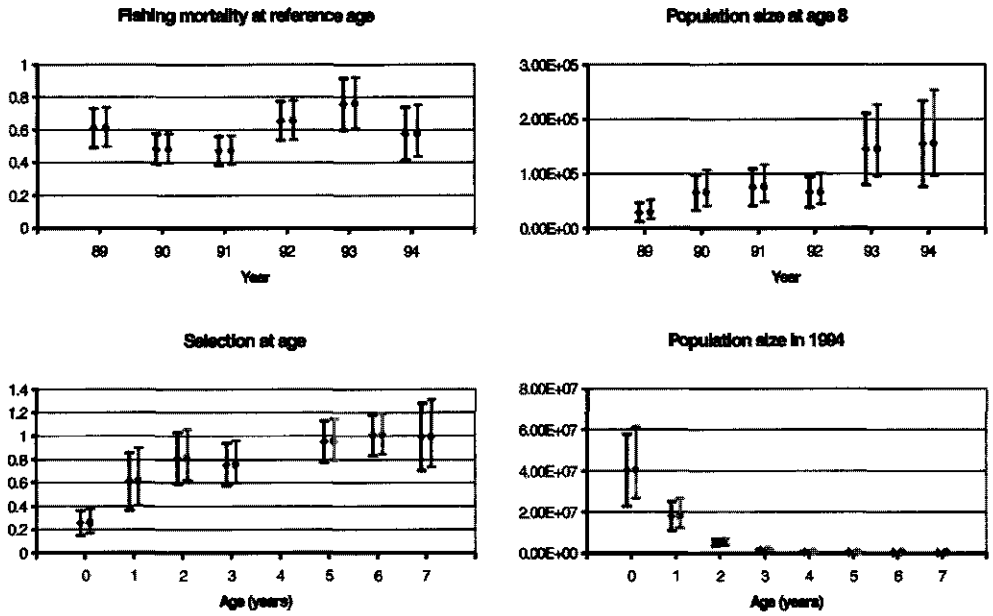



Figure 5.2: Parameter estimates and associated linear (left) and nonlinear (right) likelihood profiles.

## 5.3.2 Model selection criteria

### Age-dependent acoustic index proportionality parameters

The estimated proportionality parameters $q_2$ through $q_9$ for the standard ICA model are listed in Table 5.5. As outlined in section 5.2.3, we consider the effect of using a single acoustic index proportionality parameter $q$ for all age groups, i.e. one proportionality parameter independent of age. The resulting parameter estimate and its associated inference estimates are also listed in Table 5.5. The resulting model selection criteria values are listed in Table 5.6.

### Parameterization of the selection coefficients

The resulting parameters and associated statistical inference values of the selection curve modification described in section 5.2.3 are listed in Table 5.7. The selection criteria values for this selection modification are also listed in Table 5.6.

## 5.3.3 Local sensitivity analysis

### Explicit relations

We investigated the sensitivity of the expected spawning stock biomass in the year after the period of analysis, i.e. in 1995. This spawning stock biomass ($SSB_{95}$) is

| Parameter | Estimate | CV (%) | Likelihood interval limits lower | upper | $\|\frac{1}{2}\Gamma\hat{\sigma}c\|$ |
|---|---|---|---|---|---|
| $q_2$ | 1.37 | 15.7 | 0.970 | 1.94 | 0.178 |
| $q_3$ | 1.44 | 12.7 | 1.09 | 1.91 | 0.143 |
| $q_4$ | 1.63 | 13.3 | 1.21 | 2.19 | 0.148 |
| $q_5$ | 2.08 | 9.37 | 1.67 | 2.55 | 0.073 |
| $q_6$ | 2.24 | 13.3 | 1.64 | 3.00 | 0.111 |
| $q_7$ | 2.26 | 19.4 | 1.46 | 3.44 | 0.194 |
| $q_8$ | 2.39 | 22.8 | 1.45 | 3.91 | 0.241 |
| $q_9$ | 2.31 | 18.1 | 1.55 | 3.44 | 0.200 |
| $q$ | 1.74 | 7.03 | 1.49 | 2.04 | 0.052 |

Table 5.5: Acoustic tuning index estimates for proportionality parameters $q_2$ through $q_9$, and $q$.

| Model | Number of parameters | Degrees of freedom | ESS | Residual variance | AICc | BIC |
|---|---|---|---|---|---|---|
| I | 45 | 173 | 166.6 | 0.96 | 675.3 | 802.3 |
| II | 38 | 180 | 178.0 | 0.99 | 667.4 | 779.3 |
| III | 33 | 185 | 188.26 | 1.01 | 665.7 | 764.4 |

Table 5.6: Error sum of squares (ESS), and corrected Akaike (AICc) and BIC information criteria for (I) the standard ICA model, (II) the modified model with a single acoustic index parameter, and (III) the modified model with both the single acoustic index parameter and the selection curve parameterization.

computed by means of equation (5.4) of Table 5.2. The fishing mortality in this expression was computed by first assuming that the estimated selection at age pattern remains constant. Next, the fishing mortality at reference age $f_{95}$ can be chosen according to a fishing pressure development scenario. In this example we fixed $f_{95}$ at a value of 0.5. The normalized sensitivity coefficients indicate that the estimated $SSB_{95}$ was most sensitive to the parameters of age groups 1,2 and 3 of both the population size in 1994 and the selection coefficients (see Figure 5.3).

The normalized sensitivity to the fishing mortality at reference age is equal to zero except for the fishing mortality of 1994. For this year the normalized sensitivity coefficient was equal to -0.44. The normalized sensitivity coefficient for $f_{95}$ was equal to -0.28. A 10 percent increase in $f_{94}$ and $f_{95}$ thus gives rise to approximate decreases in $SSB_{95}$ of 4.4 percent and 2.8 percent, respectively. The results indicate that no single parameter has an excessive influence on $SSB_{95}$. We consider the combined effect of the uncertainty in the various input variables in more detail below.

| Parameter | Estimate | CV (%) | Likelihood interval limits | | $\|\frac{1}{2}\Gamma\hat{\sigma}c\|$ |
|---|---|---|---|---|---|
| | | | lower | upper | |
| $s_0$ | 0.31 | 16.8 | 0.210 | 0.444 | 0.168 |
| $a_{\text{ref}}$ | 2.57 | 25.8 | 0.984 | 3.986 | 0.059 |
| $f_{89}$ | 0.52 | 4.53 | 0.472 | 0.579 | 0.035 |
| $f_{90}$ | 0.40 | 4.60 | 0.363 | 0.447 | 0.035 |
| $f_{91}$ | 0.40 | 4.67 | 0.355 | 0.438 | 0.031 |
| $f_{92}$ | 0.56 | 4.90 | 0.496 | 0.619 | 0.017 |
| $f_{93}$ | 0.63 | 6.74 | 0.540 | 0.733 | 0.027 |
| $f_{94}$ | 0.46 | 10.6 | 0.364 | 0.585 | 0.107 |

Table 5.7: Selection coefficient and fishing mortality at reference age parameter estimates for the modified ICA model with both the single acoustic index parameter and the selection curve parameterization.

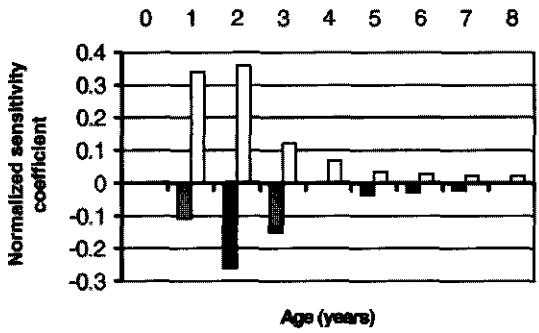



Figure 5.3: Normalized sensitivity coefficients for the predicted 1995 spawning stock biomass with respect to the selection coefficients (gray bars) and population numbers in 1994 (blank bars).

### Implicit relations

Using the local sensitivity method for implicit relations described in section 5.2.4, we investigated the sensitivity of the estimated parameters with respect to the natural mortality and total catch weighting level. The natural mortality used in the ICA assessment depends on age and is assumed known. The gray bars in Figure 5.4 show the parameter sensitivity to changes in the natural mortality level. The ratios between the values of the natural mortality for the different ages were kept constant in these changes. In particular the selection coefficients and the population sizes in 1994 for the young age groups are sensitive to the natural mortality level.

The blank bars in Figure 5.4 show the parameter sensitivity to a 10 percent increase in the catch weighting level. This level is defined as the percentage of the squared catch weighting coefficients in the total sum of squared weights. The cohort sizes at maximum age are most sensitive to the catch weighting level.

The same information can also be obtained by re-fitting the model for the changed values of the non-estimated variables. This means, however, that a new optimization must be performed for every variable for which sensitivity information is desired. If for example the sensitivity with respect to the observations must be computed, in the current case-study this would require 218 optimizations with respect to 45 parameters. The local sensitivity method for implicit relations computes the parameter sensitivity for all variables of interest in a single computation. Using our system (Pentium II, 350 MHz, 64 Mb; Huiskes (2001)) this takes about 4 seconds.

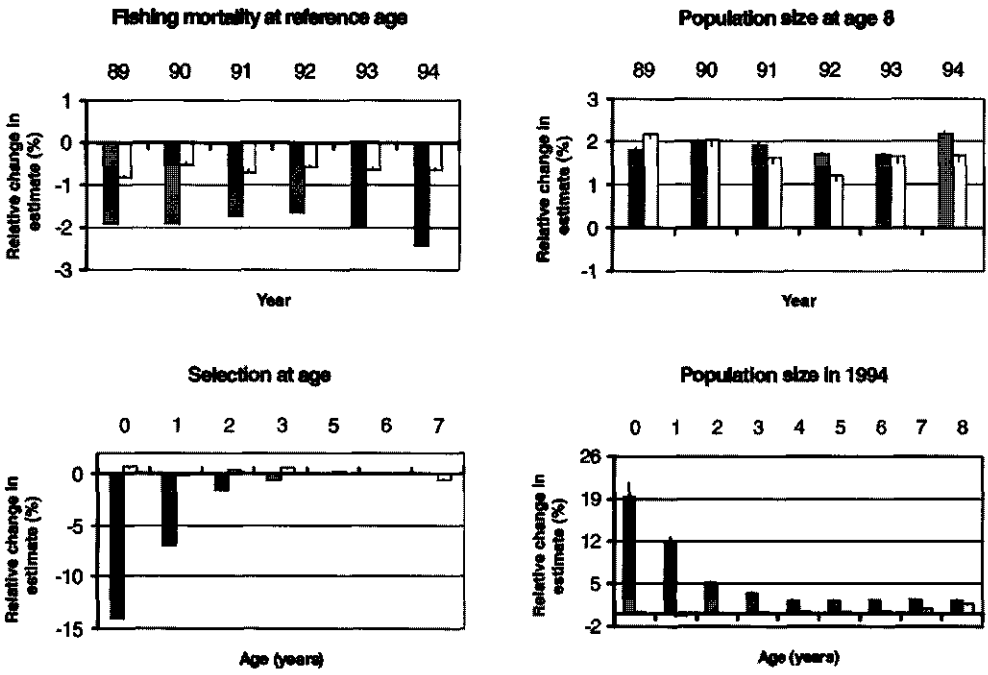



Figure 5.4: Approximate relative changes in estimated parameters resulting from an increase in the natural mortality level (gray bars) and in the catch observation weighting level (blank bars) of 10 percent. The lines on top of the bars indicate the exact values computed by re-fitting of the model.

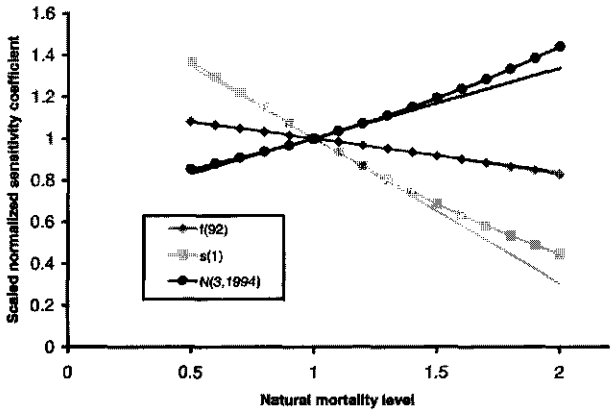



Figure 5.5: Comparison of scaled normalized sensitivity coefficients for three parameters obtained by the local sensitivity method for implicit relations (curve without symbols) and by re-fitting the model (curve with symbols). The positions of the symbols indicate the values of the coefficients obtained by re-fitting the model for the associated level of the natural mortality.

In Figure 5.5 the results of the local method are compared to exact results obtained by re-fitting the model. We considered the normalized sensitivity coefficients of three parameters for values of the natural mortality from 0.5 to 2 times the orginal level. The normalized sensitivity coefficients are scaled by their values for the original level of the natural mortality. In addition, the lines on the bars in Figure 5.4 indicate the errors of the local method in comparison to the exact values obtained by re-fitting.

### Propagation of uncertainty information

We considered the combined effect of the uncertainty of the estimated parameters and the variable $f_{95}$ on the uncertainty of the predicted spawning stock biomass $SSB_{95}$ described above. The relation between these variables and $SSB_{95}$ is described by equation (5.4) of Table 5.2. For the variance-covariance of the parameters we used the estimation variance-covariance matrix obtained in section 5.3.1. The coefficient of variation for $f_{95}$ was set, as an example, at 10 percent. The coefficient of variation of $SSB_{95}$, obtained using (5.23), was equal to 12.9 percent. The contribution of the second order terms to the total variance was less than one percent. Each of the terms of the linear approximation can be attributed to either one of the input variables, or to a correlation between two input variables. This results in the decomposition of variance shown in Figure 5.6. Note that the some of the contributions are grouped, for example $N(3:8,94)$ indicates the contribution to the variance of the variables $N_{3,94}$ through $N_{8,94}$. All contributions of the correlations are also lumped into one category. The correlation terms with a relevant contribution mainly involved correlations including at least one of the

parameters $f_{94}$, $s_2$, $N_{1,94}$ and $N_{2,94}$.

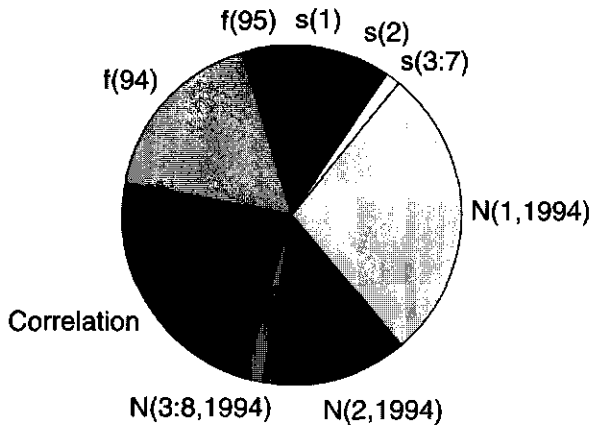



Figure 5.6: Decomposition of variance of spawning stockbiomass prediction into various sources of input uncertainty.

## 5.4 Discussion

If we use a nonlinear relation between parameters and the model expectation of observations in regression analysis, the usual methods of statistical inference based on linearization may lead to inaccurate results. To gain insight into the effects of model nonlinearity and to obtain accurate inferential information in the presence of model nonlinearity a number of methods are available. We found that in particular the method of Clarke (1987b) provides a suitable method for the investigation of nonlinearity effects. This method on the one hand provides a warning system against very serious nonlinearity effects, and on the other hand yields corrections to the likelihood intervals obtained by linearization for cases where the effects of nonlinearity are relatively mild. From Table 5.4 we found that for most parameters there was indeed some effect of nonlinearity, but these effects were not sufficiently severe to employ more elaborate methods of investigation based on reparameterization of the model. The corrected likelihood intervals, shown in Figure 5.2, were shifted in relation to the intervals based on linearization, resulting in intervals that were not symmetric around the parameter estimate.

Ratkowsky (1983) suggests that an absolute value of the relative bias in excess of 1% is a good rule for indicating nonlinear behavior in the parameter estimates and that for such cases corrections must be calculated for parameter covariance and likelihood intervals. Observing the bias listed in Table 5.4, this rule seems to be in reasonable agreement with the rule of thumb given in section 5.2.2 based on

the values of $\|\frac{1}{2}\Gamma\hat{\sigma}c\|$. It can also be noted from the table that the corrections to the estimation standard deviation due to nonlinearity may be as large as 5 percent.

Next, we considered the use of model selection criteria as an aid in the choice of an appropriate model structure complexity. In our experiments we considered two modifications to the original ICA estimation procedure that both constituted a reduction in the number of parameters to be estimated. For both model modifications we observed a decrease in model fit (Table 5.6), which was to be expected for an increasing number of degrees of freedom. In both cases we found that the selection criteria decreased. This does not necessarily mean that the modified models are 'better' in the sense of a higher level of realism. In the case of the AICc it means that it is expected that the modified models on the average get closer to the theoretical true density of the observations.

In the experiment where the acoustic proportionality parameters $q_2$ through $q_9$ were replaced by a single parameter $q$, we found that the coefficient of variation and the likelihood intervals were substantially reduced for the single parameter, see Table 5.5. The effects of nonlinearity were reduced by this modification as well. After parameterization of the selection coefficients in the second experiment, we again found smaller coefficients of variation and likelihood intervals. This was also the case for the parameters that were not directly involved in the modification such as the fishing mortality at reference age parameters.

Using the method for the sensitivity analysis of implicit relations, we investigated the sensitivity of the estimated parameters with respect to variables that are assumed to be known and fixed. An important application of this method is the investigation of the influence of the catch weighting level coefficients. The assumption of equal error variance in nonlinear regression dictates that the coefficients should be chosen by using reliable estimates of the relative inverse variance of the logarithms of the observations. Accurate information of this type is often not available. Evaluation of the sensitivity of the parameter estimates with respect to these coefficients is therefore recommended in order to assess the possible effects of such lack of information.

Comparison with model re-fitting shows that the results obtained by this method are generally accurate if the changes in the variables are within a reasonable range from their original values.

Once reliable uncertainty information is available with respect to the estimated parameters, it also becomes interesting to establish the uncertainty in variables that are derived from these parameters. We considered a local method for the propagation of variance-covariance and gave an example for a spawning stock biomass prediction. This method can be applied in the same manner for the sensitivity analysis of general biological reference points (see for example Mace (1994)). We also showed how the method can be used to decompose the output uncertainty into various sources of input uncertainty. In Figure 5.6 we observe that the population size in 1994 for age 1 has the largest contribution to the uncertainty in $SSB_{95}$. This is in line with the results obtained from the estimation and Figure 5.3: $N(1,94)$ both has a high coefficient of variation, indicating that the uncertainty in the variable can lead to large relative changes of its value, and a

high normalized sensitivity coefficient, indicating that the spawning stock biomass is sensitive to such changes.

As Fournier (1989) points out, managing a stock is not a static process where a model is used once and a policy decision based on the analysis made. It is a dynamic and ongoing process. The methods described here are relatively cheap to perform and require only the maximum likelihood estimate and first and sometimes higher order derivatives of the model functions at the maximum likelihood estimate. This makes the methods suitable for efficient and routine computation. As such, and because of their wide applicability, we believe the methods could provide a valuable contribution to present day software systems that take the intricate nature of stock assessment into account.

# Appendix A – Transformation of nonlinear regression problems

Consider the following nonlinear regression problem

$$\tilde{y} = \tilde{f}(\theta) + \varepsilon, \quad \varepsilon \sim N(0, \sigma^2 V), \tag{5.24}$$

where $\tilde{y}$ is a vector of $n$ observations, consisting for example of catch-at-age numbers and observations from various research vessel surveys. The function $\tilde{f}(\theta)$ represents the relation between the parameters $\theta$ to be estimated and the model predictions for the observed variables. The error vector $\varepsilon$ is assumed to have a multivariate normal distribution with mean 0 and covariance $\sigma^2 V$. The positive definite matrix $V$ contains the assumed relative covariance of the observations; $\sigma^2$ must be estimated. Using a Cholesky decomposition of $V$ represented by $V = LL^T$, (5.24) can be transformed by means of $y = L^{-1}\tilde{y}$ and $f = L^{-1}\tilde{f}$ to the standard problem

$$y = f(\theta) + \varepsilon, \quad \varepsilon \sim N(0, \sigma^2 I_n), \tag{5.25}$$

where the residuals are now independent and have identical variance $\sigma^2$. Information regarding the matrix $V$ usually consists of weighting coefficients indicating the relative weight assigned to the observations. The assumption of a known $V$ is rather strong and it is important to investigate the sensitivity of results with respect to this matrix.

# Appendix B – Expressions for the computation of likelihood intervals

Let

$$f_i^k = \frac{\partial f_k}{\partial \theta_i}, \quad \text{and} \quad f_{ij}^k = \frac{\partial^2 f_k}{\partial \theta_i \partial \theta_j}. \tag{5.26}$$

Consider a $p \times p$ matrix $G$, with elements $g_{ij}$ and its inverse $G^{-1}$ with elements, $g^{ij}$. The elements of $G$ are defined by

$$g_{ij} = f_i^k \delta_{kl} f_j^l, \tag{5.27}$$

where $\delta_{kl} = 1$ when $k = l$ and zero otherwise. For (5.8), the asymptotic standard deviation of the maximum likelihood estimator of $\theta_i$ is given by

$$\mathrm{sd}(\theta_i) = (g^{ii}\hat{\sigma}^2)^{\frac{1}{2}}, \quad i = 1, \ldots, p. \tag{5.28}$$

The expression for the $\Gamma$ variable used in (5.11) is given by

$$\Gamma_i = (g^{ii})^{-\frac{3}{2}} \sum_{a,b,c=1}^{p} g^{ia} g^{ib} g^{ic} f_a^k \delta_{kl} f_{bc}^l, \quad i = 1, \ldots, p. \tag{5.29}$$

# Appendix C – Expressions for propagation of uncertainty

### Expectation

The $N$-th order approximation of the response expectation in terms of central moments of the input variables is given by

$$E_N[R_p(x)] = R_p(E[x]) + \sum_{k=2}^{N} \frac{1}{k!} \sum_{i_1,\ldots,i_k=1}^{n} \left( \frac{\partial^k R_p}{\partial x_{i_1} \ldots \partial x_{i_k}} \right)_{E[x]} E[\delta x_{i_1} \ldots \delta x_{i_k}], \tag{5.30}$$

for $p = 1, \ldots, m$.

### Variance-covariance

The $N$-th order approximation of the response variance-covariance in terms of central moments of the input variables is given by

$$\mathrm{cov}_N(R_p(x), R_q(x)) = \sum_{k,l=1}^{N} \frac{1}{k!\,l!} \sum_{\substack{i_1,\ldots,i_k=1 \\ j_1,\ldots,j_l=1}}^{n} \left( \frac{\partial^k R_p}{\partial x_{i_1} \ldots \partial x_{i_k}} \frac{\partial^l R_q}{\partial x_{j_1} \ldots \partial x_{j_l}} \right)_{E[x]}$$

$$(E[\delta x_{i_1} \ldots \delta x_{i_k} \delta x_{j_1} \ldots \delta x_{j_l}] - E[\delta x_{i_1} \ldots \delta x_{i_k}] E[\delta x_{j_1} \ldots \delta x_{j_l}]), \tag{5.31}$$

for $p, q = 1, \ldots, m$.

# Chapter 6

# Virtual Population Analysis with the adjoint method[1]

In this chapter an adjoint data assimilation technique for fisheries data analysis is investigated. Adjoint data assimilation is first described for general models and its relation to automatic differentiation is considered. As an application an adjoint system is then derived for the model underlying the well-known Virtual Population Analysis. The resulting estimation procedure provides a systematic and efficient method to obtain stock parameters from a time series of catch-by-age and effort data that can be extended in several directions as desired. The procedure has been implemented in a C++ program which performs well on data from the Pacific halibut *(Hippoglossus stenolepis)* fishery. Attention is also paid to the sensitivity of the estimated parameters to various error sources.

## 6.1 Introduction

In this chapter we investigate an adjoint data assimilation technique with the purpose of using it in a fisheries data analysis context. In particular we will focus our attention on age-structured stock assessment methods. Currently there is a large number of these age-structured methods available and we show how the adjoint technique can be used for these methods. For a review of age-structured stock assessment methods we refer to Megrey (1989).

We have applied the adjoint technique to the model underlying Virtual Population Analysis (VPA). This age-structured method for estimating stock sizes and fishing mortality uses a model that describes the cohort size dynamics backward in time.

[1]The text of this chapter is published in Huiskes (1998); for the purpose of this thesis a discussion on the relation of the adjoint method to automatic differentiation was added.

Adjoint data assimilation provides a systematic and efficient procedure for the estimation of parameters in complex models by comparing predictions from the model to real world observations. In section 6.2 we present a general introduction to the adjoint method. Here we also discuss the relation of the adjoint method to automatic differentiation and describe a method to investigate the sensitivity of the estimated parameters.

Next, in section 6.3, we give an overview of the model underlying VPA and apply the adjoint method to this model. The resulting adjoint data assimilation procedure was implemented in a C++ program. The program requires a time series of catch-by-age and effort data as input and returns as output estimates of stock sizes, a number of other fishery and stock parameters and associated sensitivity information. It is applied to data from the well-documented Pacific halibut (*Hippoglossus stenolepis*) fishery in section 6.4. In the concluding remarks of section 6.5 a number of possible extensions to the data assimilation method are discussed.

## 6.2 Data assimilation using the adjoint method

### 6.2.1 Introduction

The aim of data assimilation is to combine models and data as efficiently as possible. The most prominent areas of application are currently meteorology and oceanography, where data assimilation is used to combine numerical models and large data sets to improve the forecasts that can be obtained by these models. For reviews see for instance Ghil and Malanotte-Rizzoli (1991) and Lorenc (1986).

Data assimilation methods can roughly be divided into sequential and variational methods. Sequential methods, such as successive correction, optimal interpolation and kalman filtering, update the state of a model sequentially, i.e. for every new observation that becomes available. Variational methods, which include the adjoint method, achieve their aim of assimilating data by fitting the model as closely as possible to observational data, usually over a certain time interval, by variation of the model parameters. This is usually done by minimization of a function, called the cost or penalty function. This cost function quantifies the misfit between model and observations. A cost function can be constructed in several ways. One can, for instance, use Bayesian type of analysis to derive a function proportional to the posterior probability density of the parameters. In this paper we directly compare observations and their corresponding model predictions by using a least-squares approach.

The adjoint method is specifically aimed at the efficient computation of the gradient of the cost function for large models. Using this gradient, the parameters corresponding to the best fit can be found by a first derivative unconstrained minimization method, e.g. a Quasi-Newton or conjugate gradient method. See Gill et al. (1981) for an overview of these methods.

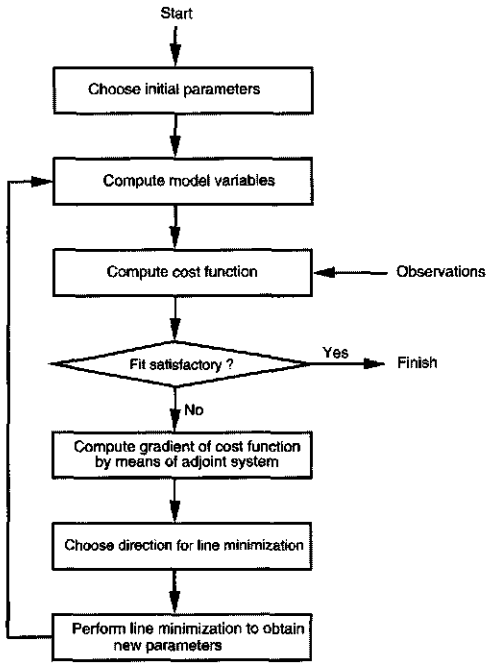



Figure 6.1: Data assimilation using the adjoint method.

To compute the gradient a so-called adjoint system for the model is derived. This can be done at several stages of the modeling process; see for instance Giering and Kaminski (1992) for a discussion. We will follow the approach described in Lawson et al. (1995) where the adjoint system is constructed from a model description that is close to computer code. The adjoint system is composed in reverse order compared to the system representing the model itself. The computation of the gradient is then equivalent to solving the adjoint system. This has two advantages compared to approximations using finite differences. First, it saves a considerable amount of runtime. It was shown in Baur and Strassen (1983) and Griewank (1989) that the computational cost of the evaluation of the adjoint model is only 2-5 times the computational cost of the cost function. Approximation of the gradient by finite differences takes at least $n+1$ computations of the cost function, where $n$ is the dimension of the parameter space, i.e. the number of parameters being estimated. Secondly, the solution of the adjoint system is the exact gradient. The adjoint method for data assimilation makes no assumptions regarding the linearity of the model. Only differentiability of the cost function in its arguments is required and a model that can be represented as a sequence of computations using differentiable functions. The structure of the data assimilation process using the adjoint method is summarized by the flow chart of Figure 6.1.

### 6.2.2 Adjoint systems

Let us consider a data vector $d$ from a data space $\mathcal{D}$. It is assumed that each element of this vector corresponds to an observation of a certain variable. Our aim is to estimate a vector of parameters $c$ (from a parameter space $\mathcal{C}$ with dimension $N_C$) for a model represented by a system of equations that iteratively produces a vector $x(c)$. This system will be discussed in more detail later. We assume that the vector $x(c)$ can be used to compute predictions for the observed variables in $d$.

To find the most suitable values for the parameter vector we assume that we have a cost function $J(d, x(c), c)$, mapping into $\mathbb{R}_0^+$, the values of which are smaller for parameters corresponding to predictions in better agreement with the observations.

Given the data variables we want to minimize $J$ with respect to the model parameters, i.e. our problem will be to find

$$\min_{c \in \mathcal{C}} J(d, x(c), c). \tag{6.1}$$

Since $x(c)$ depends on the parameter vector $c$ through an entire system of equations, it is difficult to obtain the derivative of this cost function. An approach based on Lagrange multipliers is used to tackle this problem.

We use a general model structure that is similar to a computer code representation. It is assumed that the model can be represented by the following system of equations

$$x_n = f_n(x_1, \ldots, x_{n-1}, c), \qquad n = 1, \ldots, N, \tag{6.2}$$

i.e. the model is represented by a sequence of computations where at each step a new scalar variable is computed which may depend upon all previously computed variables and the parameter vector $c$. The computations can be considered as corresponding to assignment statements in a computer program.

Now define the Lagrangian function

$$L(x(c), c, \lambda) = J(x(c), c) - \sum_{n=1}^{N} \lambda_n (x_n - f_n(x_1, \ldots, x_{n-1}, c)), \tag{6.3}$$

where $\lambda_1, \ldots, \lambda_N$ are the *adjoint variables* and we have dropped the dependency of $J$ on $d$ from our notation as $d$ is assumed to be constant from here on. If $x_1, \ldots, x_N$ are given by the equations (6.2), it follows that

$$D_c L(x(c), c, \lambda) = D_c J(x(c), c). \tag{6.4}$$

The operator $D_c$ denotes the total derivative with respect to $c$. This relation holds for all $\lambda$, since the two functions of which the derivatives are taken, are in fact the same for all $\lambda$. Notice that this is the case even if we choose the adjoint variables to be dependent on the parameter vector $c$. This is exactly what we will do, i.e. we use $\lambda = \lambda(c)$. Since we are only interested in $D_c J(c, x(c))$, we are now free to choose the adjoint variables in such a way that it becomes easier to calculate $D_c L$ which is the same as $D_c J$. Observe that

$$(D_c L)_k = \frac{\partial L}{\partial c_k} + \sum_{n=1}^{N} \left( \frac{\partial L}{\partial x_n} \frac{\partial x_n}{\partial c_k} + \frac{\partial L}{\partial \lambda_n} \frac{\partial \lambda_n}{\partial c_k} \right), \tag{6.5}$$

for $k = 1, \ldots, N_{\mathcal{C}}$ and that $\frac{\partial L}{\partial \lambda_n} = 0$ if and only if the system equations (6.2) are satisfied. It follows that if we choose the adjoint variables such that $\frac{\partial L}{\partial x_n} = 0$ for

$n = 1, \ldots, N$, we then find using (6.4) and (6.5) that

$$(D_c J)_k = (D_c L)_k = \frac{\partial L}{\partial c_k}. \tag{6.6}$$

We have

$$\frac{\partial L}{\partial x_N} = \frac{\partial J}{\partial x_N} - \lambda_N \tag{6.7}$$

and

$$\frac{\partial L}{\partial x_n} = \frac{\partial J}{\partial x_n} - \lambda_n + \sum_{i=n+1}^{N} \frac{\partial f_i}{\partial x_n} \lambda_i, \qquad n = (N-1), \ldots, 1. \tag{6.8}$$

So it follows that $\frac{\partial L}{\partial x_n} = 0$ if and only if we choose $\lambda_1, \ldots, \lambda_N$ according to the following *adjoint equations*

$$\lambda_N = \frac{\partial J}{\partial x_N}, \tag{6.9}$$

$$\lambda_n = \frac{\partial J}{\partial x_n} + \sum_{i=n+1}^{N} \frac{\partial f_i}{\partial x_n} \lambda_i, \qquad n = (N-1), \ldots, 1. \tag{6.10}$$

Together these adjoint equations constitute the *adjoint system.* Notice that the adjoint variables depend on the parameters since all partial derivatives are evaluated at the parameter vector $c$ for which the derivative is being computed. Using equation (6.6) we find the following expression for the derivative of the cost function

$$(D_c J)_k = \frac{\partial J}{\partial c_k} + \sum_{n=1}^{N} \frac{\partial f_n}{\partial c_k} \lambda_n, \qquad k = 1, \ldots, N_C, \tag{6.11}$$

where the adjoint variables are to be computed by means of the adjoint system.

### 6.2.3 Implementation

We present a construction procedure for the cost function gradient that implicitly uses the adjoint system described above. Its description allows an easy translation into computer code.

For every model variable $x_n$ we have an associated adjoint variable $\lambda_n$. We also introduce for every parameter $c_k$ a corresponding variable $\gamma_k$. At the end of the procedure these variables $\gamma_k$ must contain the corresponding components of the cost function gradient , i.e. we intend that

$$\gamma_k = (D_c J)_k = \frac{\partial J}{\partial c_k} + \sum_{n=1}^{N} \frac{\partial f_n}{\partial c_k} \lambda_n, \qquad k = 1, \ldots, N_C. \tag{6.12}$$

First initialize $\lambda_n$ and $\gamma_k$ by setting

$$\lambda_n := \frac{\partial J}{\partial x_n}, \qquad n = 1, \ldots, N, \tag{6.13}$$

and

$$\gamma_k := \frac{\partial J}{\partial c_k}, \qquad k = 1, \ldots, N_C. \tag{6.14}$$

We consider equations (6.2) one by one in reverse direction, updating the variables at each step. When finished, $\lambda_n$ and $\gamma_k$ will satisfy adjoint equations (6.9), (6.10) and equation (6.12), respectively. First notice that $\lambda_N$ already satisfies (6.9). Furthermore it can be seen that each $\lambda_n$ depends only on model equations and adjoint variables with an index larger than $n$. Thus, when arriving at equation $n$ we may use the following construction rules

- If $\frac{\partial f_n}{\partial x_i}$ is not equal to zero, then perform an update statement that adds $\frac{\partial f_n}{\partial x_i}\lambda_n$ to $\lambda_i$, i.e.

$$\lambda_i += \frac{\partial f_n}{\partial x_i}\lambda_n, \qquad i = 1, \ldots, n-1, \tag{6.15}$$

  where the $+=$ assignment operator indicates that righthand side is to be added to the lefthand side.

- If $\frac{\partial f_n}{\partial c_k}$ is not equal to zero, then perform an update statement that adds $\frac{\partial f_n}{\partial c_k}\lambda_n$ to $\gamma_k$, i.e.

$$\gamma_k += \frac{\partial f_n}{\partial c_k}\lambda_n, \qquad i = 1, \ldots, N_C. \tag{6.16}$$

Notice the similarity in form of these two construction rules. After proceeding in this manner for $N$ steps, $\lambda_n$ and $\gamma_k$ will satisfy (6.9), (6.10) and (6.12) as required.

To summarize, each model assignment will correspond to a number of assignments that update adjoint variables of smaller index and variables $\gamma_k$. Usually a model assignment depends on only a few of the system and parameters, thereby reducing the number of update assignments required. The sequence of initialization and update statements together constitute the procedure to obtain the cost function gradient.

### 6.2.4 Relation of the adjoint method to automatic differentiation

As may be verified by inspection the adjoint system (6.9), (6.10) describing the adjoint variables is essentially equal to the equation (2.28) of section 2.5. In the latter expression the dependency relations between the variables are described in somewhat greater detail. Also the Algorithm 2.5.1 is a more precise description of the reverse updating scheme corresponding to equations (6.15) and (6.16) to obtain (6.10). This means that the adjoint method formulation based on the use of the Lagrange function can be considered to be an alternative description of reverse automatic differentiation. The adjoint variables defined in Chapter 2 thus have an interpretation as Lagrange multipliers corresponding to the objective function $\bar{y}y$, constrained by the system of equations (2.21) representing the evaluation procedure.

In Thacker (1991) a number of memory storage aspects are discussed for the case that adjoint data assimilation is applied to computer codes of the size generally used in meteorology and oceanography. The application of automatic differentiation to numerical integration algorithms is discussed in Eberhard and Bischof (1996).

### 6.2.5 Sensitivity analysis

To investigate the sensitivity of the estimated parameters to the input data, we adopt a novel approach in fisheries analysis based on the implicit function theorem.

The optimal parameters $\tilde{c}$ for some data vector $\tilde{d}$ satisfy

$$D_c J(\tilde{d}, x(\tilde{c}), \tilde{c}) = 0. \tag{6.17}$$

By the implicit function theorem, applied to this vector equation stating that the derivative $D_c J(\tilde{d}, x(\tilde{c}), \tilde{c})$ is zero, there exists a function $c = g(d)$ in the neighbourhood of $\tilde{d}$ which satisfies $\tilde{c} = g(\tilde{d})$ and for which

$$D_d g(\tilde{d}) = -(D_c^2 J(\tilde{d}, x(\tilde{c}), \tilde{c}))^{-1} (D_{dc}^2 J(\tilde{d}, x(\tilde{c}), \tilde{c})). \tag{6.18}$$

Using this linear mapping one can approximate changes $\delta c$ in the parameter vector corresponding to the optimal fit as a result of changes $\delta d$ in the data vector by means of

$$\delta c = D_d g(\tilde{d}) \delta d. \tag{6.19}$$

To find the directions in the data space $\mathcal{D}$ which are particularly important in their effect on the parameters a singular value decomposition of $D_d g(\tilde{d})$ may be computed. Notice that to obtain $D_d g(\tilde{d})$ the inverse of the Hessian of $J$ with respect to $c$, i.e. $(D_c^2 J(\tilde{d}, x(\tilde{c}), \tilde{c}))^{-1}$, must be computed. This method is described in more detail in Chapter 3 (section 3.6).

## 6.3 Virtual Population Analysis

### 6.3.1 Standard VPA

The model to which we apply the adjoint method is the model underlying the well-known VPA (Gulland, 1965). We will first give a short overview of this model and the way in which this model is used in VPA to obtain estimates for fishing mortality and population sizes.

The dynamics of the size of a single cohort, i.e. a subpopulation of fish born in the same year, is assumed to be described by

$$N(y+1, a+1) = N(y, a) - C(y, a) - D(y, a). \tag{6.20}$$

This equation states that the number of fish in a cohort at the beginning of year $y+1$ is equal to the number of fish at the beginning of year $y$ minus the number

of fish caught during that year, $C(y,a)$, and minus the number of fish that died from natural causes, $D(y,a)$.

Fishing mortality during a year is modeled by assuming for each fish a fixed probability of being caught per unit of time depending on its age. Natural mortality is modeled in the same way with a fixed probability of dying from natural causes. These probabilities are denoted by $F(y,a)$ and $M(y,a)$, respectively. Usually it is assumed that natural mortality has a fixed known value $M$ independent of year and age. We then have

$$\frac{dC}{dt} = F(y,a)N(t) \qquad \text{and} \qquad \frac{dD}{dt} = MN(t), \tag{6.21}$$

and consequently,

$$\frac{dN}{dt} = -(F(y,a) + M)N(t), \tag{6.22}$$

where $t$ is time during year $y$ with $N(0) = N(y,a)$ and $N(1) = N(y+1, a+1)$. We obtain

$$N(t) = N(0)e^{-(F(y,a)+M)t}, \tag{6.23}$$

so

$$N(y,a) = e^{(F(y,a)+M)}N(y+1,a+1). \tag{6.24}$$

From (6.21) and (6.23) it follows that catch during year $y$ for a certain age group satisfies

$$\frac{dC}{dt} = F(y,a)e^{-(F(y,a)+M)t}N(y,a), \tag{6.25}$$

with $C(0) = 0$ giving

$$C(t) = \frac{F(y,a)}{F(y,a)+M}(1 - e^{-(F(y,a)+M)t})N(y,a). \tag{6.26}$$

Using $C(1) = C(y,a)$ and assuming that relation (6.24) holds, we obtain the *catch equation*:

$$C(y,a) = \frac{F(y,a)}{F(y,a)+M}(e^{(F(y,a)+M)} - 1)N(y+1,a+1). \tag{6.27}$$

It is assumed that data for the catch $C(y,a)$ are available in a catch-at-age table for years in the range $y_{\min}$ to $y_{\max}$ and ages in the range $a_{\min}$ to $a_{\max}$. If we have an estimate for $N(y+1,a+1)$, we can use the catch equation to obtain the fishing mortality $F(y,a)$. Once $F(y,a)$ is computed, the population size $N(y,a)$ follows from (6.24). VPA consists of a sequence of such computations for each cohort which ends when no more catch data are available for the cohort under consideration. This happens when the minimum age (usually the age of recruitment to the fishery) $a_{\min}$ or the minimum year $y_{\min}$ is reached. To obtain a starting estimate for the cohort size at the maximum age or maximum year for which the catch $C(y,a)$ is available, one usually assumes that the terminal fishing mortality $F_{\text{term}}$, i.e. the fishing mortality for that maximum age or year, is known. The final cohort

size $N(y,a)$, which is either of the form $N(y_{\max},a)$ or $N(y,a_{\max})$ depending on whether the data corresponding to the cohort under consideration is limited due to a maximum age or a maximum year, can then be computed from

$$N(y,a) = \frac{(F_{\text{term}}+M)C(y,a)}{F_{\text{term}}(1-e^{-(F_{\text{term}}+M)})}. \tag{6.28}$$

Notice that in general population sizes for the maximum year $y_{\max}$ are especially interesting, since they correspond to estimates for the most recent stock size. However, as we see here, in standard VPA the estimates for these stock sizes are based on only one catch observation $C(y_{\max},a)$ and a terminal fishing mortality $F_{\text{term}}$, so these values cannot be considered to be very reliable.

### 6.3.2 Adjoint VPA

In our approach, which for convenience we shall call adjoint VPA, we will also use equation (6.24) to describe the cohort size dynamics. However, parameters are now obtained by minimization of a cost function instead of the sequential procedure used in standard VPA. In fact, the method is quite similar to statistical catch-at-age methods that maximize likelihood functions as described in for instance Deriso et al. (1985) and Megrey (1989). We assume that besides catch-at-age data for years $y_{\min}$ to $y_{\max}$ and age groups $a_{\min}$ to $a_{\max}$, an indicator of fishing effort is available for every year. We further use the common assumption that the fishing mortality is a product of an age dependent term and a year dependent term:

$$F(y,a) = q(a)E(y), \tag{6.29}$$

where $E(y)$ is the indicator of fishing effort during year $y$ and $q(a)$ is an age dependent catchability coefficient. This catchability coefficient determines the relation between the effort and the actual fishing mortality of the different age groups. The product of (6.29) is well-defined if we assume that the effort indicator has a fixed value, say 100, for $y_{\min}$ and that values for subsequent years are expressed in an index relative to the first year. Instead of using a fixed terminal fishing mortality as in VPA we now introduce the final cohort sizes as additional parameters $N_{\text{fin}}(y,a_{\max})$ and $N_{\text{fin}}(y_{\max},a)$. Thus, we try to estimate the parameters

- $q(a)$, for $a = a_{\min},\ldots,a_{\max}$,
- $E(y)$, for $y = (y_{\min}+1),\ldots,y_{\max}$,
- $N_{\text{fin}}(y,a_{\max})$, for $y = y_{\min},\ldots,y_{\max}$,
- $N_{\text{fin}}(y_{\max},a)$, for $a = a_{\min},\ldots,(a_{\max}-1)$,

such that the difference between model predictions and available observations is as small as possible. This is achieved by minimizing the cost function

$$J = \frac{1}{2}\Big(\sum_{y,a}(C_{\text{data}}(y,a)-C_{\text{model}}(y,a))^2 + \kappa\sum_{y}(E_{\text{data}}(y)-E_{\text{model}}(y))^2\Big), \tag{6.30}$$

where $\kappa$ is a weighting factor which quantifies the relative weight attributed to effort observations relative to the catch observations. More advanced weighting schemes are discussed in Chapter 5. Based on the ideas presented in section 6.2 we proceed with the following algorithm:

1. Take an initial guess for the parameters to be estimated.

2. Apply the cohort dynamics model represented by (6.24).

3. Compare the observations to the model predictions using the cost function (6.30). Model predictions for $C(y,a)$ are obtained from (6.26).

4. Compute the gradient of the cost function for the chosen set of parameter values using the adjoint method.

5. Use this derivative information to find a direction in which to minimize the cost function.

6. Use a line minimization routine to find a new set of parameters corresponding to a better fit to the observations.

7. Repeat step 2. through 6. until the best fit has been found.

We have implemented this algorithm in C++ code using the procedure described in 6.2.3 to compute the gradient of the cost function.

To find a direction in parameter space in which to perform a line minimization the Polak-Ribiere conjugate gradient method is used, see Gill et al. (1981). Line minimization is achieved by means of Brent's method (Brent, 1973). The code returns the optimal parameters $q(a)$, $E(y)$, $N_{\text{fin}}(y, a_{\max})$ and $N_{\text{fin}}(y_{\max}, a)$ for all years and ages under consideration. Also the matrices required to perform the sensitivity analysis described in section 6.2.5 are computed.

## 6.4 Results for the Pacific halibut fishery

We apply the adjoint VPA procedure to data from the Pacific halibut fishery and compare the results to those from standard VPA. We use catch-at-age data of both the setline and the trawl fishery in International Pacific Halibut Commission regulatory Area 2 from 1935 to 1976 and age classes 8 to 20 as reported in Hoag and McNaughton (1978). Effort data for these years are taken from Hoag et al. (1983). As natural mortality a fixed value of $M = 0.2$ is chosen for all classes, i.e. it is not chosen as a parameter to be estimated. In general the available data is not sufficient to estimate both fishery and natural mortality, as also our study will confirm. This means that an estimate of the natural mortality must be obtained by alternative means.

### 6.4.1 Performance of adjoint VPA

The optimization procedure performs well as can be seen in Figure 6.2. The procedure converges to the optimal parameter vector in about 30 to 100 iterations. The cost function is reduced to approximately 15 percent of its value for the initial parameters. The fit of the model predictions to the catch observations depends on the value of the weighting factor $\kappa$. For small values of $\kappa$ the effort parameters $E(y)$ can freely be chosen without substantially affecting the cost function. This results in a good fit to the catch data. The average absolute error of the catch predictions compared to the observations is 18%. Figure 6.3 shows the fit to the catch data for $\kappa = 0$.

Since the effort observations are not used in this analysis, it can be expected that the effort values found for the optimal fit differ substantially from the observed values for the effort indicators. This is shown in Figure 6.4.

Choosing larger values of $\kappa$ will make the optimal effort parameters come closer to the observed values, see again Figure 6.4. For very large values of $\kappa$ the effort parameters are forced to correspond to their observed values, leaving less freedom to fit to the catch data. This is shown in Figure 6.5. The average absolute error of the catch predictions compared to the observations is now 20%.

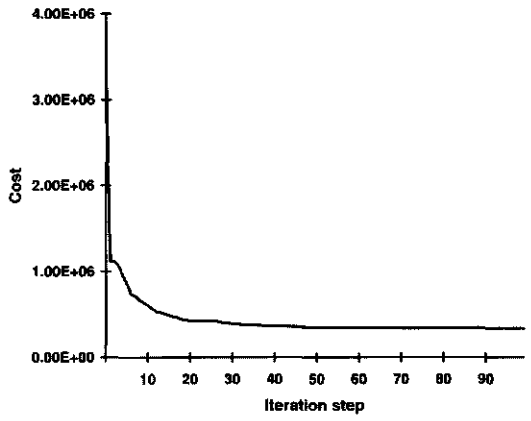



Figure 6.2: Cost versus iteration step. In 100 iterations 108 variables are estimated. The cost function value for the optimal parameters is about 15% of the cost for the initial parameters.

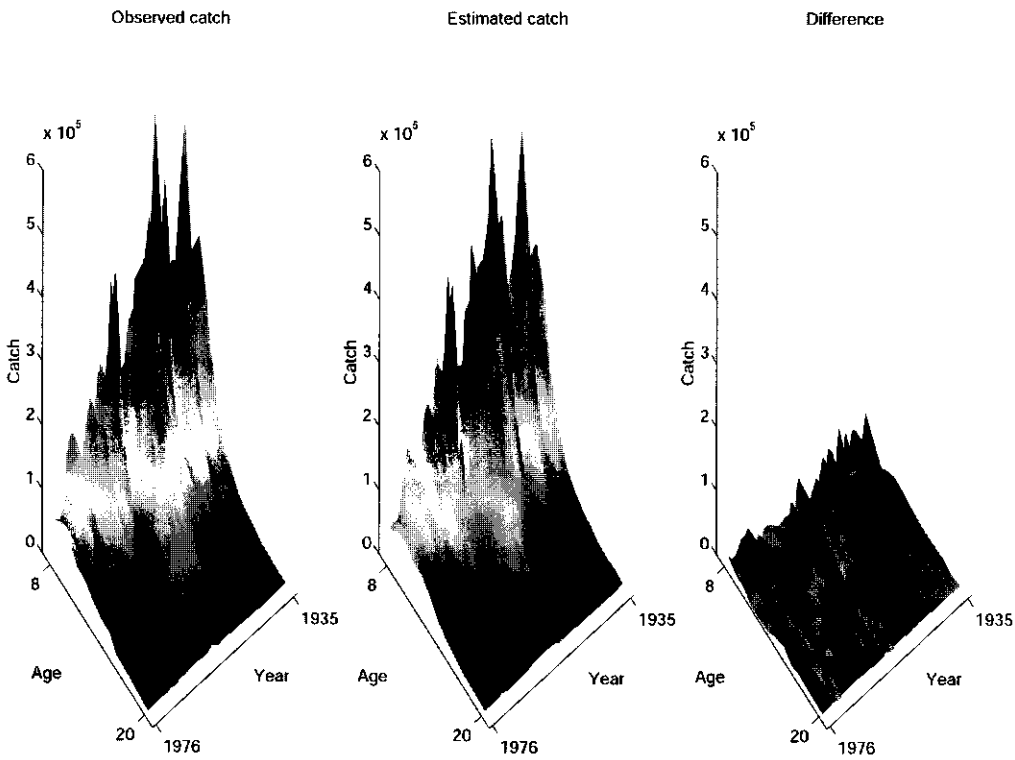



Figure 6.3: Fit to catch observations for $\kappa = 0$. The average absolute difference relative to the observed catch is 18%.

### Comparison to standard VPA

Estimated population sizes are close to those obtained by standard VPA. Due to separation of fishing mortality into effort and catchability coefficients, adjoint VPA cannot predict observed catches exactly. To get an idea of the performance of the catch fitting procedure, we compare our results to those obtained for a standard VPA that is also submitted to a separability assumption (6.29). To this end, first a standard VPA is performed. To obtain catchability coefficients every fishing mortality $F(a, y)$ is divided by its corresponding effort indicator $E(y)$. The resulting catchability coefficients depend on both year and age. To obtain catchability coefficients which depend on age only, they are averaged over time. These coefficients can be used to obtain new fishing mortality coefficients using (6.29). The catches corresponding to these fishing mortality coefficients are no longer the same as the observed catches.

It turns out that the cost function for this set of parameters is about six times as large as the value obtained by adjoint VPA. The average absolute error between predicted catches and observations is now 35% which is about twice as large as

the average absolute error relative to the observed catch for adjoint VPA.

In Figure 6.6 stock size estimates for the 1950 cohort are compared for standard VPA, adjoint VPA and the approach described above using averaged catchability coefficients, referred to as *restricted* VPA.

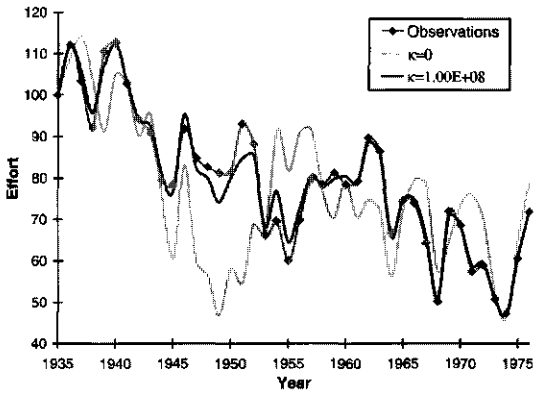



Figure 6.4: Effort comparison between observations and parameter estimates. For $\kappa = 0$ deviation from observations is largest. For larger values of $\kappa$ parameter estimates and observations are in better agreement. For $\kappa = 1.0 \cdot 10^{10}$ parameter estimates and observations are equal.

### 6.4.2 Sensitivity considerations

We have repeated our procedure for various values of the natural mortality parameter and found no significant change in the minimum value of the cost function; see Table 6.1.

| $M$ | Minimum Cost |
|---|---|
| 0.05 | 341582 |
| 0.10 | 339790 |
| 0.15 | 340219 |
| 0.20 | 341172 |
| 0.25 | 342215 |
| 0.30 | 343020 |

Table 6.1: Optimal cost function values for various natural mortality coefficients. Relative changes in the optimal cost function values are small for variations of the natural mortality coefficient in the value region of interest.

We conclude that additional information is needed to estimate this parameter or that it should be estimated by other means. We have also performed the sensitivity analysis described in 6.2.5. To test which catch-at-age table entries are particularly important in their effect on the optimal parameters, all parameters are first scaled to the same order of magnitude. Next we use the linear mapping

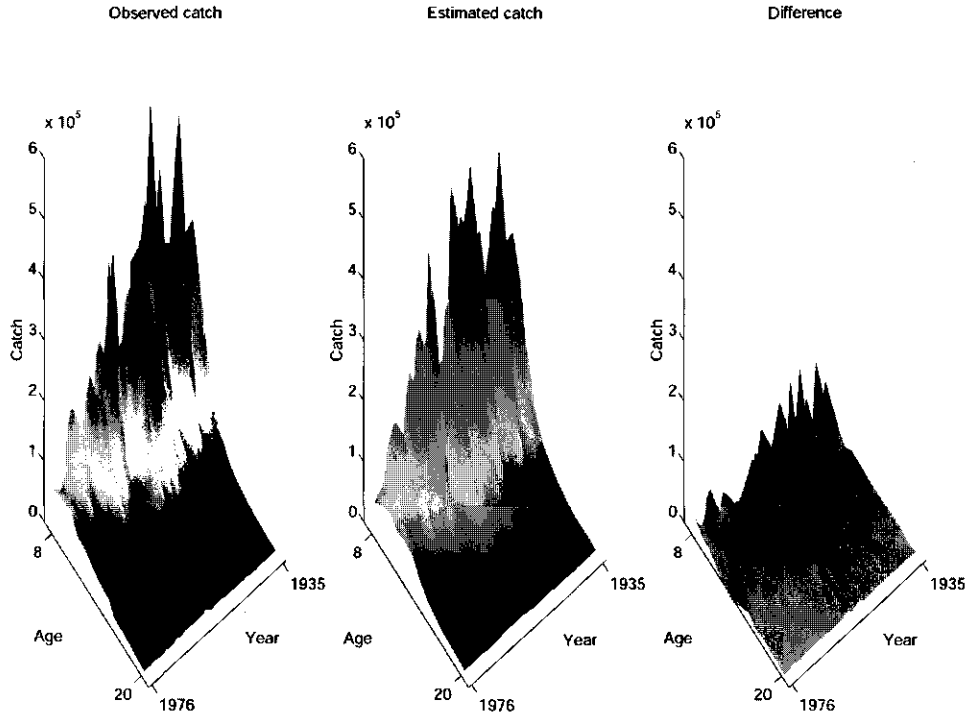



Figure 6.5: Fit to catch observations for $\kappa = 1.0 \cdot 10^{10}$. The average absolute difference relative to the observed catch is 20%.

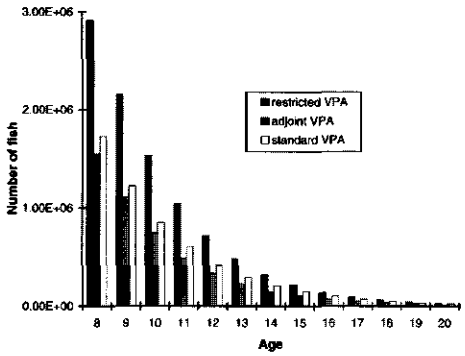



Figure 6.6: Comparison between restricted VPA and adjoint VPA. Large relative errors in predicted catches for restricted VPA also result in large deviations in predicted population sizes from those obtained by standard and adjoint VPA.

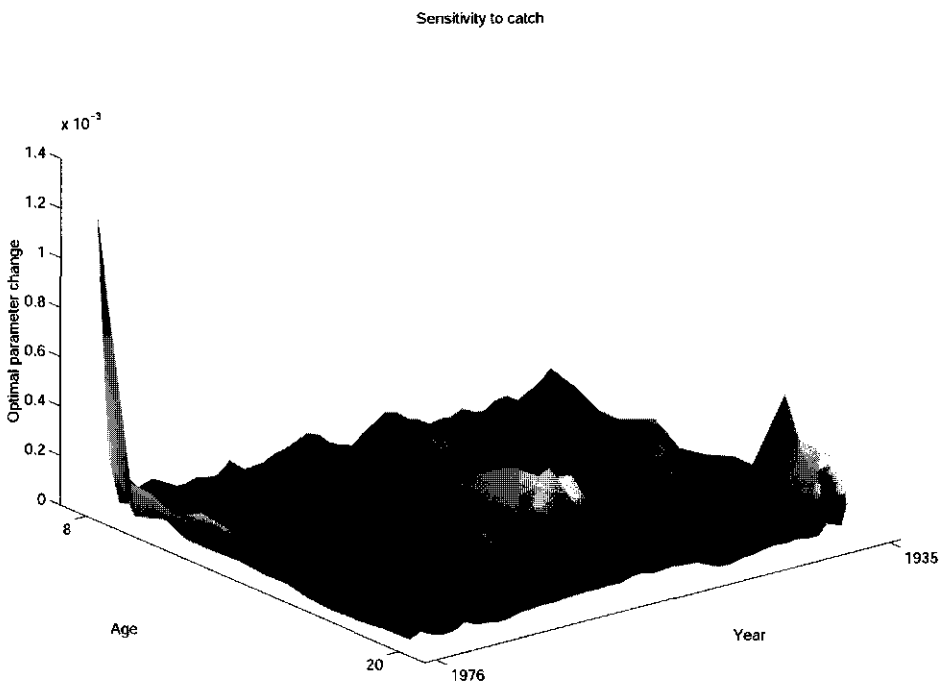



Figure 6.7: Parameter vector sensitivity to catch observation changes for $\kappa = 0$. For each observation the norm of the vector change in the scaled optimal parameters due to a unit change in the observation value is plotted. The entries in the catch-at-age table corresponding to incomplete cohorts have the largest impact on the optimal parameters. For $\kappa = 0$ changes in the effort observations have no effect.

from (6.18) to investigate the change in the parameters resulting from a change in the catch data. For each catch-at-age table entry the norm of the difference in the optimal parameters as a result of a unit change in that entry is plotted. See Figure 6.7 and Figure 6.8 for $\kappa = 0$ and $\kappa = 1.0 \cdot 10^{10}$, respectively.

Notice that catch measurement errors for table entries corresponding to incomplete cohorts cause the largest change in optimal parameter values. A similar analysis is performed for the effort time series. Measurement errors in effort observations have a negligible effect on the optimal parameters for $\kappa = 0$. They do however become important for large values of $\kappa$. See Figure 6.8. The optimal parameter that is mainly affected by a change in an effort observation is, as may be expected, the effort parameter corresponding to this observation.

Next, as another test, all catch observations are increased by 1%. We can compute the resulting relative change in the optimal parameter vector. It turns out that most parameters change by approximately 1%. Some of the final cohort sizes change by up to 3%. For $\kappa = 1.0 \cdot 10^{10}$ the changes are somewhat smaller.

Using a singular value decomposition of the linear mapping of (6.18) we can find the directions in the data space that result in the largest change of the parameters.

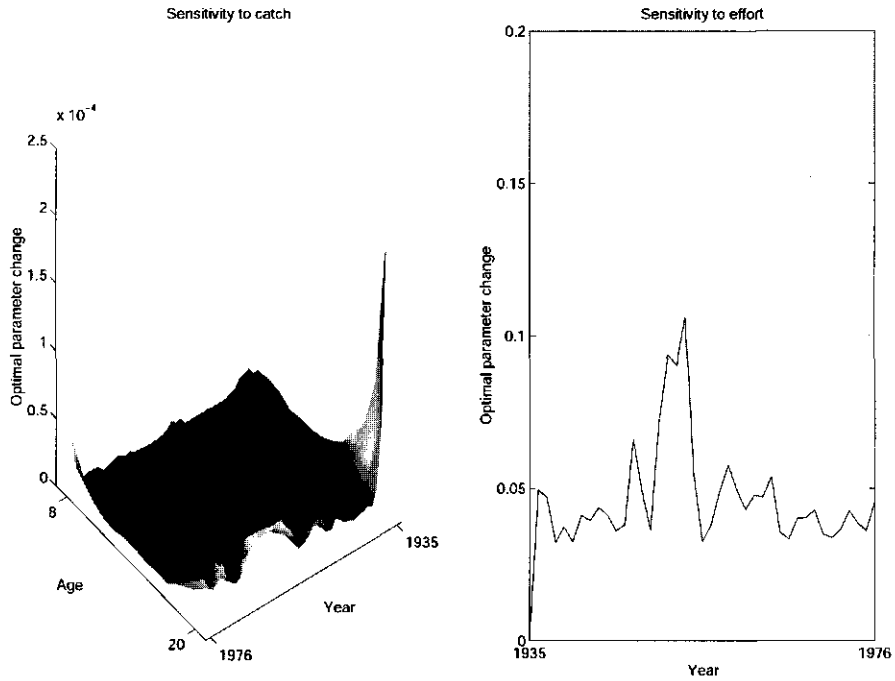



Figure 6.8: Parameter vector sensitivity to catch and effort observation changes for $\kappa = 1.0 \cdot 10^{10}$. Impact of a unit change in the catch entries is now smaller than for $\kappa = 0$. Especially the effort parameters are now hardly affected by changes in catch observations, since they are determined by the effort observations. Changes in effort variables now have a substantial effect.

If we use an observation error vector of length equal to the vector used in the example where all catches were increased by 1%, the resulting changes in the parameters are very large. This is as expected however, since this error observation vector corresponds to large relative changes in some of the catch observations.

Changing the effort observations by 1% has no effect for $\kappa = 0$ and results in a change of exactly 1% for $\kappa = 1.0 \cdot 10^{10}$.

## 6.5 Concluding remarks

We have implemented a procedure for estimating fishery parameters using the same hindcasting model as in VPA. Our procedure is similar to a number of statistical age-structured methods that are currently employed for stock assessment. For these methods the adjoint method can be a useful technique for the estimation of the parameters. The main advantage of the adjoint method is its efficient and reliable computation of the cost function gradient which in turn allows the simultaneous estimation of a large number of parameters.

This chapter describes how to implement the adjoint method for relatively simple models. One advantage of deriving an adjoint model by hand is that it provides very efficient code. For more complex models or models that often have to be changed it is usually more practical to use software packages for reverse automatic differentiation, even though these may produce code that is slightly less efficient. For an overview of automatic differentiation algorithms, see Chapter 2 of this thesis.

Adjoint VPA performed well on data from the Pacific halibut fishery. For these data a total of 108 parameters are estimated in about 100 iterations. The results have been used in Grasman and Huiskes (2001) for a stochastic model of halibut recruitment.

The procedure can be extended in many ways and this study should be considered to be exploratory in nature. The approach taken here is based on least-squares parameter estimation without further statistical considerations, even though an indication of the accuracy of the parameters is obtained by investigation of the senstivity of the estimates with respect to the observation values. A more statistical approach to parameter estimation and inference is described in Chapter 4 of this thesis. Here also the influence of model nonlinearity and model complexity on the accuracy of parameter estimates are investigated. In Chapter 5 a case-study is presented in which these ideas are applied. Here also the use of a lognormal distribution function for the error structure is described, and more advanced weighting schemes are discussed.

An interesting possibility lies in the fact that the code for minimizing a cost function can be adapted to find extremes of arbitrary functions depending on the system and model parameters. As a simple example one could consider the definition of a function that describes the total catch, perhaps subject to a number of constraints, and maximize this function with respect to the effort variables. To compute the gradient of such a function would require only a change in the initialization part of the adjoint method code.

# Chapter 7

# Automatic differentiation for modern nonlinear regression[1]

For modern nonlinear regression routines, the efficient computation of first and higher-order derivatives is highly important. Automatic differentiation constitutes an opportunity to achieve both higher run-time efficiency and an increased feasibility of higher-order uncertainty analysis of complex models. In this article we present an overview of the derivative requirements of nonlinear regression routines. We further describe our experience in developing a C++ library for model analysis that uses the ADOL-C package for automatic differentiation. We show how the model analysis library, named MAP, has benefited from using automatic differentiation. In addition, a number of experiments are presented that show how more flexible and efficient execution trace management could further enhance the ease-of-use of ADOL-C.

## 7.1 Introduction

The ideas expressed in this chapter are based on experience obtained in developing a software library, named MAP, for nonlinear regression and subsequent uncertainty analysis. MAP is aimed at the analysis of models that are represented by smooth maps, and are provided as computer code. MAP consists of routines for model fitting, parameter uncertainty analysis, propagation of uncertainty information, model selection and various types of sensitivity analysis. The C++ library has a simple object-oriented structure, and all routines become operative by the definition of a `MAP` object. Some examples of calls to MAP routines are shown in Figure 7.1.

The library may be directly linked to user programs, or be employed in combination with a graphical user interface (GUI), which can be used to control the

[1]The text of this chapter is published in Huiskes (2001); for the purpose of this thesis, the MAP library is described here in some additional detail.

```
MAP->Estimation->Auto();
MAP->PU->ComputeBias();
MAP->Propagate->Covariance();
```

Figure 7.1: Examples of calls to routines from the MAP library.

various routines and view the resulting output. Figure 7.2 shows a screenshot of the graphical user interface.

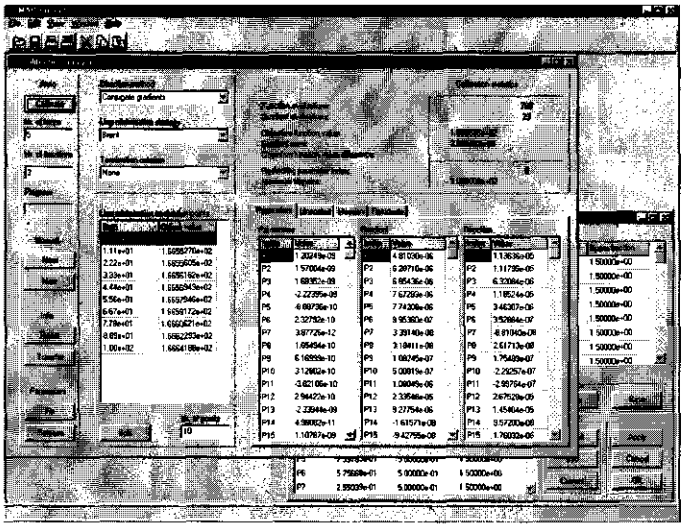


Figure 7.2: Screenshot of the graphical user interface for MAP.

In case the GUI is used, a new model can be analyzed by adapting two files: (i) the `MAP_Model` file in which the model is defined, and (ii) the `MAP_Data` file in which the dimensions of the problem, the initial estimate for the parameters, and the regressor variables and associated observations are specified.

For the computation of first order and higher-order derivatives MAP relies on the ADOL-C library (Griewank et al., 1996). This automatic differentiation tool is based on operator overloading. An introduction to the automatic differentiation terminology used in this chapter is given in Chapter 2. Further information about MAP can be found at: `http://www.cwi.nl/~markh`.

In section 7.2, we present a characterization of modern nonlinear regression. Particular emphasis is placed on the type of derivative information required by the various routines. In section 7.3 we delve deeper into our experience with the ADOL-C library and describe a number of experiments that indicate of the performance of the system. For these experiments we have linked a fishery stock

assessment model to the library. This model is used in practice by the International Council for Exploration of the Sea (ICES) to reconstruct the population dynamics of North Sea herring (Patterson and Melvin, 1996).

## 7.2 Elements of modern nonlinear regression

### 7.2.1 Introduction

In nonlinear regression, model parameters are estimated by fitting a model to a set of observations. This is usually achieved by means of a likelihood function maximization. In case of a normal error structure for the difference between observations and model estimates, the maximum likelihood method is equivalent to least squares minimization. After a transformation, that makes the error vector components of $\varepsilon$ independent and identically distributed, we have

$$y = f(\theta) + \varepsilon, \qquad \varepsilon \sim N(0, \sigma^2 I_n), \tag{7.1}$$

constituting a parameterized set of $n$ dimensional random vectors $y$, which are candidates for the description of the true random observation vector. See Chapter 4 for a more detailed discussion of this type of estimation problem. The components of the function $f : \mathcal{P} \rightarrow \mathcal{S}$ from $p$ dimensional parameter space $\mathcal{P}$ to $n$ dimensional sample space $\mathcal{S}$ can be computed by means of a function that will be referred to as the *model map*. The model map, which will be described in more detail in section 7.3, may also depend on variables that are not estimated: the *regressor variables*. It is assumed that for a certain parameter vector $\theta^*$, the maximum likelihood estimator $\hat{\theta}$ corresponds to the global minimum of the objective function $S(\theta)$:

$$S(\theta) = \|y - f(\theta)\|^2. \tag{7.2}$$

The analysis of a nonlinear regression problem may be characterized as a process consisting of three elements:

1. Parameter estimation by fitting the model to the data. Unlike for linear regression the parameter estimate cannot be computed directly as a solution of the normal equations, but is obtained through an iterative optimization process.

2. Statistical inference, which is traditionally based on linearization of the model around the parameter estimate. Several computational procedures have been developed to take also higher-order derivatives into account in the computation of the parameter estimation bias, the variance-covariance matrix, likelihood contours and confidence regions.

3. Model structure analysis. The assumptions of the model must be checked, e.g. the residual error structure. Further the sensitivity of the results with respect to the model structure and the observations must be investigated. Model selection criteria can aid in evaluation of the appropriateness of the complexity of the model structure in relation to the available data.

In the following sections we discuss the requirements of each of the nonlinear regression procedure elements with respect to derivative information.

### 7.2.2 Parameter estimation

The computational cost of the iterative optimization process depends on the complexity of the model to be fitted, i.e. the evaluation cost of the objective function depends mainly on the evaluation cost of the model map. For problems with a large number of parameters, derivative information is required for efficient optimization. With automatic differentiation using the reverse mode, the evaluation cost of the objective function derivative is only a small multiple of the evaluation cost of the objective function itself (Griewank, 2000).

The nature of the derivative information that is required depends on the optimization algorithm chosen. For the conjugate gradient and quasi-Newton methods, the derivative of the model map or objective function with respect to the parameters is required. Newton type methods also require the second derivative with respect to the parameters.

In the case that the number of parameters is large, the problem generally must be scaled or preconditioned to ensure convergence of the optimization procedure. In this case derivatives with respect to the transformed parameters are required.

In MAP, scaling of the parameters can be handled from the graphical user interface. The optimization process can be performed automatically, but also manually if this is required in order to monitor the process step by step.

### 7.2.3 Statistical inference

The traditional statistical inference for nonlinear regression that is based on linearization, may be extended using higher-order model derivatives. These derivatives can be used both to obtain more accurate parameter uncertainty information, and to obtain inference estimates in case a non-normal error structure is assumed. Seber and Wild (1989) present an overview of higher-order corrections to linear parameter inferences. These are mainly based on the curvature measures described in Bates and Watts (1980). Hamilton (1986) and Draper and Smith (1981) developed second order inference methods. Computational methods for third order accurate $p$-values, used to obtain confidence intervals for both normal and non-normal error structures, are presented in Fraser (1999).

The corrections to the parameter uncertainty estimates implemented in MAP are based on curvature arrays that quantify the nonlinearity of the model around the parameter estimate. The curvature arrays constitute normal and tangential projections of second order model map derivatives. In Chapter 4 a detailed overview has been presented of methods of statistical inference based on curvature arrays. In addition, a method of Clarke (1987b) is implemented that provides likelihood contours for individual parameters based on marginal curvatures. This method is described in more detail in Chapter 5.

### 7.2.4 Model structure analysis

Model structure analysis consists of more than checking the assumptions with respect to the assumed error structure by means of diagnostic checks of the estimated residuals. Model selection criteria can be used to evaluate whether the complexity of the model structure is appropriate for the amount and quality of the available data by investigation of model modifications (see again Chapter 4).

Further, the sensitivity of the results can be investigated with respect to various 'what if'-type of scenarios. In Chapter 3 a method has been described to investigate the sensitivity of the parameters with respect to observations, regressor variables and weighting factors. It is based on the fact that the estimated parameters are defined implicitly as the solution of an optimization problem. Also a general overview is presented there of sensitivity analysis based on first order and higher-order derivatives.

In MAP, routines are available for the sensitivity analysis of both explicit and implicit relations. Furthermore, routines are implemented for the propagation of uncertainty information, for instance from uncertainty in the estimated parameters into uncertainty of variables which depend on the estimated parameter values.

## 7.3 Experience with the ADOL-C library

The procedures discussed in the previous section have been implemented in MAP using the ADOL-C library to compute the required derivatives. The most important function of which derivatives are computed, is the model map that is used to compute the components of the function $f$ in equation (7.1). These components may depend both on a vector of parameters that must be estimated and on a vector of measured variables, the regressor variables. The model map is supplied by the user and has Function prototype 1. The prototype provides a structure by which other routines can refer to the model.

```
ModelMap(adouble* Regs, adouble* Pars, adouble* Resp);
```

Function prototype 1: Model map that computes response variables for given parameters and regressor variables.

All input variables are declared active using ADOL-C datatype `adouble` so that derivatives with respect to both the parameter and the regressor variables may be obtained.

### 7.3.1 A North Sea herring case study

For the performance experiments that will be discussed in the next sections, we used a model map to generate estimates by means of an age-structured fishery model. The associated parameter estimation procedure, of type (7.1), constitutes

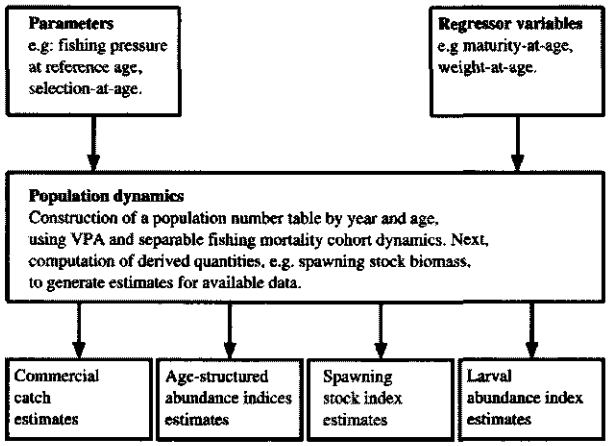



Figure 7.3: Schematic representation of the model map for the North Sea herring case study. The responses consist of estimates for stock observations from multiple information sources.

a re-implementation of the Integrated Catch-Age (ICA) stock assessment procedure (Patterson and Melvin, 1996). This procedure is currently used by the International Council for Exploration of the Sea (ICES) to assess the North Sea herring stock. The goal of the fitting procedure is to reconstruct the history of the herring stock, to investigate its current state and to determine the influence of the commercial fishery. The results of the analysis are used to advise the European Commission on future fishing levels.

The data used to fit the model is obtained both from the commercial fishery and from several research vessel surveys. An example of the latter is an acoustic abundance index that is collected in cooperation by the various fishery institutes of the North Sea countries. Furthermore, data is used from surveys aimed at the measurement of larval stock size and spawning stock biomass.

A schematic representation of the model map is given in Figure 7.3. The parameters and regressor variables are used to generate a population numbers table over a certain time period and for a certain number of age classes. This table is generated using two methods: by means of virtual population analysis and separable fishing mortality cohort dynamics. Both methods are described in Patterson and Melvin (1996). In each case the population is divided into cohorts, i.e. into groups of fish that were born in the same year. The cohort size dynamics is then determined by natural and fishing mortality. A number of parameters quantify the fishing pressure by parameterizing the fishing mortality. For a general overview of age-structured stock assessment methods of this type, see Megrey (1989).

The model contains 45 parameters and has 218 response variables as output. For a detailed description of the model and the results of the uncertainty analysis,

see Chapter 5. The maximum likelihood optimization was performed by means of a Polak-Ribiere conjugate gradient procedure (see for instance Gill et al. (1981)) on a Pentium II machine (350 MHz, 64 Mb).

### 7.3.2 Computation of the objective function

For a number of design reasons it is preferable to have just a single model map function, represented by Function prototype 1. Since all variables for which derivatives are potentially required are active, all derivatives of the function can be computed by defining those variables as independent. Moreover, using this prototype, only one function has to be modified if the user wants to change the underlying model.

This means, however, that for evaluation of the model map, we will have to use a function that is less efficient than a function without overloaded operations, i.e. a function using variables of type `double` instead of `adouble`. ADOL-C offers the possibility to evaluate a function of which an execution trace has been recorded by means of a call to `function()`. Ideally, the overhead in evaluating a function in this manner is small.

The most critical function evaluation with respect to run-time efficiency is that of the objective function, since it is required so often in the iterative optimization procedure. To investigate the overhead of using the `adouble` type we compared the performance of a conjugate gradient procedure for two evaluation methods. First by means of the `function()` call and then by an objective function based on the original model map, i.e. a function that does not use any ADOL-C specific constructs.

For the fitting of the fisheries model the overhead turns out to be quite small, i.e. the conjugate gradient procedure using the function with variables of type `double` is faster only by a factor of 1.3.

### 7.3.3 Computation of the objective function derivative

One of the most important contributions of AD to regression analysis is the efficient computation of the objective function derivative. This derivative depends in turn on the model map derivative, according to

$$\frac{dS}{d\theta} = 2\,(f(\theta) - y)^T \frac{df}{d\theta}. \tag{7.3}$$

It is possible to compute the objective function derivative in two ways: (i) by tracing the entire objective function computation using the parameters as independent variables and the objective value as dependent variable, and by then using a `gradient()` call; (ii) by tracing the model map computation using the model map response variables as dependent variables; after a `jacobian()` call, the objective function derivative can then be computed using (7.3). We compared these methods and found that for the fisheries model map the first method turns out to be surprisingly more efficient, viz. approximately 8 times faster.

This can be understood from the fact that for the first method the reverse mode is very efficient since the derivative of a scalar variable is computed. For the second method one has to resort to the less efficient forward mode of differentiation since in this case the derivative with respect to a large number of components, 218 in fact, are required. The fact that the computation of the function of the first method is slightly longer because of the summing and squaring of the model map components, is in this example easily offset by the difference in efficiency of the derivative computation.

Next, we compared the performance of the automatically computed objective function derivative with one computed by means of central finite differences. We found that for the fishery model map the conjugate gradient procedure using the ADOL-C derivatives was approximately 7 times as fast as the procedure using the numerical approximation.

### 7.3.4 Computation of higher-order derivatives

An interface to facilitate the computation of higher-order derivatives was not yet available in the ADOL-C 1.6 version that we used (except for the computation of Hessians), but has now been implemented in newer versions[2].

In order to compute the curvature arrays mentioned in section 7.2.3, first the Hessian of each of the model map response vector components must be computed, i.e. a total of 218 Hessians. Once again, we would like to make use of the function declared by Function prototype 1 to avoid the situation that if model modifications are desired, changes must be carried out in several functions. This prototype is not optimal, however, from the computational point of view for two reasons. The first is that we do not need active regressor variables, which constitutes an overhead comparable to that discussed in section 7.3.2. The second reason is that the derivative is required with respect to only one of the response components.

We tested the performance of using Function prototype 1 with the regressor variables not defined as independent variables, and with only one of the components declared dependent. This was compared to the performance of a special function that only computes the required component and for which the regressor variables are not active at all. For the computation of the 218 Hessian matrices this special function was about twice as fast.

Finally, in many regression problems one is confronted with the situation that derivatives of the model map are needed with respect to the parameters, but each time for a different regressor vector. There are then two alternative evaluation methods: (i) Define both parameters and regressors as independent, then use only that part of the derivatives that corresponds to the parameters. In this case only one execution trace has to be recorded. (ii) Define only the parameters to be

[2]At the time the experiments of this chapter were performed, MAP was linked to the ADOL-C 1.6 library. In the current version the ADOL-C 1.8 library is used, which has more flexible facilities for dealing with higher-order derivatives.

independent, and record a new trace for every regressor vector. Which alternative is more efficient depends on the number of regressor variables and the overhead associated with the recording of the trace. Both options, however, seem to be unnecessarily inefficient.

As a solution to this problem, a new type of variable can be introduced. Just as active variables can be labeled independent, it should then be possible to label active variables as of an auxiliary type. This type then indicates that no derivatives with respect to these variables are required, but that in the computation of the derivatives these variables are allowed to take a different input value. Taking this approach the execution trace would have to be computed only once.

## 7.4 Concluding remarks

The MAP library has benefited substantially from the use of the ADOL-C library. ADOL-C has proved to be stable, and its interface is convenient to use. Automatic differentiation has made the parameter estimation more efficient and has made it feasible to perform higher-order uncertainty analysis on complex models.

It is well-known that the current performance bottleneck for ADOL-C lies in the file access of the execution trace. Improvements using memory mapped I/O have been announced on the ADOL-C 2.0 website.

One of the MAP design restrictions has been the use of a single model map function, i.e. for the actual overloaded model computation code, such that user modifications to the model must be carried out in only one place. This requires, however, that the overhead in function evaluation from execution traces is small. In this respect we have also noted the desired feature to obtain derivatives with respect to only a subset of the active variables while keeping the possibility to change the values of the remaining active input variables without re-tracing the computation.

As an illustration, the North Sea herring stock assessment was investigated. As described in Chapter 5, we obtained more accurate parameter uncertainty information using higher-order model derivatives. Further, by investigation of a number of model modifications, we showed that the precision of the estimates improves for models of smaller complexity. Also several types of analysis based on derivatives obtained by means of AD were used to investigate the sensitivity of variables of interest to the fishery management with respect to various assumptions and input variables.

# Chapter 8

# Summary and conclusions

In this thesis we exploit a combination of automatic differentiation algorithms and derivative-based methods to improve tools for model analysis. Improvements can be attained with respect to a variety of issues, for example the feasibility of application in terms of runtime and memory requirements in relation to problem size, the efficiency of implementation, the convenience of operation in practice, and also, the clarity by which tools are understood.

In the introduction of this thesis, we have formulated four main objectives to contribute to such improvements (in section 1.4); in the following we summarize the main results and conclusions with respect to the objectives.

**1. To present and develop derivative-based methods for sensitivity analysis of modeling outcomes, and to investigate their relative merits with respect to other types of methods, in particular in the context of nonlinear parameter estimation.**

Chapter 3 presents an overview of local sensitivity analysis based on first and higher-order derivative information.

Sensitivity analysis based on global sampling methods is suitable for the study of a wide range scenarios and assumptions. For large models, however, computation of global sensitivity information may become prohibitively expensive. In that case, local sensitivity analysis may provide a feasible alternative to obtain the required sensitivity information. It was found that local sensitivity analysis can generally be treated as a process in two steps:

1. Construct a local approximation for the function relating input to output variables. For both explicit and implicit relations derivatives for a Taylor expansion that can serve as the approximation, can be obtained automatically.

2. Use this approximation for the propagation of the uncertainty of the input

variables to the uncertainty of the output variables. Output uncertainty can be decomposed into contributions of the sources of input uncertainty.

We presented a local method for the propagation of expectation and variance-covariance in particular detail. We discussed its derivation, its validation, and showed that it may be used to obtain a decomposition of variance with respect to the input sources of uncertainty. In the context of nonlinear parameter estimation the method is useful for the investigation of the uncertainty in quantities that depend on estimated parameters. We further introduced precision sensitivity coefficients that provide additional insight into the influence of the input sources of uncertainty on the output precision. It was shown that the components into which the variance is decomposed (based on a linear approximation) have an alternative interpretation as marginal precision measures.

We discussed standard methods for propagation of perturbations, and also proposed a method for the sensitivity analysis of implicit relations. In the context of nonlinear parameter estimation, this method can be used to investigate the sensitivity of estimated parameters with respect to other variables used in the estimation procedure. Examples were given with respect to different types of such variables, such as observation weighting coefficients and regressor variables (in Chapter 5), and observations (in Chapter 6).

The wide applicability of the approach of Chapter 3 was further demonstrated in Chapter 4 where it provides a straightforward derivation for the moments of the maximum likelihood estimator given an assumed probability density function of the observation variables.

**2. To present and develop statistical methods that allow for the computation of accurate inferential information for nonlinear models, and for the assessment of the appropriateness of model structure complexity in relation to quality of data.**

In Chapter 4 we have presented a detailed treatment of methods for statistical inference for nonlinear regression models. We have done so with the aim of providing the statistical background for the development of a system for the analysis of parameter estimation procedures by derivative-based methods.

The methods discussed allow for a thorough investigation of the effects of nonlinearity on statistical inference. The likelihood theory presented in the first part of the chapter provides the various methods with a clear interpretation. This theory also leads to the interpretation of model selection criteria as a rate of exchange between data support and model simplicity, thus providing a tool for the investigation of the appropriateness of model structure in relation to data quality. The Akaike and corrected Akaike criteria are useful in this respect if models are only mildly misspecified. This does not have to pose a serious restriction, since large misspecification can usually be screened out by diagnostic checks, for instance by the investigation of the estimated residuals.

By using higher order derivatives of model functions more accurate inferential

information can be obtained. In Chapter 5 we found that in particular a method proposed by Clarke based on marginal curvature provides a suitable method for the investigation of nonlinearity effects. This method on the one hand provides a warning system against very serious nonlinearity effects, and on the other hand yields corrections to the likelihood intervals obtained by linearization for cases where the effects of nonlinearity are relatively mild. An important characteristic of the corrected likelihood intervals is that they are generally shifted in relation to the intervals based on linearization, resulting in intervals that are not symmetric around the parameter estimate.

Another example is the computation of estimation bias and estimation variance-covariance. We have presented a method based on the theory for local sensitivity analysis to obtain higher-order corrections to linear approximations for both non-linear and misspecified models.

**3. To develop a C++ library of routines for model analysis based on automatic differentiation, and to show the benefits of implementation by means of automatic differentiation.**

Chapter 2 provides a general introduction to the principles underlying automatic differentiation algorithms. The main benefits are developed there in detail: (i) AD algorithms are generally more efficient than their numerical or symbolic counterparts; (ii) AD algorithms lead to exact derivatives; (iii) derivative generation by AD algorithms requires no hand coding thus substantially reducing the risk of introducing bugs in the computer code.

Using the ADOL-C library for automatic differentiation based on operator overloading, we have developed a C++ library, named MAP, for the analysis of models represented by differentiable relationships. Chapter 7 describes the main features and design considerations of this library. The structure of the MAP library is summarized in Figure 8.1.

The user provides C++ code for the model relationships to be analyzed, and in case of an estimation problem also the data used to estimate the parameters. The resulting problem definition, combined with the derivatives obtained through the ADOL-C library, are then used in the various routines for estimation, statistical inference, sensitivity analysis and model selection. The MAP library may be directly linked to user programs, or be employed in combination with a graphical user interface, which can be used to control the various routines and view the resulting output.

In the current implementation the user can utilize the full flexibility and efficiency of the C++ language in defining the model relationships and using the routines. One can, however, also imagine a system in which the model relationships are defined in a more simple and user-friendly environment. This would then require compilation of the relationships from the chosen model language to the object code to be used by the analysis routines. This step could then also be combined with an AD precompiler to provide a way to generate the derivative code of the model relationships without using operator overloading.

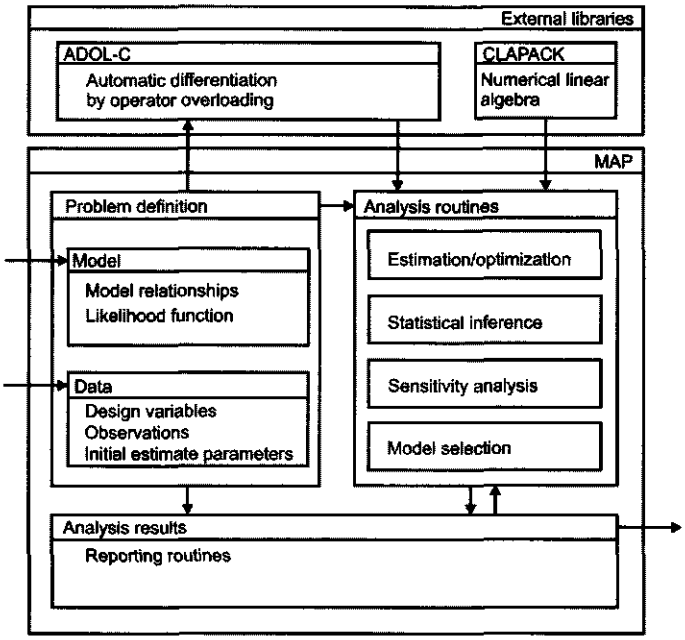



Figure 8.1: Structure of the MAP library.

The experiments presented in Chapter 7 show that the AD routines indeed perform very well in practice and that the ADOL-C library is suitable for models of the complexity encountered in the case studies of this thesis.

Overall we conclude that integration of AD algorithms with statistical packages and numerical libraries is relatively straightforward and is recommended given the increased performance it provides. The design of the MAP library can serve as an example to this end. Implementation should be in a modular manner such that analysis systems can easily be updated once more advanced AD systems become available.

**4. To demonstrate the application of the derivative-based methods and the routines of the library by means of real-life case studies.**

As case studies we considered three problems in ecological assessment:

- Assessment of the contribution of Russian forests to the carbon cycle. In collaboration with the International Institute for Applied Systems Analysis (Chapter 3).

- North Sea herring stock assessment. In collaboration with the Dutch Institute for Fisheries Research (Chapter 5).

- Pacific halibut stock assessment (Chapter 6).

The first study was mainly used as an example to demonstrate the application of the derivative-based methods of sensitivity analysis. A computation is investigated that is used by the Forestry Project at the International Institute for Applied Systems Analysis (IIASA) as part of the assessment of the contribution of Russian forests to the global carbon cycle. For this purpose a database was developed with detailed figures on forest growing stock volumes. For so-called ecoregions the dominant tree species are identified and for each species the database contains the growing stock volumes by age class. Most of the data are obtained from the Russian State Forest Account, which is assembled from forest management enterprises, natural reserves and national parks. The computation first transforms the database of forest growing stock volumes into a number of so-called phytomass fractions, such as wood of stems, wood of branches, and foliage. Next, the carbon content of each of these phytomass fractions is computed and added to obtain the total carbon content.

The studies on fish stock assessment are estimation problems with large numbers of parameters, making them challenging from a model analysis perspective. This is even more so because of difficulties inherent to ecological assessment problems, such as poor replication and general inability to control the processes under study. The model codes and numbers of parameters, particularly of the North Sea herring assessment, are too large for general-purpose statistical packages, but not so large that AD implementation issues regarding the recording of the computations become dominant.

For the North Sea herring study, we have used the routines of the MAP library to re-implement the Integrated Catch-Age Analysis (ICA) assessment procedure currently in use by the International Council for Exploration of the Sea (ICES). The main aim of ICA is the reconstruction of fishery population dynamics by means of commercial catch-at-age data and tuning indices obtained from research vessel surveys.

The study has shown that implementation of an existing complex assessment procedure using the MAP library is feasible. The ICA re-implementation can now be used as a practical assessment tool that can be easily modified if new circumstances or questions require model modifications.

In the final study we have applied adjoint data assimilation to the model underlying Virtual Population Analysis (VPA). VPA is an age-structured method for the estimation of stock sizes and fishing mortality which uses a model that describes the fish population dynamics backward in time. The adjoint VPA procedure was applied to catch-at-age and fishing effort data from the well-documented Pacific halibut fishery and compared to standard VPA.

We conclude with the observation that any model by its nature remains a product of the human imagination; a construction of relationships based on a variety of assumptions. In this thesis we presented tools that on the one hand can use data to quantify the support for such assumptions relative to alternative assumptions and, on the other hand, allow for the accurate computation of the

consequences of given assumptions. In applying models to real world problems, however, the proof of the pudding must be in the eating.

The methods discussed generally require only the computation of the maximum likelihood estimate and first and higher-order derivatives of model functions at this estimate. Since no additional evaluations of the model functions for different values of the parameters or other variables are needed, these methods are suitable for the analysis of large models. If the tools presented here are used while keeping their purpose and limitations clearly in mind they should prove a valuable asset for the extraction of information from data.

# References


Archer, G., A. Saltelli, and I. Sobol (1997). Sensitivity measures, ANOVA-like techniques and the use of bootstrap. *Journal of Statistical computation and simulation 58*, 99–120.

Atherton, R., R. Schainker, and E. Ducot (1975). On the statistical sensitivity analysis of models for chemical kinetics. *AIChE Journal 21*(3), 441–448.

Aubert, P., N. Di Césaré, and O. Pironneau (1999). Automatic differentiation in C++ using expression templates and application to a flow control system. Technical report, Lyon, France.

Barndorff-Nielsen, O. (1983). On a formula for the distribution of the maximum likelihood estimator. *Biometrika 70*, 343–365.

Bartholemew-Biggs, M. C., L. Bartholemew-Biggs, and B. Christianson (1994). Optimization and automatic differentiation in Ada: Some practical experience. *Optimization Methods and Software 4*, 47–73.

Bates, D., D. Hamilton, and D. Watts (1983). Calculation of intrinsic and parameter-effects curvatures for nonlinear regression models. *Commun. Stat. Simul. Comput. B 12*, 71–81.

Bates, D. and D. Watts (1980). Relative curvature measures of nonlinearity. *Journal of the Royal Statistical Society B 42*(1), 1–25.

Baur, W. and V. Strassen (1983). The complexity of partial derivatives. *Theoretical Computer Science 22*, 317–330.

Bendsten, C. and O. Stauning (1996). FADBAD, a flexible C++ package for automatic differentiation using forward and backward methods. Technical Report 1995-x5-94, Department of Mathematical Modelling, Technical University of Denmark, Lyngby, Denmark.

Bernardo, J. and A. Smith (1994). *Bayesian theory*. New York: John Wiley and Sons, Inc.

Berz, M., C. Bischof, G. Corliss, and A. Griewank (Eds.) (1996). *Computational Differentiation: Techniques, Applications, and Tools.* Philadelphia, Penn.: SIAM.

Birnbaum, A. (1962). On the foundations of statistical inference. *Journal of the American Statistical Association 53*, 259–326.

Bischof, C., A. Carle, P. Khademi, and A. Mauer (1996). The ADIFOR 2.0 system for the automatic differentiation of Fortran 77 programs. *IEEE Computational Sci. Engrg. 3.* MCS-P481-1194.

Bischof, C., G. Corliss, L. Green, A. Griewank, K. Haigler, and P. Newman (1993, January). Automatic differentiation of advanced CFD codes for multidisciplinary design. Preprint MCS–P339–1192, Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, Ill.

Box, G. and N. Draper (1987). *Empirical Model-Building and Response Surfaces.* New York: John Wiley and Sons, Inc.

Box, G. E. and G. C. Tiao (1992). *Bayesian inference in statistical analysis.* Wiley Classics Library. New York: John Wiley and Sons, Inc.

Brent, R. (1973). *Algorithms for Minimization without Derivatives.* Englewood Cliffs, N.J.: Prentice Hall.

Burnham, K. P. and D. R. Anderson (1998). *Model selection and inference: a practical information-theoretic approach.* New York: Springer-Verlag.

Campolongo, F., A. Saltelli, T. Sorensen, and S. Tarantola (2000). Hitchhiker's guide to sensitivity analysis. In A. Saltelli, K. Chan, and E. Scott (Eds.), *Sensitivity analysis*, Wiley Series in Probability and Statistics, pp. 15–47. Chichester, England: John Wiley and Sons, Ltd.

Chan, K., M. E. Scott, and T. Andres (2000). Software for sensitivity analysis – a brief review. In A. Saltelli, K. Chan, and E. Scott (Eds.), *Sensitivity analysis*, Wiley Series in Probability and Statistics, pp. 451–465. Chichester, England: John Wiley and Sons, Ltd.

Chan, K., S. Tarantola, A. Saltelli, and I. M. Sobol (2000). Variance-based methods. In A. Saltelli, K. Chan, and E. Scott (Eds.), *Sensitivity analysis*, Wiley Series in Probability and Statistics, pp. 167–197. Chichester, England: John Wiley and Sons, Ltd.

Christianson, D. B. (1991). Reverse accumulation and accurate rounding error estimates for Taylor series coefficients. *Optimization Methods and Software 1*(1), 81–94. Also appeared as Technical Report No. NOC TR239, The Numerical Optimisation Centre, University of Hertfordshire, U.K., July 1991.

Clarke, G. (1980). Moments of the least squares estimators in a nonlinear regression model. *Journal of the Royal Statistical Society B. 42*, 227–237.

Clarke, G. (1987a). Approximate confidence limits for a parameter function in nonlinear regression. *Journal of the American Statistical Association 82*(397), 221–230.

Clarke, G. (1987b). Marginal curvatures and their usefulness in the analysis of nonlinear regression models. *Journal of the American Statistical Association 82*(399), 844–850.

Cook, R. (1986). Assessment of local influence. *Journal of the Royal Statistical Society, Ser. B 48*, 133–169.

Cook, R. D. and S. Weisberg (1990). Confidence curves in nonlinear regression. *Journal of the American Statistical Association 85*(410), 544–551.

Cooke, R. (1991). *Experts in Uncertainty, Opinion and Subjective Probability in Science.* Oxford: Oxford University Press.

Corliss, G. e. a. (Ed.) (2001). *Proceedings of Automatic Differentiation 2000: From Simulation to Optimization (to appear).* Springer-Verlag.

Cover, T. M. and J. A. Thomas (1991). *Elements of information theory.* Wiley Series in Telecommunications. New York: John Wiley and Sons, Inc.

Cox, D. C. (1982). An analytic method for uncertainty analysis of nonlinear output functions, with applications to fault-tree analysis. *IEEE Transactions on Reliability R-31*(5), 465–468.

Davis, P. H., J. D. Pryce, and B. Stephens (1990). Recent developments in automatic differentiation. In J. C. Mason and M. G. Cox (Eds.), *Scientific Software Systems*, pp. 153–165. 11 New Fetter Lane, London EC4P 4EE: Chapman and Hall. Also appeared as Technical Report ACM–89–1, Royal Military College of Science at Shrivenham, Shrivenham, U.K.

Deriso, R., T. Quinn II, and P. Neal (1985). Catch-age analysis with auxiliary information. *Canadian Journal of Fisheries and aquatic sciences 42*, 815–824.

Dieudonné, J. (1960). *Foundations of modern analysis.* New York: Academic Press.

Donaldson, J. and B. Schnabel (1987). Computational experiences for confidence regions and confidence intervals for nonlinear least-squares. *Technometrics 29*, 67–82.

Doubleday, W. (1976). A least squares approach to analyzing catch at age data. *Res. Bull. Int. Comm. NW Atl. Fish. 12*, 69–81.

Draper, N. and H. Smith (1981). *Applied Regression Analysis, second edition.* Wiley Series in Probability and Mathematical Statistics. New York: John Wiley and Sons, Inc.

Eberhard, P. and C. Bischof (1996). Automatic differentiation of numerical integration algorithms. Technical Report ANL/MCS-P621-1196, Argonne National Laboratory, Mathematics and Computer Science Division, 9700 S. Cass Ave., Argonne, IL 60439.

Edwards, A. (1972). *Likelihood.* Cambridge University Press.

Fisher, R. (1956). *Statistical Methods and Scientific Inference.* Edinburgh: Oliver and Boyd.

Fournier, D. Warburton, A. (1989). Evaluating fisheries management models by simulated adaptive control – introducing the composite model. *Canadian Journal of Fisheries and aquatic sciences 46*, 1002–1012.

Fournier, D. and C. Archibald (1982). A general theory for analysing catch at age data. *Canadian Journal of Fisheries and aquatic sciences 39*, 1195–1207.

Fraser, D. (1999). Regression analysis, nonlinear or nonnormal: Simple and accurate p values from likelihood analysis. *Journal of the American Statistical Association 94*(448), 1286–1295.

Fuller, W. (1987). *Measurement error models.* New York: John Wiley and Sons, Inc.

Gelman, A., J. B. Carlin, H. S. Stern, and D. B. Rubin (1995). *Bayesian Data Analysis.* Statistical Science. London: Chapman and Hall.

Ghil, P. and P. Malanotte-Rizzoli (1991). Data assimilation in meteorology and oceanography. *Adv. Geophys. 33*, 141–266.

Giering, R. and T. Kaminski (1992). Recipes for adjoint code construction. Technical report, Max-Planck-Institut für Meteorologie.

Gill, P., W. Murray, and M. Wright (1981). *Practical optimization.* London: Academic Press.

Goudriaan, J. (1992). Biosphere structure, carbon sequestering potential and the atmospheric $^{14}C$ carbon record. *Journal of Experimental Botany 43*(253), 1111–1119.

Grasman, J. and M. Huiskes (2001). Resilience and critical stocksize in a stochastic recruitment model. *Journal of Biological Systems 9*(1), 1–12.

Griewank, A. (1989). On automatic differentiation. In M. Iri and K. Tanabe (Eds.), *Mathematical Programming: Recent Developments and Applications*, pp. 83–108. Dordrecht: Kluwer Academic Publishers.

Griewank, A. (1992). Achieving logarithmic growth of temporal and spatial complexity in reverse automatic differentiation. *Optimization Methods and Software 1*, 35–54.

Griewank, A. (1993). Some bounds on the complexity of gradients, Jacobians, and Hessians. In P. M. Pardalos (Ed.), *Complexity in Nonlinear Optimization*, pp. 128–161. World Scientific Publishers.

Griewank, A. (2000). *Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation.* Frontiers in Applied Mathematics. Philadelphia, Penn.: SIAM.

Griewank, A. and G. F. Corliss (Eds.) (1991). *Automatic Differentiation of Algorithms: Theory, Implementation, and Application.* Philadelphia, Penn.: SIAM.

Griewank, A., D. Juedes, and J. Utke (1996). ADOL–C, a package for the automatic differentiation of algorithms written in C/C++. *ACM Trans. Math. Software 22*(2), 131–167.

Gudmundsson, G. (1986). Statistical considerations in the analysis of catch-at-age observations. *Journal du Conseil International pour l'Exploration de la Mer 43*, 83–90.

Gulland, J. (1965). Estimation of mortality rates. Annex to Arctic Fisheries Working Group Report (meeting in Hamburg) 3, International Council for the Exploration of the Sea, Copenhagen.

Gustafson, P. (2001). On measuring sensitivity to parametric model misspecification. *Journal of the Royal Statistical Society, Ser. B 63*, 81–94.

Hacking, I. (1965). *Logic of statistical inference.* New York: Cambridge University Press.

Hamilton, D. (1986). Confidence regions for parameter subsets in nonlinear regression. *Biometrika 73*, 57–64.

Hamilton, D., D. Watts, and D. Bates (1982). Accounting for intrinsic nonlinearity in nonlinear regression parameter inference regions. *The Annals of Statistics 10*(2), 386–393.

Helton, J. C. and F. J. Davis (2000). Sampling-based methods. In A. Saltelli, K. Chan, and E. Scott (Eds.), *Sensitivity analysis*, Wiley Series in Probability and Statistics, pp. 101–153. Chichester, England: John Wiley and Sons, Ltd.

Helu, S., D. Sampson, and Y. Yin (2000). Application of statistical model selection criteria to the stock synthesis program. *Canadian Journal of Fisheries and aquatic sciences 57*, 1784–1793.

Hill, D. R. and L. C. Rich (1992). Automatic differentiation in MATLAB. *Applied Numerical Mathematics 9*, 33–43.

Hoag, S. and R. McNaughton (1978). Abundance and fishing mortality of Pacific halibut, cohort analysis 1935-1976. Technical report, International Pacific Halibut Commission.

Hoag, S., J. Myhre, G. St-Pierre, and D. McCaughran (1983). The pacific halibut resource and fishery in regulatory area 2, i. management and biology. Scientific Report 67, International Pacific Halibut Commission.

Horwedel, J. E. (1991). GRESS: A preprocessor for sensitivity studies on Fortran programs. In A. Griewank and G. F. Corliss (Eds.), *Automatic Differentiation of Algorithms: Theory, Implementation, and Application*, pp. 243–250. Philadelphia, Penn.: SIAM.

Huiskes, M. (1998). Virtual population analysis with the adjoint method. In F. F. et al (Ed.), *Fishery Stock Assessment Models*, pp. 639–658. University of Alaska, Fairbanks: Alaska Sea Grant College Program Report No. AK-SG-98-01.

Huiskes, M. (2001). Automatic differentiation for modern nonlinear regression. In *Automatic Differentiation: From Simulation to Optimization*, pp. 79–88. New York: Springer.

Hurvich, C. and C. Tsai (1989). Regression and time series model selection in small samples. *Biometrika 76*, 297–307.

Iri, M. (1991). History of automatic differentiation and rounding estimation. In A. Griewank and G. F. Corliss (Eds.), *Automatic Differentiation of Algorithms: Theory, Implementation, and Application*, pp. 1–16. Philadelphia, Penn.: SIAM.

Isukapalli, S., A. Roy, and P. Georgopoulos (2000). Efficient sensitivity/uncertainty analysis using the combined stochastic response surface method and automated differentiation: Application to environmental and biological systems. *Risk Analysis 20*(5), 591–602.

Juedes, D. (1991). A taxonomy of automatic differentiation tools. In A. Griewank and G. F. Corliss (Eds.), *Automatic Differentiation of Algorithms: Theory, Implementation, and Application*, pp. 315–329. Philadelphia, Penn.: SIAM.

Kass, R. and A. Raftery (1995). Bayes factors. *Journal of the American Statistical Association 90*, 773–795.

Kass, R. and L. Wasserman (1996). Formal rules for selecting prior distributions: a review and annotated bibliography. *Journal of the American Statistical Association 91*, 1343–1370.

Kimura, D. (1986). Variability, tuning and simulation for the Doubleday-Deriso catch-at-age model. *Canadian Journal of Fisheries and Aquatic Sciences 46*, 941–949.

Kubota, K. (1991). PADRE2, a FORTRAN precompiler yielding error estimates and second derivatives. In A. Griewank and G. F. Corliss (Eds.), *Automatic Differentiation of Algorithms: Theory, Implementation, and Application*, pp. 251–262. Philadelphia, Penn.: SIAM.

Kulisch, U. (1996). Memorandum ueber computer, arithmetik und numerik. Bericht 01–1996, Institut fuer Angewandte Mathematik, Universitaet Karlsruhe, D-76128 Karlsruhe.

Lakida, P., S. Nilsson, and A. Shvidenko (1997). Forest phytomass and carbon in european russia. *Biomass and Bioenergy 12*(2), 91–99.

Lawson, L., Y. Spitz, E. Hoffman, and R. Long (1995). A data-assimilation technique applied to a predator-prey model. *Bulletin of Mathematical Biology 57*(4), 593–617.

Linhart, H. and W. Zucchini (1986). *Model Selection.* Wiley Series in Probability and Mathematical Statistics. John Wiley and Sons, Inc.

Lorenc, A. (1986). Analysis methods for numerical weather prediction. *Q. J. R. Meteorol. Soc. 112*, 1177–1194.

Ludwig, D. and C. Walters (1989). A robust method for parameter estimation from catch and effort data. *Canadian Journal of Fisheries and Aquatic Sciences 46*, 137–144.

Mace, P. M. (1994). Relationships between common biological reference points used as thresholds and targets of fisheries management strategies. *Canadian Journal of Fisheries and Aquatic Sciences 51*, 110–122.

McKay, M. (1995). Evaluating prediction uncertainty. Technical Report NUREG/CR-6311, US Nuclear Regulatory Commission and Los Alamos National Laboratory.

McKay, M., R. Beckman, and W. Conover (1979). A comparison of three methods of selecting values of input variables in the analysis of output from a computer code. *Technometrics 21*, 239–245.

Megrey, B. (1989). Review and comparison of age-structured stock assessment models from theoretical and applied point of view. In B. Megrey and E. Edward (Eds.), *Mathematical Analysis of Fish Stock Dynamics.* American Fisheries Society.

Meyer, M. and J. Booker (1991). *Eliciting and Analyzing Expert Judgment: A Practical Guide.* New York: Academic Press.

Morris, M. (1991). Factorial sampling plans for preliminary computational experiments. *Technometrics 33*, 161–174.

Mukerjee, R. and N. Reid (1999). On confidence intervals associated with the usual and adjusted likelihoods. *Journal of the Royal Statistical Society, Ser. B 61*, 945–953.

Nelder, J. (1961, March). The fitting of a generalization of the logistic curve. *Biometrics 17*, 89–110.

Nilsson, S., A. Shvidenko, V. Stolbovoi, M. Gluck, M. Jonas, and M. Obersteiner (2000, August). Full carbon account for russia. Interim Report IR – 00 – 021, International Institute for Applied Systems Analysis, A-2361 Laxenburg, Austria.

Patterson, K. and G. Melvin (1996, October). Integrated catch at age analysis. version 1.2. Scottish Fisheries Research Report 58, The Scottish Office; Agriculture, Environment and Fisheries Department, HMSO Edinburgh.

Pope, J. and J. Shepherd (1982). A simple method for consistent interpretation of catch-at-age data. *Journal du Conseil International pour l'Exploration de la Mer 40*, 176–184.

Rabitz, H., M. Kramer, and D. Dacol (1983). Sensitivity analysis in chemical kinetics. *Ann. Rev. Phys. Chem. 34*, 419–461.

Ratkowsky, D. (1983). *Nonlinear regression modeling.* New York: Marcel Dekker.

Reid, N. (2000). Likelihood. *Journal of the American Statistical Association 95*(452), 1335–1340.

Richards, F. (1959, June). A flexible growth function for empirical use. *Journal of Experimental Botany 10*(29), 290–300.

Richards, L. and J. Schnute (1998). Model complexity and catch-age analysis. *Canadian Journal of Fisheries and aquatic sciences 55*, 949–957.

Robins, J. and L. Wasserman (2000). Conditioning, likelihood, and coherence: a review of some foundational concepts. *Journal of the American Statistical Association 95*(452), 1340–1349.

Ronen, Y. (1988). Uncertainty analysis based on sensitivity analysis. In Y. Ronen (Ed.), *Uncertainty analysis*, pp. 41–70. Boca Ration, Florida: CRC Press, Inc.

Ross, G. (1970). The efficient use of function minimization in non-linear maximum-likelihood estimation. *Applied Statistics 19*, 205–221.

Ross, G. (1978). Exact and approximate confidence regions for functions of parameters in nonlinear models. In Corstein and J. Hermans (Eds.), *Compstat 78, Third symposium on computation.* Vienna: Physica-Verlag.

Rostaing, N., S. Dalmas, and A. Galligo (1993). Automatic differentiation in Odyssée. *Tellus 45A*, 558–568.

Royall, R. (1997). *Statistical evidence: a likelihood paradigm.* Monographs on statistics and probability. London: Chapman and Hall.

Royall, R. (2000). On the probability of obserserving misleading statistical evidence. *Journal of the American Statistical Association 95*(451), 760–780.

Rudin, W. (1976). *Principles of mathematical analysis.* Singapore: McGraw-Hill Book Company.

Saarinen, S., R. Bramley, and G. Cybenko (1991). Neural networks, backpropagation, and automatic differentiation. In A. Griewank and G. F. Corliss (Eds.), *Automatic Differentiation of Algorithms: Theory, Implementation, and Application*, pp. 31–42. Philadelphia, Penn.: SIAM.

Schimel, D. (1995). Terrestrial ecosystems and the carbon cycle. *Global Change Biology 1*, 77–91.

Schnute, J. (1994). A general framework for developing sequential fisheries models. *Canadian Journal of Fisheries and aquatic sciences 51*, 1676–1688.

Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics 6*(2), 461–464.

Seber, G. and C. Wild (1989). *Nonlinear regresssion.* Wiley Series in Probability and Mathematical Statistics. New York: John Wiley and Sons, Inc.

Shepashenko, D., A. Shvidenko, and S. Nilsson (1998). Phytomass (live biomass) and carbon of siberian forests. *Biomass and Bioenergy 14*(1), 21–31.

Shvidenko, A. and S. Nilsson (1998). Phytomass, increment, mortality and carbon budget of russian forests. Interim Report IR–98–105, International Institute for Applied Systems Analysis, A-2361 Laxenburg, Austria.

Soofi, E. S. (2000). Principical information theoretic approaches. *Journal of the American Statistical Association 95*(452), 1349–1353.

Stefanski, L. (2000). Measurement error models. *Journal of the American Statistical Association 95*(452), 1353–1358.

Stone, M. (1974). Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society, Ser. B 36*, 111–147.

Stone, M. (1977). An asymptotic equivalence of choice of model by cross-validation and akaike's criterion. *Journal of the Royal Statistical Society, Ser. B 39*, 44–47.

Thacker, W. C. (1991). Automatic differentiation from an oceanographer's perspective. In A. Griewank and G. F. Corliss (Eds.), *Automatic Differentiation of Algorithms: Theory, Implementation, and Application*, pp. 191–201. Philadelphia, Penn.: SIAM.

Tsai, C.-L. (1988). Power transformations in nonlinear regression models. *Technometrics 30*(4), 441–448.

Welsh, A. (1996). *Aspects of statistical inference.* Wiley Series in Probability and Statistics. New York: John Wiley and Sons, Inc.

White, H. (1982). Maximum likelihood estimation of misspecified models. *Econometrica 50*(1), 1–25.

White, H. (1994). *Estimation, inference and specification analysis.* Cambridge University Press.

# Samenvatting

In diverse takken van de wetenschap worden grote wiskundige modellen gebruikt om complexe systemen te analyseren. Men kan hierbij denken aan modellen voor weersvoorspelling in de meteorologie, aan macro-economische modellen in de econometrie of aan transportmodellen voor luchtvervuiling in de milieuwetenschappen.

Modellen worden opgesteld voor allerlei doeleinden: het voorspellen of reconstrueren van het gedrag van een proces, het identificeren van mechanismen die in grote lijn het gedrag van een proces bepalen, het nemen van beslissingen, het doorrekenen van scenario's, of het verkennen van wetenschappelijke ideeën met betrekking tot de veronderstelde aard van een proces.

Met modellen probeert men bepaalde aspecten van de werkelijkheid in wiskundige termen te beschrijven. Data worden daarbij gebruikt om de aannemelijkheid van de alternatieve beschrijvingen te onderzoeken. Vaak neemt dit de vorm aan van het fitten van een model: een aantal parameters in een model wordt onbepaald gelaten, en observaties van procesvariabelen worden gebruikt om die parameters te kalibreren. De parameterwaarden die corresponderen met het alternatief dat het best overeenstemt met de data, leiden tot de uiteindelijke modeluitkomsten. Zulke modeluitkomsten dan als 'waar' accepteren, zonder verdere analyse van de onzekerheid van het resultaat zou zeer onwenselijk zijn. Liever zou men bijvoorbeeld in staat willen zijn om de onzekerheid in de geschatte parameters te kwantificeren, evenals de gevoeligheid van de uitkomsten met betrekking tot de veronderstellingen die aan het model ten grondslag liggen.

Met behulp van statistische modellen kunnen niet alleen systematische effecten in een proces worden beschreven, maar kan ook rekening worden gehouden met onverklaarde variatie en onzekerheid. Het fitten van een model wordt dan, in statistische terminologie, een parameterschattingsprobleem. Daarbij kan de inductieve statistiek worden gebruikt om de ondersteuning van modelveronderstellingen ten opzichte van alternatieve veronderstellingen te kwantificeren. Met behulp van gevoeligheidsanalyse kan bovendien de onzekerheid met betrekking tot modelveronderstellingen worden gerelateerd aan de onzekerheid in de modeluitkomsten.

De uitvoerbaarheid van de diverse vormen van analyse hangt af van de aard van de beschouwde modellen. Om redenen die hieronder duidelijk zullen worden, beperken we ons in dit proefschrift tot die gevallen waarbij de systematische effecten worden gemodelleerd door middel van differentieerbare functies.

Voor dergelijke modellen is de theorie voor inductieve statistiek en gevoeligheidsanalyse meestal beschikbaar, met dien verstande dat niet-lineariteit van een model vaak speciale aandacht vergt. In de praktijk van het modelleren van complexe modellen is het echter zo dat de modelfuncties kunnen bestaan uit vele duizenden regels computercode; bovendien moeten soms honderden parameters worden geschat. Voor dit soort grote modellen kan het benodigde aantal berekeningen onuitvoerbaar blijken te zijn. Dit probleem heeft geleid tot een belangrijke richtpunt voor het proefschrift, namelijk om inductieve statistiek en gevoeligheidsanalyse breder inzetbaar te maken in de praktijk. De aanpak die we hiertoe kiezen bestaat uit het gebruik van een combinatie van automatische differentiatie algoritmen en methoden voor modelanalyse gebaseerd op afgeleiden. Dit verklaart ook de vertaalde titel van dit proefschrift: 'Automatische Differentiatie Algoritmen in de Analyse van Modellen'.

Dit vereist wel enige uitleg over de rol die afgeleiden spelen in de modelanalyse. Deze rol kan men moeilijk overschatten in het modelleren zelf; denk bijvoorbeeld alleen al aan het gebruik van afgeleiden in de karakterisatie van optima en in differentiaalvergelijkingen bij het modelleren van verandering. In de analyse van modellen kan het belang van eerste en hogere orde afgeleiden worden verklaard aan de hand van hun rol in het construeren van lokale benaderingen van functies. Dergelijke benaderingen worden gebruikt in bijvoorbeeld de optimalisatie, de bifurcatie analyse, en zoals in dit proefschrift wordt aangetoond, ook in de inductieve statistiek en gevoeligheidsanalyse. Methoden gebaseerd op zulke benaderingen kunnen vaak een gunstig alternatief vormen voor de duurdere globale analyses.

Er bestaan diverse manieren om afgeleiden uit te rekenen. Bekende aanpakken zijn numerieke differentiatie, bijvoorbeeld met behulp van eindige differenties, en symbolische differentiatie, het met behulp van de computer differentiëren van wiskundige uitdrukkingen met computeralgebra systemen. Automatische differentiatie algoritmen vormen een nieuwe klasse van algoritmen voor het berekenen van afgeleiden, speciaal gericht op functies die beschikbaar zijn in de vorm van computercode. Implementaties van AD algoritmen gebruiken de computer code van een functie als input om als output code voor de afgeleiden te genereren. De algoritmen hebben een aantal gunstige eigenschappen die later aan de orde zullen komen.

Een belangrijk onderdeel van de studie bestaat uit het ontwikkelen van software om het potentieel van routines gebaseerd op automatische differentiatie aan te tonen. Dit heeft geleid tot een C++ bibliotheek, MAP genaamd, voor de analyse van niet-lineaire modellen die gerepresenteerd kunnen worden door middel van (voldoende) differentieerbare functies. In deze bibliotheek zijn de in het proefschrift behandelde methoden voor inductieve statistiek, modelselectie en gevoeligheidsanalyse geïmplementeerd.

De doelstellingen van het proefschrift kunnen nu als volgt worden samengevat:

1. Het presenteren en ontwikkelen van op afgeleiden gebaseerde methoden voor gevoeligheidsanalyse van modeluitkomsten, en het onderzoeken van hun ver-

diensten in vergelijking tot andere methoden, met name in de context van niet-lineaire parameterschattingsproblemen.

2. Het presenteren en ontwikkelen van nauwkeurige methoden voor de analyse van niet-lineaire statistische modellen, en het geven van een overzicht van methoden voor het beoordelen van de geschiktheid van de complexiteit van de modelstructuur in relatie tot de kwaliteit van data.

3. Het ontwikkelen van een C++ bibliotheek met routines voor modelanalyse gebaseerd op automatische differentiatie, en het aantonen van de voordelen van implementatie door middel van automatische differentiatie.

4. Het demonstreren van de toepassing van de softwarebibliotheek en de op afgeleiden gebaseerde methoden voor modelanalyse door middel van case-studies.

Deze doelstellingen zijn uitgewerkt in dit proefschrift. We geven van elk hoofdstuk een korte samenvatting.

**Hoofdstuk 1** geeft een algemene introductie tot de doelstellingen en de aanpak van het onderzoek.

**Hoofdstuk 2** behandelt de principes die aan de automatische differentiatie (AD) algoritmen ten grondslag liggen. Hierbij worden verschillende implementatietypen onder de loep genomen. Verder worden de belangrijkste redenen om AD algoritmen te gebruiken uitgewerkt: (i) AD algoritmen zijn over het algemeen efficiënter dan numerieke methoden; (ii) AD algoritmen leiden tot code voor de exacte afgeleiden, zodat geen benaderingen nodig zijn; (iii) voor het verkrijgen van de afgeleiden hoeft niet met de hand te worden geprogrammeerd zodat de kans op programmeerfouten kleiner is.

**Hoofdstuk 3** geeft een overzicht van lokale methoden voor gevoeligheidsheidsanalyse gebaseerd op eerste en hogere orde afgeleiden. De methoden kunnen worden gekarakteriseerd als een proces met de volgende stappen:

1. Construeer een lokale benadering van de functie die de inputvariabelen aan de outputvariabelen relateert. Voor zowel expliciete als impliciete relaties is het mogelijk om automatisch de afgeleiden voor een Taylorontwikkeling te genereren die als benadering kan dienen.

2. Gebruik deze benadering om de onzekerheid in de inputvariabelen te transformeren naar de onzekerheid in de outputvariabelen. De onzekerheid in de output kan hierbij worden ontbonden in bijdragen van de verschillende bronnen van inputonzekerheid.

Voor beide stappen worden technieken gepresenteerd en vergeleken met globale vormen van gevoeligheidsanalyse. De methoden zijn toegepast in een studie naar de bijdrage van de levende biomassa van bossen aan de koolstofkringloop.

**Hoofdstuk 4** behandelt methoden voor niet-lineaire parameterschattingsproblemen. Twee onderwerpen staan hierbij centraal: modelselectie en inductieve statistiek. Beide onderwerpen worden geïntroduceerd aan de hand van de zogenaamde aannemelijkheidstheorie. Methoden voor modelselectie kunnen een bijdrage leveren aan de keuze van een modelstructuur die geschikt is voor de kwaliteit van de beschikbare data.

We laten zien dat met behulp van hogere orde afgeleiden van de modelfuncties meer nauwkeurige statistische informatie kan worden verkregen. De onzuiverheid en variantie van meest aannemelijke schatters kunnen bijvoorbeeld nauwkeuriger worden bepaald met de methoden van hoofdstuk 3. Verder wordt ingegaan op het meten van niet-lineariteit en de invloed van niet-lineariteit op verschillende vormen van inductieve statistiek.

In **hoofdstuk 5** is een schattingsprocedure geanalyseerd voor het reconstrueren van de ontwikkeling van de toestand van beviste populaties. De procedure maakt deel uit van het onderzoek van de Internationale Raad voor het onderzoek van de zee (ICES) om te komen tot adviezen voor een veilig niveau van bevissing voor onder meer het haringbestand. We hebben onderzoek gedaan naar de invloed van niet-lineariteit op de kwaliteit van de parameterschattingen en naar de gevoeligheid van de parameterschattingen met betrekking tot wegingsfactoren van verschillende observaties en natuurlijke sterfte. Daarnaast hebben we de onzekerheid van de paaibiomassa beschreven in termen van de onzekerheid in de geschatte parameters en een aantal vereenvoudigingen van het model voorgesteld. Met de studie is aangetoond dat het mogelijk is om met de MAP-bibliotheek grote ecologische assessment procedures te implementeren.

**Hoofdstuk 6** beschrijft een studie naar de toepassing van een zogenaamde adjoint data-assimilatie methode op het model dat ten grondslag ligt aan Virtuele Populatie Analyse. Deze methode kan worden gebruikt voor het beoordelen van de kwaliteit van visbestanden. We beschrijven hoe de adjoint methode kan worden geïmplementeerd en laten zien dat de methode in discrete vorm equivalent is met achterwaartse automatische differentiatie. De methode is toegepast op data van de goed gedocumenteerde heilbotvisserij voor de westkust van de Verenigde Staten en Canada.

**Hoofdstuk 7** beschrijft het ontwerp en het gebruik van de softwarebibliotheek. Ook worden experimenten voor het meten van de prestaties van de routines beschreven aan de hand van enkele testen met de parameterschattingsprocedure uit hoofdstuk 5.

**Hoofdstuk 8** geeft de belangrijkste conclusies.

# Curriculum Vitae

De auteur van dit proefschrift werd geboren op 17 april 1973 te Almelo. Zijn VWO opleiding aan het Christelijk Lyceum Almelo rondde hij af met een Gymnasium B diploma in 1991.

In datzelfde jaar begon hij zijn studie Technische Wiskunde aan de Technische Universiteit Delft. Hij studeerde af bij de vakgroep Toegepaste Analyse op het gebied van de kwalitatieve analyse van niet-lineaire systemen en in september 1996 behaalde hij zijn ingenieursdiploma (met lof).

Vanaf januari 1997 heeft hij als assistent in opleiding bij de leerstoelgroep Mathematische en Statistische Methoden van de Wageningen Universiteit aan zijn promotie-onderzoek gewerkt. Dit heeft geleid tot het voor u liggende proefschrift 'Automatic Differentiation Algorithms in Model Analysis'.

Sinds augustus 2001 werkt hij als postdoc bij de groep Systemen en Signalen van het Centrum voor Wiskunde en Informatica te Amsterdam.