153

MOVE

MOdel for terrestrial VEgetation, Version 4.0

M.H.C. van Adrichem
F.G. Wortelboer
G.W.W. Wamelink

werkdocumenten

WOt
Wettelijke Onderzoekstaken Natuur & Milieu

WAGENINGEN UR
For quality of life

Planbureau voor de Leefomgeving

MOVE

The 'Working Documents' series presents interim results of research commissioned by the Statutory Research Tasks Unit for Nature & the Environment (WOT Natuur & Milieu) from various external agencies. The series is intended as an internal channel of communication and is not being distributed outside the WOT Unit. The content of this document is mainly intended as a reference for other researchers engaged in projects commissioned by the Unit. As soon as final research results become available, these are published through other channels. The present series includes documents reporting research findings as well as documents relating to research management issues.

This document was produced in accordance with the Quality Manual of the Statutory Research Tasks Unit for Nature & the Environment (WOT Natuur & Milieu) and has been accepted by Paul Hinssen, programme coordinator of WOT Natuur & Milieu.

WOt Working Document **153** presents the findings of a research project commissioned by the Netherlands Environmental Assessment Agency (PBL) and funded by the Dutch Ministry of Economic Affairs, Agriculture & Innovation (EL&I). This document contributes to the body of knowledge which will be incorporated in more policy-oriented publications such as the Nature Balance and Environmental Balance reports, and Thematic Assessments.

MOVE

MOdel for terrestrial VEgetation
version 4.0

M.H.C. van Adrichem
F.G. Wortelboer
G.W.W. Wamelink

Werkdocument 153
Wettelijke Onderzoekstaken Natuur & Milieu
Wageningen, december 2010

Abstract

Adrichem, M.H.C. van, F.G. Wortelboer and G.W.W. Wamelink, 2010. *MOVE; MOdel for terrestrial VEgetation, version 4.0.* Wageningen, Statutory Research Tasks Unit for Nature & the Environment, WOt-rapport 153. 182 p.; 46 fig.; 8 Tables.; 49 ref.; 7 Annexes

The model MOVE4 predicts the probability of occurrence for over 900 plant species based on the Ellenberg indicator values for acidity, moisture, nutrient availability and salinity, the geophysical region and the vegetation type. The model was developed as a follow up of MOVE3.2. Although extensively tested and applied, the principles and the tests of the model were never laid down. This report fills this gap and contains the principles of MOVE4 and the internal tests that were carried out. The tests show that the model gives reasonable results predicting the probability of occurrence of plant species, but that there is room for improvement.

Keywords: Ellenberg indicator values, model, MOVE, plant species, probability of occurrence, vegetation

Alterra Wageningen UR
Postbus 47, 6700 AA Wageningen
Tel: (0317) 48 07 00; fax: (0317) 41 90 00; e-mail: info.alterra@wur.nl

Netherlands Environmental Assessment Agency (PBL)
Postbus 303, 3720 AH Bilthoven
Tel: (030) 274 274 5; fax: (030) 274 44 79; e-mail: info@pbl.nl

---

The Working Documents series is published by the Statutory Research Tasks Unit for Nature & the Environment (WOT Natuur & Milieu), part of Wageningen UR. This document is available from the secretary's office, and can be downloaded from www.wotnatuurenmilieu.wur.nl.

Statutory Research Tasks Unit for Nature & the Environment, P.O. Box 47, NL-6700 AA Wageningen, The Netherlands
Phone: +31 317 48 54 71; Fax: +31 317 41 90 00; e-mail: info.wnm@wur.nl;
Internet: www.wotnatuurenmilieu.wur.nl

F-0008 vs. 1.6 [2009]		Project WOT-04-006-104 – 5230016-10		[Werkdocument 153 - december 2010]

Contents

Summary

The model MOVE was first developed in the late eighties of the last century. The here described model version MOVE4 is the latest version. The model was developed as a follow up of MOVE3.2. Although extensively tested and applied, the principles and the tests of MOVE4 were never laid down. This report fills this gap and contains the principles of MOVE4 and the internal tests that were carried out.

The model MOVE describes the relationship between the occurrence of plant species and environmental variables using regression models. The regression models are calibrated on a large dataset of standardized vegetation samples (relevés). For each relevé the Ellenberg indicator values for acidity, moisture content, salinity and nutrient availability were calculated. Regression equations were fitted per plant species, by appointing the average Ellenberg indicator values to each species present in the relevé and by adding information about the geographical region and vegetation type. Thus for each species a specific regression equation is fitted, containing one or more of the environmental variables as independent variables. The model MOVE4 contains equations for over 900 plant species, which can be used to predict the chance of occurrence per plant species. The percentage deviance explained of the species regression models in the test ranges between 4.5% and 83.0%

The model performance was tested with several statistical techniques. In total we concluded that the model performance was adequate, but not more than that. The biggest gains compared to the MOVE3.2 model version are most likely the better statistical foundation and simulation, the exclusion of the effect of heavy metals and the calculation of the model uncertainty. One drawback of the model (and all earlier versions) is that the effect of management is not included in the regression equation. Including this may result in a lower uncertainty. Another drawback is the use of Ellenberg indicator values. The uncertainty for them is quit big and it is recommended to replace the Ellenberg indicator values by indicator values based on measured data.

1 Introduction

1.1 MOVE

The vegetation model MOVE is part of the decision support system 'The Nature planner' (figure 1, van der Hoek and Bakkenes, 2007), which is used to calculate and evaluate the quality of nature on the Dutch national scale (PBL, 2008; MNP, 2006, 2007). In combination with the soil model SMART2 (Kros, 2002), the hydrological model LGM (Pastoors, 1992), the succession model SUMO2 (Wamelink, 2007), The Transfer module (P2E; transfers SMART2 output to Ellenberg indicator values), the plant dispersal model DIMO (Wamelink *et al.*, in prep.) and the biodiversity indication model BIODIV (van der Hoek *et al.*, 2000), MOVE predicts the probability of occurrence of a large number of plant species as a function of environmental variables.

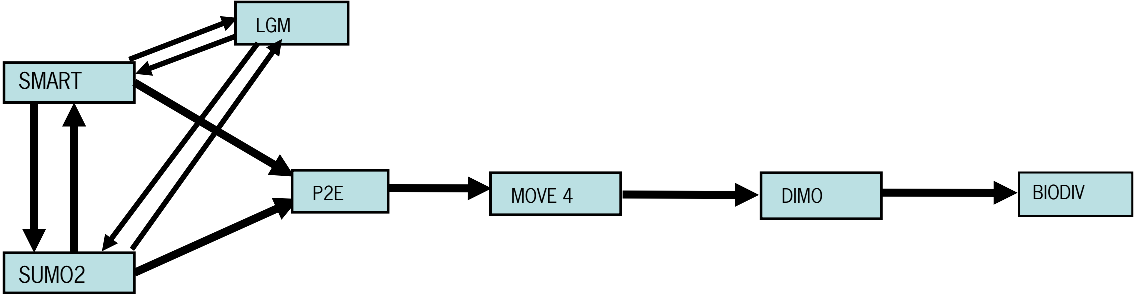


Figure 1. Data exchange between the models of the Nature planner

The model MOVE describes the relationship between the occurrence of plant species and environmental variables using regression models. The regression models are calibrated on a large dataset of standardized vegetation samples (relevés). Based on the species composition of a relevé the Ellenberg indicator values and the vegetation type are determined. Based on the location of the relevé the geophysical region is determined. With multiple logistic regression-analysis (Hosmer & Lemeshow, 1989; McCullagh & Nelder, 1989) a regression model is then fitted for each plant species separately, based on 0 1 data. Thus for each species a specific regression nodel is fitted, containing one or more of the environmental variables as independent variables. MOVE uses these regression models to predict the probability of occurrence for one or many species in a certain area. These calculations are normally performed within the scope of the Nature Planner. Links with other models in this framework will be discussed below. MOVE4 is based on the principles of MOVE3.2, the differences and improvements are described as well.

1.2 Calculations within the Nature planner

Within the Nature Planner, MOVE4 is, as all previous MOVE versions, linked with the soil model SMART2 via three major abiotic parameters describing the soil quality: acidity, moisture and nutrient status. The values are given by SMART2 in physical units (e.g. soil pH). Since MOVE is based on Ellenberg indicator values they have to be translated into Ellenberg indicator values. Although translation functions exist (Alkemade *et al.*, 1996, Ertsen *et al.,* 1998, Wamelink *et al.*, 2002) this is a highly uncertain step in the process (Schouwenberg *et al.*, 2000, Wamelink *et al.*, 2003).

The Groundwater Model for the Netherlands (LGM) is a model for the simulation of quantity and quality aspects of saturated groundwater systems (Pastoors 1992). The model is amongst others used for the simulation of the effects of groundwater abstractions on the geohydrological system.

The model SMART2 (Kros, 2002) simulates processes in the litter and the uppermost mineral layers. It consists of a complete nutrient cycle, including nitrogen and base cation cycles. The simulated nitrogen availability and soil acidity (pH) values are input for the model MOVE4. The spring groundwater table is also input for MOVE4 and derived from a map, it is not influenced by SMART2.

The model SUMO2 (Wamelink, 2007) is integrated in the SMART2 model and simulates the nutrient cycle in the vegetation. It uses factors like nitrogen availability (from SMART2) and light availability to simulate biomass development for five 'functional plant types': grasses and herbs, dwarf shrubs, shrubs and two different tree species. The simulated biomass increment is affected by management (mowing, grazing, sod cutting and forest management at various levels of intensity). Management removes biomass from the system, and hence nitrogen (and in the case of sod-cutting also acid). Both SMART2 and SUMO2 are dynamic process models that produce site-specific output.

P2E is a small module that contains regression equation to transfer the output of SMART2 into Ellenberg indicator values. There are transfer functions for nitrogen availability into Ellenberg N (nutrients), soil pH into Ellenberg R (acidity) and spring groundwater table into Ellenberg F (moisture). From previous research (e.g. Schouwenberg et al. 2000, Wamelink et al. 2003) it is known that this is a very uncertain step in the model chain.

DIMO (Wamelink *et al.*, in prep.) is a plant dispersal model. It simulates plant dispersal in time, given (abiotic) suitability and species-specific characteristics, e.g. different dispersal types, dispersal distances and seed longevity. MOVE4 predicts the probability of occurrence for a species, based on abiotic conditions, after which DIMO refines these maps by calculating the ability of a plant species to actually colonize a suitable spot. Hence, DIMO may set the probability of occurrence for a plant species on a suitable location to zero when this location is not (yet) within reach of currently occupied locations. Colonization can occur either from dispersal or from the seed bank.

At the end of the model chain, BIODIV integrates the data on occurrences of plant species from DIMO, and if desired also occurrences of butterflies (MOVE or LARCH (Verboom *et al.*, 2001)) birds and mammals (LARCH) and compares this with a given reference. This reference is region specific, so site predictions are scaled up to region predictions using the surface areas of the sites. This comparison results in an index on the quality and value of the existing nature in relation to a reference nature.

This report describes the theoretical background of the model MOVE4 and the goodness-of-fit tests of the regression models. How to run the model and several other aspects involving the quality assurance of the model are described in a technical documentation (Wamelink *et al.*, 2009).

This report with the principles of MOVE4 was written by M. H. C. van Adrichem, G.W.W. Wamelink and F.G. Wortelboer, the model MOVE4 was developed and programmed by F.G. Wortelboer. This report was reviewed by Prof. Dr. C.J. ter Braak, Dr. A. van Hinsberg and Dr. P.W. Goedhart. The review is enclosed as appendix 7 including the response of the authors to the reviewers.

2 Material and methods

2.1 Main principles and assumptions behind the MOVE model family

All versions of the MOVE model, including the related GBMOVE version have some principles and assumptions in common. The main principle behind MOVE is that the occurrence of plant species can be estimated on the basis of a very limited set of parameters, all related to soil conditions. In MOVE4 it is assumed that the Ellenberg indicator values for acidity (R), moisture (F), nutrient availability (N) and salinity (S), the vegetation type and geographical region are sufficient to simulate the probability of occurrence of plant species. The model itself consists of a set of response function for the six parameters, including interaction terms per plant species. Only statistical significant model terms were included in the species models, thus simplifying many of the response functions. The Dutch version of the model is calibrated for the Netherlands, and therefore only applicable for the Netherlands. Since the calibration set is retrieved from natural areas, strictly speaking the model is only calibrated for the natural areas, thus further limiting the application range. The calibration set consists of so called vegetation relevés; small vegetation plots mostly ranging from 1*1m (grassland, open sand) till 10*10 m (forest) where species composition and coverage are denoted. For the Netherlands a set of approximately 160,000 relevés is assembled which is assumed by specialist to represent the Dutch flora (Schaminée *et al.*, 1995). This set is also used for the description of the Dutch vegetation.

MOVE predicts the probability of occurrence per species and translates that into absence presence data applying kappa-statistics (see below). The resulting species ‘presences’ gives an indication of the plant species that in principle could be present giving the abiotic conditions; it shows the potential of a plot. Whether or not a species will occur depends also on e.g. plant dispersal capacity or sheer coincidence.

Input for MOVE4 is normally delivered by other models, such as SMART2 (see Figure 1) and existing maps of geographical region and vegetation types. Especially the latter is known to be not constant. This information can be derived from SUMO2 (via SMART2). When the vegetation type is assumed constant, effects of succession are neglected and may lead to wrong conclusions.

2.2 Main differences MOVE3.2 and MOVE4

Bakkenes *et al.* (2002) describe the response models of MOVE 3.2 with eutrophication, acidification and desiccation (as Ellenberg indicator values), vegetation structure, potential impact due to high heavy metals concentrations, salinity and geophysical region as explanatory factors. Version 3.2 was subjected to an external review. Developments in aquatic response models (Ertsen & Wortelboer, 2002) also led to new insights. Both these factors, and questions about the performance of the previous MOVE version, were the reason to develop MOVE4. The new MOVE model is therefore different from its predecessors in several ways.

The variable combined potential affected fraction for heavy metals (*cpaf*), that was added in MOVE3 for the first time, was again omitted from MOVE4, because the reliability of the computations for estimating the potential effects of heavy metals were seriously questioned. Besides, effects due to heavy metals were likely to occur in local heavily polluted sites in the Kempen and Zuid-Limburg areas only, thus limiting its usability for general situations.

In MOVE3.2 the categorical factors vegetation type and geophysical region were presented as factor variables (one variable having different values). These variables were included in all the regression models by default. In MOVE4 these factor variables were split into separate variables (known as dummy variables each having value 1 or 0) and were all subject to addition to or removal from the models within the multiple regression analysis. The main advantage from this change is the increased understanding of the meaning of the variables included in the regression models.

All fitted curves were allowed to have optimum curves by including both linear and quadratic terms. Furthermore, interactions between variables were allowed to enter the model, with the exception of interactions with N, R, F on the one hand and S on the other hand, and the interaction between vegetation type and geophysical region. Interactions were not included in MOVE3.2.

The SPlus-script for fitting the logistic regression model has been changed as well. In MOVE3.2 the standard SPlus methods for multiple regression using the Akaike Information Criterion (AIC) and the Bayesian Information Criterion (BIC) were used, starting from both a null-model (without initial variables included) and a complete model (with all the variables included). The model with the highest percentage of deviance explained was then selected. As the addition of extra variables explains more of the total deviance, generally the more complex models were selected. However, not all of the variables included in the selected regression model added significantly to the deviance explained. In MOVE4 the procedure was adapted so that variables were added to the model when the deviance based test statistic is significant at the 5% level. Also variables that were not significant at 5% are removed from the regression model. Thus in the end all the variables in the regression models are significant. The stepwise selection procedure starts with the null-model. Among the models visited during the stepwise procedure, the BIC criterion was used to select the final model (that we term the best model).
For the continuous variables (Ellenberg's N, R, F and S) both linear and quadratic terms can enter the model, allowing for increasing, decreasing and optimum curves. In MOVE4 interactions between the Ellenberg indicator values are allowed, as well as interactions between vegetation type or geophysical region (FGR) and the Ellenberg indicator values. For example, a different response of one species to Ellenberg N for different vegetation types is allowed for.

New are also the calculation of the so-called Kappa statistic (paragraph 2.5) within the SPlus procedure and the listing of the results in the standard output.

MOVE4 is able to calculate the 95% confidence interval around the predicted probability of occurrence of a species. Furthermore, extra information about the regression model analysis is retained, such as the correlation matrix between the variables within a regression model and the corresponding variance-covariance matrix of the estimates. These can be used in further analyses of sensitivity and uncertainty of the regression models and of model chains in which MOVE4 is included.

All model terms that can enter the logistic regression model during the stepwise procedure are given in paragraph 2.4.

As in earlier versions, MOVE4 regression models are fitted with SPlus. New is the way in which the SPlus output is processed to a standard format within an Access database and the standardization of the response calculations within another Access database. Thus a typical MOVE4 application consists of 4 Access databases:

1. General database for processing the Splus results;
2. Model-specific database with the standardized response model information;
3. General database for calculating the response models (combining 2 with a dataset of values of the environmental variables);
4. Application-specific database with the results of the response models.

The Access databases make extensive use of the built-in programming language Visual Basic for Applications (VBA), generating dynamic queries in the standard query language SQL.

The new implementation of MOVE4 was designed for automated calculation e.g. within the Nature Planner. Moreover, no programming knowledge of VBA or Access experience is needed to perform calculations with MOVE4, as the calculations are totally driven by input files (so-called steering files). See also the MOVE4 User's Guide (Wamelink *et al.*, 2009). (The program listings included in the appendices are given merely for completeness.)

One more difference is that the number of relevés used for calibration and validation was reduced from 109065 in MOVE3.2 to 108826 in MOVE4. 239 Relevés were removed from the dataset because they had a sub-FGR of 0 (i.e. an unknown FGR).

2.3 Vegetation Data

The response models are calibrated on a large dataset of vegetation relevés. This dataset was composed within the framework of the Alterra-project 'Plantengemeenschappen van Nederland' where the vegetation of The Netherlands was described (Schaminée *et al.*, 1995). The dataset is designed to give a balanced overview of the Dutch flora en contains about 169.000 vegetation relevés from the period 1901 to 1997. The relevés are from terrestrial, semi-terrestrial and aquatic environments. The relevé data especially refer to higher plants, but lichens and mosses are also included. The plant species are mostly determined up to the species level. Species that are difficult to determine are sometimes recorded as species group, e.g. the genus *Agrostis.* This study only considers indigenous, higher plant species determined on the species level. The database contains 1599 species that meet these constraints. Of these 1599 species almost 700 species with less than 50 positive observations were removed from the dataset, leaving 914 species. For species that have less than 50 positive observations it is unlikely that a significant model can be fitted. However, this is an arbitrary choice, but our judgement was that the reliability of the possibly significant models is to low, also because data are not randomly collected from the field. A drawback is that species that do have a significant model, despite of the small number of positive observations, are not being calculated. Although for most relevés abundance data is available, these data are not used. Earlier research showed that absence-presence data are much more important than abundance data (van Dobben, 1993).

Relevés that concern heterogeneous environments (banks and other linear elements) have been excluded for this study.

The relevé database was realized by non-random sampling. It is known that there is a relative overrepresentation of relevés on locations with high species diversity or with rare species. These sites have a relatively high botanical value at the expense of more common locations.

Rare species will thus be overrepresented and common species will be underrepresented in the dataset. This non-random sampling can cause displacements in the response curves when using univariate modelling (Runhaar *et al.*, 1994). When response functions are derived based on different explanatory variables at the same time, as in this study, the drawbacks of a non-random dataset may be partly nullified. Because of this multivariate analysis a far-reaching split of separate environments may occur, as a result of which information about different environments is mixed less and the response equations displace less.

It is thus expected and hoped that environments with and without many rare species will be separated based on one or more environmental variables. However this cannot be checked in a quantitative way because there is little information on the non-random nature of the database. For the same reason selection could not be performed to correct for the non-random sampling.

2.4 Environmental data

The majority of relevés in the dataset (Schaminée *et al.*, 1995) only contains plant species records. Simultaneous measurement of abiotic parameters is usually lacking. For the first version of MOVE (Wiertz *et al.*, 1992) the values for environmental variables were therefore deducted by using Ellenberg's indicator values. This was done by calculating the average indicator value per relevé for the total set of plant species present in that relevé. Subsequently, this average was appointed to each individual species in the relevé. For MOVE the vegetation type was determined using species composition, and the geophysical region was appointed using the coordinates of the relevé and the geophysical region map (De Heer *et al.*, 2000). These environmental variables serve as a basis for the responses per plant species for MOVE4.

A criterion for calculating an average Ellenberg indicator value is that there are at least two species with an indicator value per variable per relevé (Runhaar *et al.*, 2002). Indicator variables for relevés that do not meet this requirement are set to missing. In statistics missing values either have to be removed or substituted by other values (e.g. by means of imputation). The risk of removing missing values is that the remaining dataset is biased. Imputation in multivariate analysis can however distort coefficients of association and relationships between variables (Kalton and Kasprzyk, 1982). Moreover, imputed values have an uncertainty of their own, and are therefore usually represented by 5 values sampled from their distribution. Dealing with this uncertainty in the regression analyses makes model estimation much more complicated. Therefore, all relevés with missing values for any of the variables were removed from the database beforehand. This may lead to some bias as it will tend to remove species-poor relevés.

The explanatory variables used in MOVE4 are Ellenberg indicator values for nutrient availability (N, here denoted as *n*), soil moisture (F, here denoted as *f*), acidity (R, here denoted as *r*) and salinity (S, here denoted as *s,* Ellenberg *et al.*, 1991) and furthermore vegetation structure (*veg*) and geophysical region (FGR, in Table 1 denoted as *fgr*). Table 1 gives an overview of all variables. The calculation of the variables will be discussed below.

Table 1. Environmental variables from the dataset.

Variable	Abbreviation	Type	Range
Nutrient	*n*	continuous	1-9
Moisture	*f*	continuous	1-12
Acidity	*r*	continuous	1-9
Salinity	*s*	continuous	0-9
Deciduous forest	*veg*1	binary	
Grassland	*veg*2	binary	
Heathland	*veg*3	binary	
Pine forest	*veg*4	binary	
Spruce forest	*veg*5	binary	
Hills	*fgr*1	binary	
Higher sandy soils	*fgr*2	binary	
River area	*fgr*3	binary	
Fenland	*fgr*4	binary	
Sea clay area	*fgr*5	binary	
Sand dunes	*fgr*6	binary	
Closed sea inlets	*fgr*7	binary	
Tidal zone	*fgr*8	binary	
North Sea	*fgr*9	binary	

The original Ellenberg indicator values for individual species were given as integers (Ellenberg *et al.*, 1991). However the method of assigning Ellenberg indicator values to all relevés, based upon the occurrence of all the species within relevant relevés and their Ellenberg values, leads to real numbers. Thus these indicator values were treated as continuous variables.

Table 2 shows the correlations between the continuous explanatory environmental variables. These correlations are calculated on the reduced dataset. All correlation coefficients are below 0.3, with the exception of a correlation of 0.78 between *n* and *r*. This is a well-known phenomenon in the Dutch flora, where nutrient poor and acid circumstances and nutrient rich and more neutral soil circumstances often coincide, at least under more or less natural circumstances (Dobben and de Vries, 2001). For MOVE4 this has consequences, because this indicates that some combinations of soil characteristics that are theoretically possible are not reflected in the model. So if these circumstances would occur in the future (e.g. due to acid and nitrogen deposition leading to nutrient rich acid circumstances) the model predictions will be less accurate because of extrapolation. Another consequence is that *n* and *r* are exchangeable to some extent, and the statistical stepwise selection procedure might favour one over the other. Thus no definite conclusions can be drawn about the presence or the absence of nutrient availability or acidity in the response models in MOVE.

Table 3 shows some summary statistics for the continuous variables *n*, *f*, *r* and *s* in the dataset (see also Figure 2).

Table 2. Correlations between the continuous environmental variables.

	F	**r**	**s**
n	0.27	0.78	0.09
f	1	0.28	0.05
r		1	0.24
s			1

Table 3. Summary statistics for the continuous variables in the dataset (n=108826).

	n	***f***	***r***	***s***
minimum	1.00	2.00	1.00	0.00
First quartile	4.93	5.42	4.93	0.12
average	5.68	6.90	5.68	0.50
Median	6.11	6.50	6.11	0.28
Third quartile	6.75	8.25	6.75	0.50
maximum	8.67	12.00	8.67	8.67
variance	2.07	4.38	2.07	1.07
Standard deviation	1.44	2.09	1.44	1.03

The indicator value for nutrients has a range of nutrient poor (1) to nutrient rich (9). The moisture indicator value has a scale from 1 to 12: from extremely drought-tolerant to submerged aquatic species. The indicator value for acidity also has a range from 1 (acid) to 9 (basic). The salinity values range from salt intolerant (0) to salt resistant (9), although higher values are very rare.

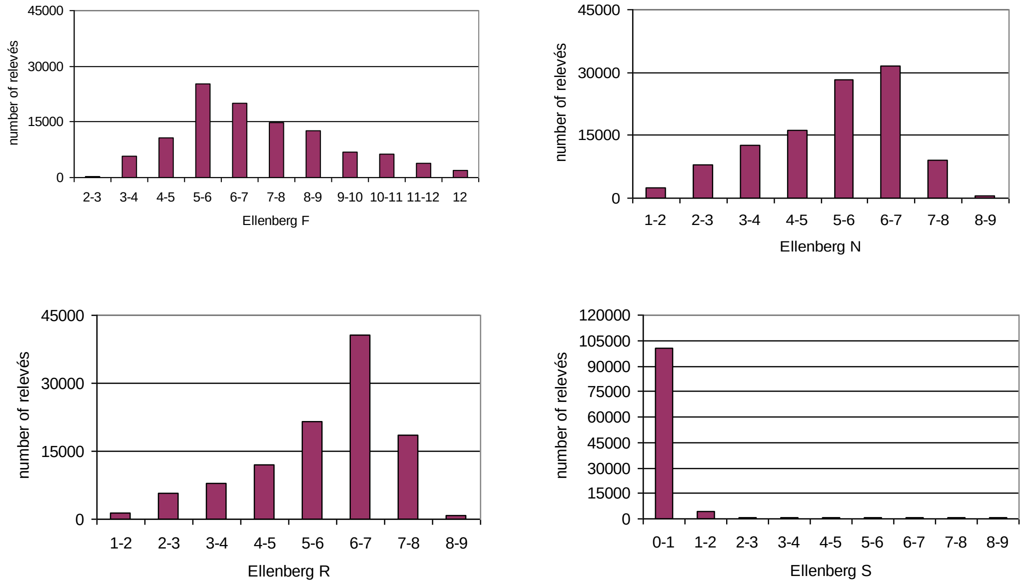


Figure 2. Number of relevés per Ellenberg indicator value (N: nutrients, F: moisture, R: acidity and S: salinity).

The 'vegetation structure' has 5 classes: grassland, heather, deciduous forest, pine forest and spruce forest. These classes correspond to the vegetation classes that are used in the soil model SMART (Kros, 2002). The classes are assigned based on the species that are present in the relevé. Figure 3 shows the distribution of vegetation types over the dataset.

The Netherlands is divided into so called geophysical regions. This classification supplies information about the soil type and the spatial distribution of flora. By incorporating this variable (FGR) into some regression models the reliability of the predicted occurrence probability might increase since it might prevent regression models from predicting plants with a specific geographical distribution on the wrong locations. Different classifications can be used for this purpose, like soil types, flora districts or geophysical regions. The classification with geophysical regions was chosen because this was available as a digital map and because it has relations with flora district as well as soil type.

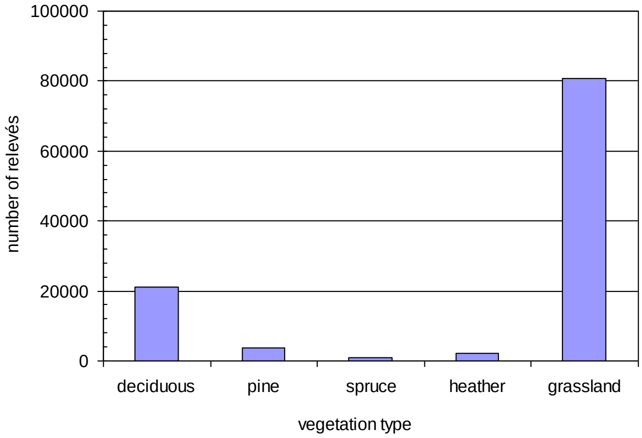


Figure 3. Distribution of vegetation types (per relevé) over the dataset.

The geophysical region classification has 9 categories: hills (only in the South of The Netherlands), higher sandy soils, river area, fenland, sea clay area, sand dunes, closed sea inlets, tidal zone and the North Sea (Figure 4).

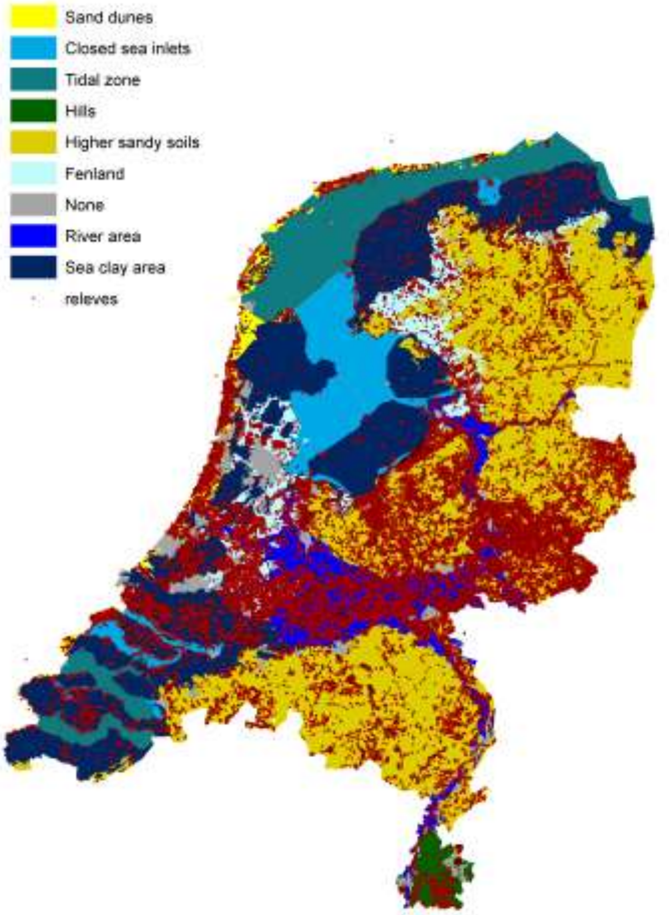


Figure 4. Distribution of relevé locations over the different geophysical regions of The Netherlands.

The distribution of the relevés for each geophysical region was evaluated. Table 4 gives the number of relevés for each FGR that was used for the calibration of MOVE4. A major part of the relevés was recorded in the FGR higher sandy soils, where also a major part of Dutch nature is situated. Table 5 lists the relative area of nature for each FGR in relation to the total area of nature in The Netherlands, and the relative number of relevés in that area. The correlation coefficient between these is 0.80, indicating that the number of relevés is rather equally distributed across the area of nature per FGR. However, when the percentages are compared mutually, it appears that there is a poor relation and that the number of relevés per region deviates from the number that may be expected based on the area of nature in that region. For example, the number of relevés in the river area deviates strongly from what may be expected when the relevés are evenly distributed across the nature areas in the geophysical regions. Only 4.5% of the amount of nature in The Netherlands in situated in the river area, but more than 17% of the total number of relevés is found in this region (Table 5). When the distribution of relevés across the entire country is evaluated, it becomes clear that there is an overrepresentation of relevés on locations with a relative high nature value at the expense of more common locations (Figure 3). This concerns locations with high dynamics, like escarpments (steep slope), seepage areas and coastal areas. This will thus overestimate the probability of occurrence of the species for high biodiversity areas. The same can be said for species from FGRs that are overrepresented in the database.

Table 4. Number of relevés used for the calibration of MOVE4 per geophysical region.

Number of relevés	FGR code	FGR
1655	1	Hills
40625	2	Higher sandy soils
18738	3	River area
13746	4	Fenland
13313	5	Sea clay area
14403	6	Sand dunes
4053	7	Closed sea inlets
1186	8	Tidal zone
1107	9	North Sea

Table 5. Relative area of nature and relative number of relevés per FGR in relation to the total area of nature in The Netherlands.

Geophysical region	% Nature-area	% Relevés
Hills	0.59	1.52
Higher sandy soils	64.34	37.33
River area	4.49	17.22
Fenland	7.97	12.63
Sea clay area	6.3	12.23
Sand dunes	10.87	13.23
Closed sea inlets	2.88	3.72
Tidal zone	2.56	2.11

2.5 Stepwise Logistic Regression

Logistic regression is a type of regression analysis that is appropriate for modelling binary response variables, in this case the presence (1) or absence (0) of plant species in a relevé.
The regression models were derived using a special stepwise procedure, programmed in SPlus (as mentioned in 2.1 and listed in appendix 1, procedures for reading SPlus results,

MOVE4 and VBA-functions in Move 4 can be found in appendix 2, 3 and 4 respectively). Starting from an empty model, only significant terms at the 5% level were added to the model in an addition step and at each step all terms in the model were re-evaluated for their significance and deleted if necessary. There is a large number of explanatory variables (Textbox 1) and thus a large set of candidate models. The Bayesian Information Criterion was used to choose the best model for a species among the models visited by this stepwise procedure. Therefore the explanatory variables in the models differ per plant species. Interaction terms can be selected even when the main effect is not present in the model.

Textbox 1: Model terms used in the stepwise logistic regression analysis employing standard notation for statistical models. veg[1…5] denotes the 5 dummy variables for vegetation and fgr[1…9] represents the 9 geophysical regions (in the text denoted as FGR)

$$(veg[1\ldots5] + fgr[1\ldots9]) * (r + n + f + s + r^2 + n^2 + f^2 + s^2) + r.n + r.f + n.f$$

2.6 Statistical analyses

Goodness-of-fit statistics give an indication of how good a model fits the field situation, i.e. how good the models perform. The model fit is better when there is a higher similarity between model predictions and observations. Because the response is either 0 (absence) or 1 (presence) and the model predictions are in the interval (0,1) assessing the model fit is rather cumbersome. However, the following measures were used to determine the goodness of fit: the percentage deviance explained (formula 1), the Kappa criterion (Cohen, 1960) and the Hosmer-Lemeshow test (Hosmer and Lemeshow, 1989).

Percentage deviance explained:

An often used measure of the goodness of fit of models is the percentage deviance explained, i.e.

$$\%D \text{ explained} = \frac{\text{dev(null)} - \text{dev(model)}}{\text{dev(null)}} \qquad [1]$$

Where dev(model) is the deviance for the fitted model and dev(null) is the deviance for the null model with only a constant in it.

Kappa statistic:

Another way to evaluate the goodness of fit is to use a classification matrix as given in Figure 4. A species that is present in a certain relevé can also be predicted to be present or absent in that relevé and vice versa. These (dis)agreements can be put into a classification matrix for the total set of relevés in order to evaluate the species model. The so-called Kappa statistic (Cohen, 1960; Fielding and Bell, 1997) condenses the classification matrix into a single statistic and represents the proportion of agreement beyond chance ([2]). A kappa statistic of 1 indicates perfect agreement, while a value of 0 indicates no agreement beyond chance. A negative value indicates an agreement less than expected by chance. Note that negative kappa's were not found in this study.

		Actual	
		1	0
Predicted	1	a	b
	0	c	d

Figure 4. Classification matrix.

$$\text{Kappa} = [(a + d) - (((a + c)(a + b) + (b + d)(c + d))/N)] / [N - (((a + c)(a + b) + (b + d)(c + d))/N)] \qquad [2]$$

where a, b, c, and d are the values from the classification matrix and N is the number of relevés.

The abundance of a species in the dataset determines the maximum POO calculated within the given set of relevés. The Kappa method is used to link the calculated POO values to presence or absence of a species. Species that are abundant in the dataset will have high values for the maximum POO calculated for a certain combination of environmental conditions. Likewise, species that are rare will have low values of the maximum calculated POO for any combination of environmental variables. Thus, a calculated POO value of 0.05 may be low for an abundant species (i.e. it never occurs in the relevés with corresponding environmental conditions) whereas 0.05 may be high for a rare species (i.e. it always occurs in the relevés with corresponding environmental conditions). The Kappa statistic is used for finding a critical value of the POO above which a specific plant species is assumed to be present. By applying this value to the model results (in terms of POO) of a species one can assess its presence or absence. Combining the calculated presence and absence data from one relevé, a list of the species present can be made, the number of species can be calculated and even the resulting vegetation association might be assessed. Thus, the critical POO is the unifying concept by which results of multiple species can be combined. In earlier versions of MOVE, results of different species were summed to calculate the number of species present. Because of the huge differences in the levels of calculated POO's between species in situations where they are present, summation of the calculated POO's of different species has no ecological meaning. Now, with this version of MOVE, the presence or absence of each species in a relevé can be assessed first, by comparing the calculated POO with the critical POO, and then the calculated number of species present in that relevé can be derived at.

The method for the calculation of Kappa is adopted from Fielding and Bell (1997). The maximum value of Kappa was found by iteratively increasing the POO (in 100 steps of size (Pmax-PMin)/100 where Pmax and PMin is the maximum and minimum, respectively, of all predicted POO values) and calculating Kappa. The critical POO is the POO at the highest value of Kappa, see Figure 5.

The maximum value of Kappa, derived from the above procedure, can be used as a measure of goodness of fit. When the model agreement is perfect (Kappa = 1) the model predicts the species to be present in exactly the same relevés as it was observed to be present. Table 6 gives two examples from the literature on how Kappa values may be classified. In this report, we follow the classification of Landis and Koch (1977). However it should be kept in mind that the maximum value of the kappa statistic does depend on the relative number of presences in the data, with smaller maximum values for species that are rarer.

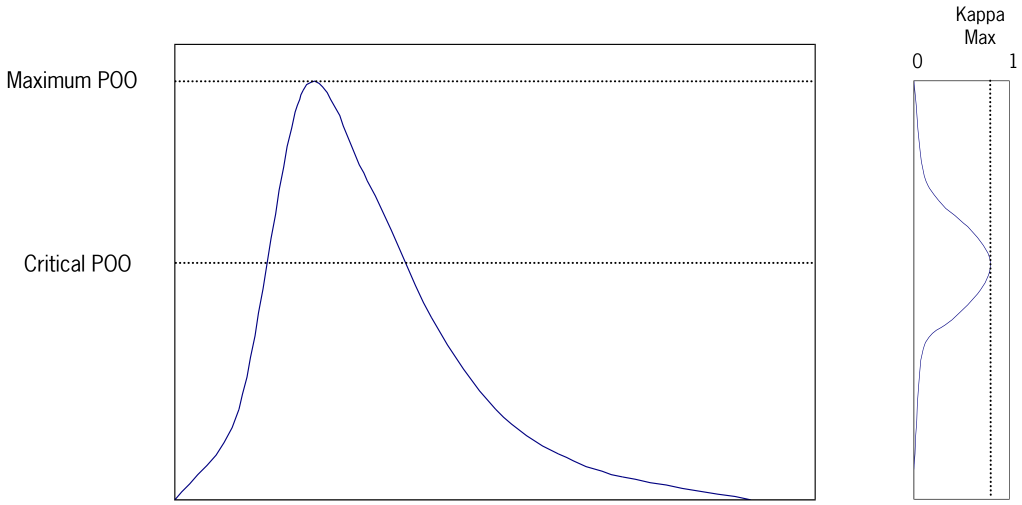


Figure 5. The maximum POO (Pmax) is the POO at the top of the response curve (optimum). The critical POO is the POO at the highest Kappa.

Table 6. Interpretation of agreement for a given kappa value. The classes are arbitrary.

Landis and Koch (1977)			**Svanholm *et al.* (1989)**	
Kappa value	Strength of agreement		Kappa value	Strength of agreement
<0.00	Poor		<0.50	Poor
0.00–0.20	Slight		>0.75	Excellent
0.21–0.40	Fair			
0.41–0.60	Moderate			
0.61–0.80	Substantial			
0.81–1.00	Almost perfect			

Hosmer-Lemeshow test

The Hosmer-Lemeshow test (Hosmer and Lemeshow, 1989) compares groups of observed and predicted responses and then tests the hypothesis that the difference between observed and predicted responses is zero for all the groups. This test is only available for binary response models.

The Hosmer-Lemeshow statistic is calculated similarly to the Pearson chi-square statistic (formula 3):

$$\hat{C} = \sum_{k=1}^{g} \frac{(o_k - n_k \overline{\pi})^2}{n_k \overline{\pi}_k (1 - \overline{\pi}_k)} = \sum_{k=1}^{g} \frac{(o_k - e_k)^2}{e_k (1 - \overline{\pi}_k)} \qquad [3]$$

Where n_k is the number of observations in class k, o_k is the number of occurrences, $\overline{\pi}_k$ the mean predicted probability and e_k the number of estimated occurrences in class k (Hosmer-Lemeshow, 1989). The Ĉ goodness-of-fit statistic sorts observations according to their predicted probability and divides the observations into ten groups of equal size. The Hosmer-Lemeshow statistic is then compared to a chi-square distribution with (10-2) degrees of freedom. A large Ĉ relates to a small p value, implying significant differences between observed and predicted values, and thus lack of fit of the model. A P-value that is greater than 0.05 (test value < 15.51) indicates a good fit. A disadvantage of the Hosmer-Lemeshow tests is that the value of the statistic is sensitive to the choice of the cut-off points that define the groups. In this report the method of Bio (2000) is used.

Model validation

Model validation is inherently difficult for 0/1 data. Nevertheless an effort was made to validate the regression models by first clustering the relevés based on the continuous environmental variables *n*, *f*, *r* and *s*. This resulted in 504 clusters with 10-3027 relevés per cluster. The observed probability of occurrence in each cluster can then be compared with the predicted probability of occurrence. This was done in 4 different ways which are fully described in chapter 4.

2.7 Quality control

This report and the applied methods are extensively reviewed by Dr. P.W. Goedhart and by Prof. Dr C. ter Braak of Biomteris, part of PSG, Wageningen UR and Dr. A. van Hinsberg of PBL, Bilthoven. The review report is added to this report as Appendix 7. This includes a reaction of the authors of this report and whether or not the comment is incorporated in the report. This Document is part of a series of Documents to achieve the A-quality status for the model MOVE4 (for explanation see http://www.wotnatuurenmilieu.wur.nl/NL/Kwaliteit_modellen_en_data/).

3 Results

3.1 Introduction

For all 914 species that were selected from the database it was possible to estimate significant response curves using absence/presence data of 108826 relevés. Considering a total of 1300 – 1500 species occurring in The Netherlands, this allows for a good overview of the calculation of the species composition on any plot in The Netherlands. Rare species are however underrepresented in the database. Although these are often species that are very rare in The Netherlands or even extinct, this may cause some problems, since we are often interested in (protecting) rare species.

In this chapter the results of several tests that MOVE4 was subjected to are presented. This includes several statistical tests on the performance of the regression models. Results for model validation are presented in chapter 4.

Figure 6 shows the distribution across species of the number of terms selected by the stepwise procedure. Most species have less than 20 model terms. So most models are relatively simple; they contain only a fraction of the possible number (137) of model terms. There are just a few species with many model terms (over 40).

The environmental variables for moisture, nutrients, acidity and salinity are included in 896 species models (98%). The 18 models not including any environmental variables are all rare species occurring in at most 129 relevés. However, this is only 10% of the total number of species in this frequency class. The low number of presences may have influenced the lack of resolution in the environmental variables thereby causing the environmental variables not to be selected in the modelling procedure, but this is apparently not true for all rare species.

Of the environmental variables that could be selected (table 7), the salinity variable *s* is the least selected one. This is not very surprising because this variable only has high values in certain parts of the country and in the rest of The Netherlands the values are very low. In contrast with the other environmental factors, salinity, when included in models with a linear term, has a sign of the coefficient value which is almost as often negative as it is positive. For moisture, nutrient and acidity 80-90% of the linear terms have a positive value (i.e. the variables are selected because of a positive relation between presence and high values of the variable). This means that relatively few species have a preference for a very dry, oligotrophic, or acidic environment.

In almost 1/3 of the models no variable for vegetation type (*veg*1 to *veg*5) is included. This indicates that these species are not typical for one specific vegetation type nor that they avoid one specific vegetation type. However, they are not very common species as they all occur in at most 10% of the relevés.

Only 39 species models do not include any geographical region. The low number of models with variable fgr8 (tidal zone) included reflect the relative small number of species that is adapted to these growing conditions.

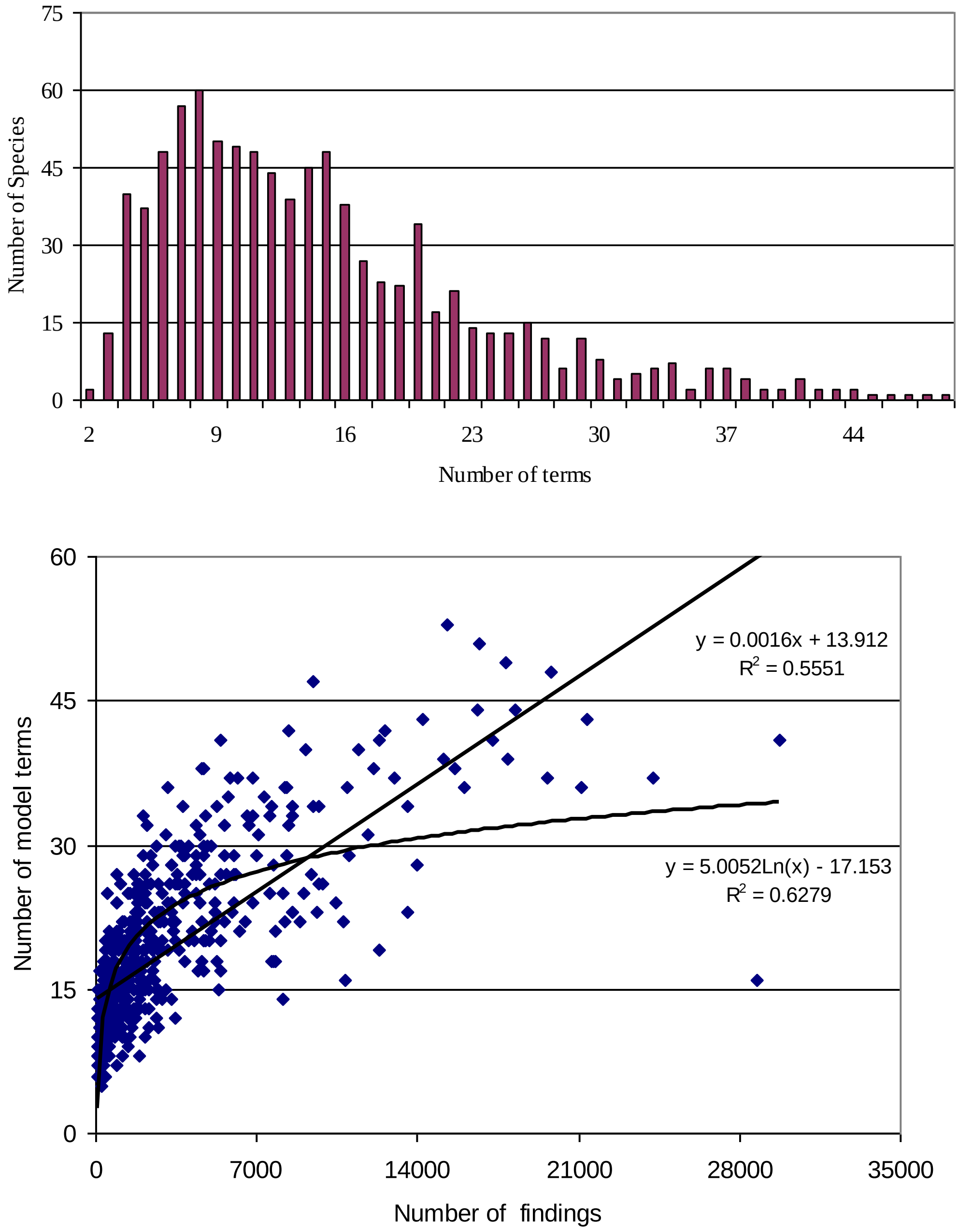


Figure 6. Distribution of number of selected terms in the response models (above, with the median value 13 and the range is 2 – 53) and number of findings per species plotted against the number of model terms for this species (below). See also Appendix 6.

Table 7. Variables in species models of MOVE4 and number of occurrences. Total number of species models: 914 (see also Appendix 6).

Variable	**No of models**	**Linear only**	**Quadratic only**	**Linear + Quadratic**	**Linear Interaction**	**Quadratic Interaction**
elbf-elbs	896					
elbf	812	141	29	642	457	294
elbn	771	69	122	580	415	270
elbr	679	47	128	504	402	247
elbs	550	240	7	303	314	166
veg1-veg5	639					
veg1	251	251			136	
veg2	427	427			229	
veg3	90	90			32	
veg4	65	65			30	
veg5	49	49			15	
fgr1-fgr9	875					
fgr1	278	278			112	
fgr2	443	443			220	
fgr3	433	433			217	
fgr4	267	267			153	
fgr5	319	319			164	
fgr6	413	413			220	
fgr7	236	236			111	
fgr8	90	90			25	
fgr9	181	181			50	

Table 7 shows that quadratic terms of the environmental variables can be included into the models without a linear term present. Most models with environmental variables have both a linear and a quadratic term. The number of models with linear and quadratic terms of the environmental variables sum up to the total number of models with that environmental variable (Table 7). This means that interaction terms (of an environmental variable with another environmental variable, vegetation type or geographical region) is always accompanied by a simple environmental variable model term in the model.

3.2 Statistics

All fitted regression equations are highly significant ($p<<0.001$).This means that in general there is a relationship between the predicted presence and the observed presence of species. However, since there are so many relevés available for the estimation of the regression equations it is more interesting to look at the percentage deviance explained per model. The percentage deviance explained (Figure 7) ranges between 4.6% and 83.0% (median: 39.1%, average: 39.8%), which is not too bad for ecological (survey) data.

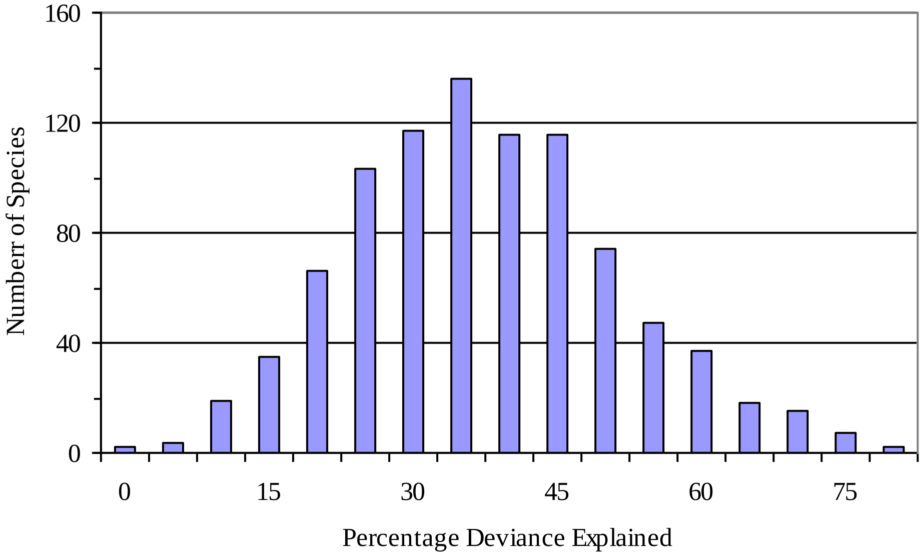


Figure 7. Distribution of percentage of deviance explained by the regression equations. Class boundaries e.g. 5%: >=5 and < 10.

The maximum POO (Pmax) is the POO at the top of the response curve (optimum; Figure 5). Most of the modelled species have a high maximum POO (Figure 8). This means that according to MOVE4, most of the modelled species are quite dominant in the calibration set under certain circumstances. For 30% of the species the maximum POO is higher than 0.80, indicating that according to MOVE4 these species are very dominant under certain circumstances.

Furthermore, Figure 8 shows that there is a relatively high number of species with a very low maximum POO. This means that a relatively high number of modelled species is subdominant in the calibration set under all circumstances.

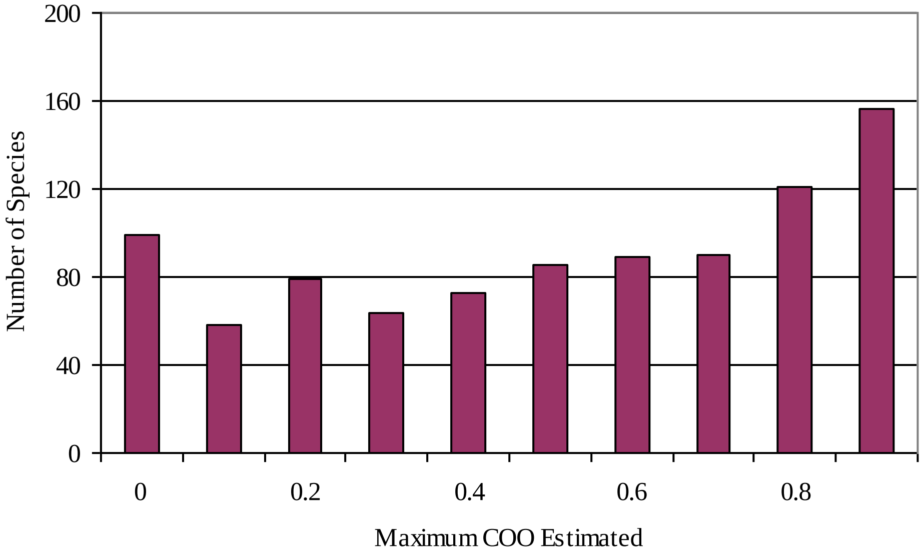


Figure 8. The maximum POO estimated by the regression equations.

The critical POO is the POO at the highest kappa value (Figure 9). This critical POO is rather low for many species, especially compared to the maximum POO estimated (Figure 8). When the predicted POO is higher than the critical POO then the species is assumed to be present.

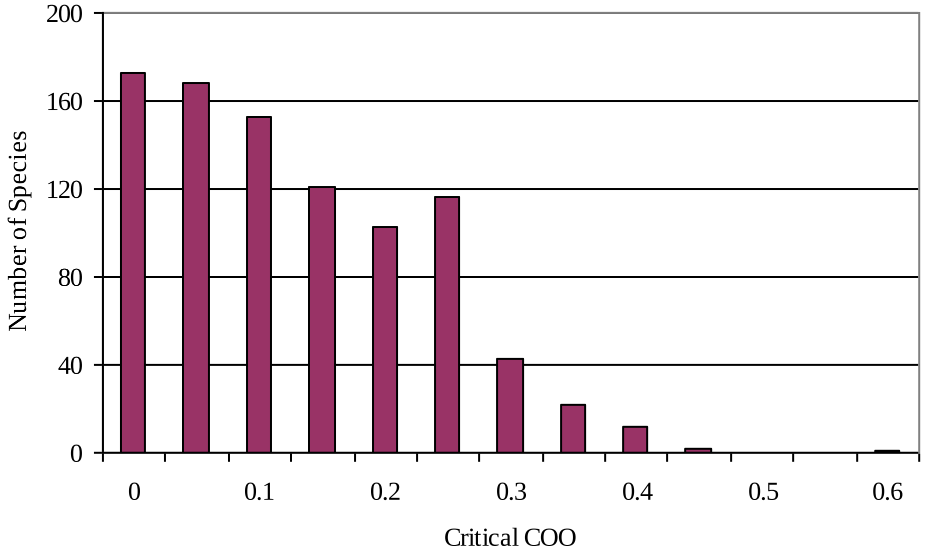


Figure 9. The critical POO plotted against the number of species. The Critical POO is the POO at the highest Kappa value.

Figure 10 shows the distribution of the ratio of the maximum POO and the critical POO. Occasionally this ratio is very large.

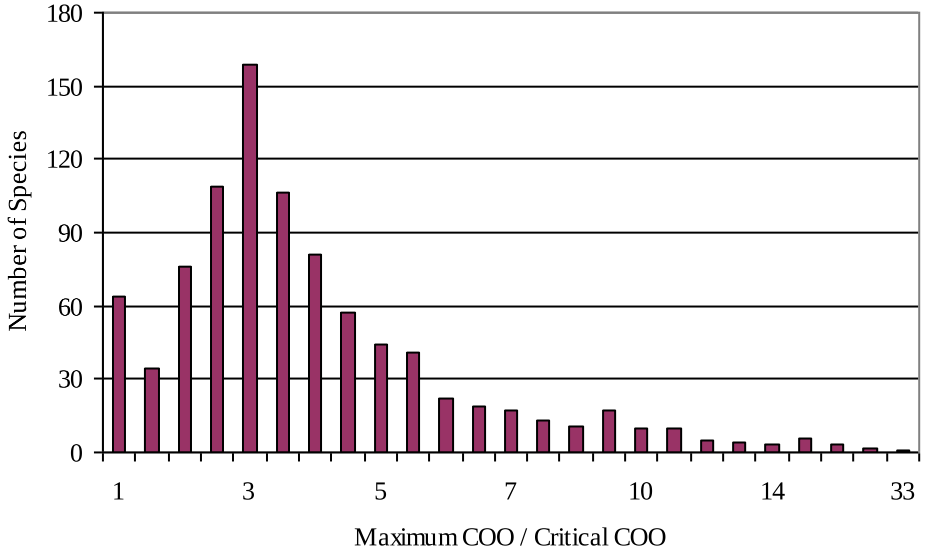
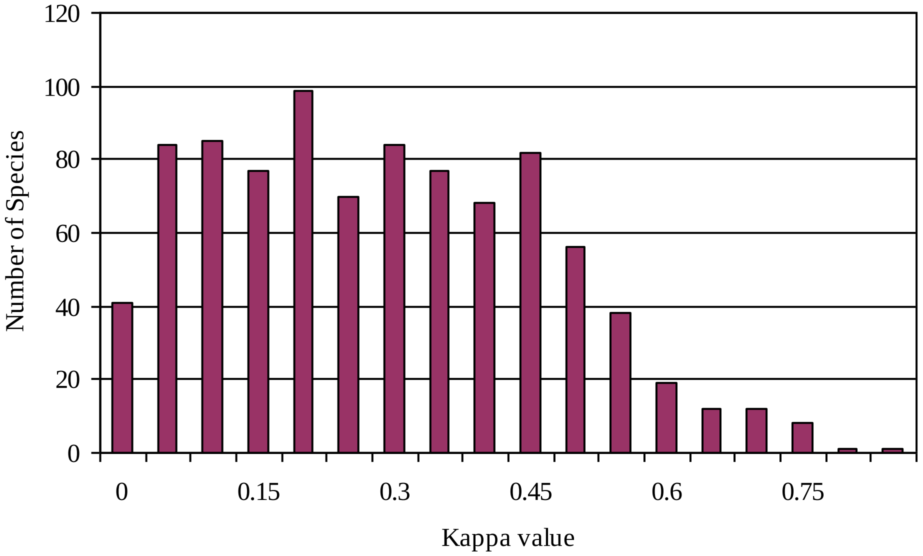


Figure 10. Distribution of the ratio of maximum POO to critical POO of the species. Median value: 3.57.

Figure 11 shows the distribution of the maximal kappa values for the 914 modelled species. Almost all species have a maximal kappa lower than 0.6. Using the classification method for Kappa values of Landis and Koch (1977; table 6) 622 models have a poor to fair agreement, 290 have a moderate to substantial agreement between observed and predicted presence and two models (Potamogeton perfoliatus and Zannichellia palustris subsp. Palustris) have an agreement between observed and predicted presence that is almost perfect.



.

Figure 11. Distribution of values of the Kappa criterion plotted against the number of species.

As explained in paragraph 2.4 a high Ĉ relates to a small p value, implying significant difference between observed and predicted values, and thus indicates a lack of fit of the model. A p value that is greater than 0.05 (test value < 15.51) indicates a good fit.

The figure below (Figure 12) shows that most of the values for the Hosmer-Lemeshow Criterion are relative low. Only a few species show a very large value for this criterion indicating a bad fit. Thus most of the regression models seem to have relatively good fit when this criterion is applied. However when the criterion is tested, most model fits (560 species) give a P-value smaller than 0.05 (Figure 13) which indicates a lack of fit. For a smaller number of species (354 species), the test shows a good fit. When a higher test value is taken (test value 20.1, p-value >= 0.01), there is a significant difference between observed and predicted values (lack of fit) for 424 regression models.

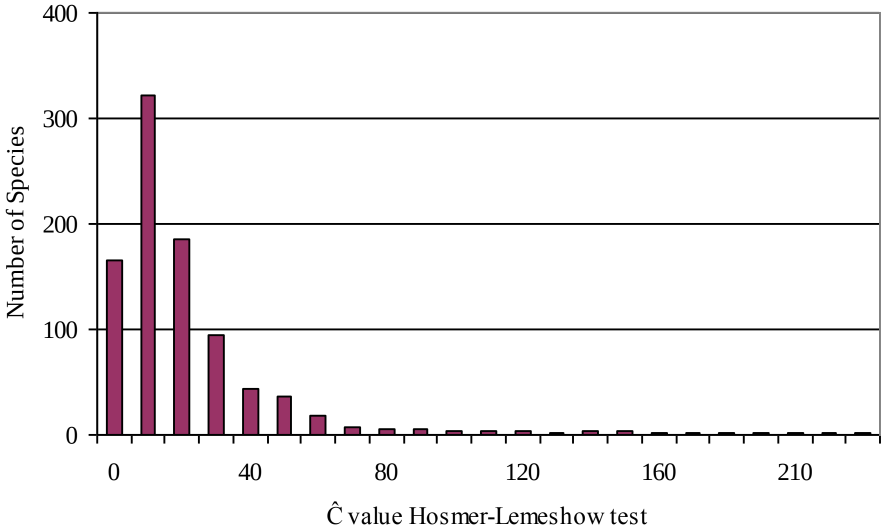


Figure 12. Distribution of Hosmer-Lemeshow criterion (X^2 Goodness-of-fit test) plotted against the number of modelled species

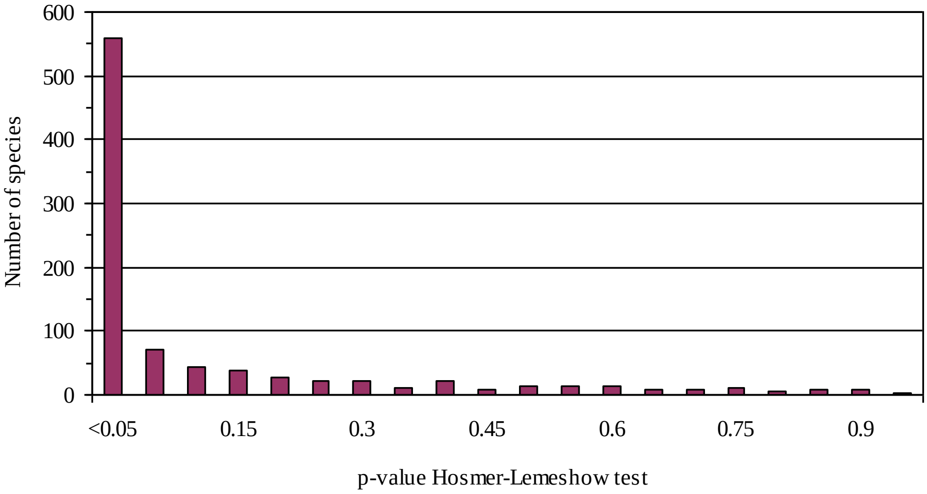


Figure 13. Probability of Hosmer-Lemeshow Criterion plotted against the number of modelled species.

4 Model validation methods and results

4.1 Model validation methods

A MOVE4 regression model gives two types of output: probability of occurrence (POO) and presence or absence. The predicted presence was found by first founding a critical POO which maximizes the Kappa statistic (statistical measure for agreement) and then using this critical POO as limiting value for rounding the POO to either absence (0) or presence (1).

The predicted probability of occurrence and predicted presence can be compared to the observed presences (field data) to validate the models. Note that this is not used in a cross-validation manner, because we used the same data to fit and to validate the models.

Move fits the logistic regression model for each species separately, using all relevés. Validation of these models is unwieldy because the response is either 0 (absence) or 1 (presence). To be able to estimate the goodness of fit of a regression model, the relevés were clustered based on the continuous environmental variables (Ellenberg *n*, *f*, *r*, and *s*). Each variable was split into five classes. A cluster thus has a certain range for every Ellenberg variable. Too small clusters (<10 relevés) were joined to the nearest, larger cluster. This resulted in 504 clusters with 10-3027 relevés per cluster. When relevés are clustered, the predicted POO can be compared to the observed POO, e.g. by summing the observed presences and the predicted POO values for each cluster. In this comparison, all clusters have the same weight, regardless of their size, since it is unknown why the cluster is small or large. A cluster may be small because it consists of relevés with rare indicator values and thus possibly also rare species, then this is an important cluster.

The results over all relevés were assessed in four ways (see below). For each of these assessments a score was given (0, 1 or 2), indicating the goodness of fit.

1 Difference between predicted and observed presence. The deviation between observed presence over all the relevés and the predicted presence (summed POO's or summed calculated presence) is divided by the root of the observed presence. Thus a difference between observed and predicted of for example 10% is given more weight when the total number of observations is large. At the same time small differences for rare species are weighted more lightly.

$$AbsoluteWeightedDeviation = \left| \frac{Predicted - Observed}{Observed^{0.5}} \right| \qquad [4]$$

score: If the Absolute Weighted Deviation is smaller than 2, score = 2
Else If the Absolute Weighted Deviation smaller than 4, score = 1
Else score = 0

2 Pearson correlation (R) or explained variance (R^2) between total observed presence and total predicted presence across the 504 clusters. In multiple regression (many independent variables) the adjusted R^2 is usually used, since this statistic takes into account the number of independent variable that is included in the model. However, when the number of observations is large compared to the number of independent variables the adjustment is negligible (Oude Voshaar, 1995).

Score: If R is smaller than 0, score = 0
Else If R^2 smaller than 0.0077, score = 0
Else If R^2 smaller than 0.7, score = 1
Else score = 2

There should be a positive correlation between the observed presence and predicted presence across clusters. When $R > 0.088$ ($R^2 > 0.0077$), there is a statistically significant correlation at the 5% level between observed and predicted presences across the clusters. An R^2 of 0.7 or higher means that there is a good fit between the response curve and the data. Note that these cut-off values are somewhat arbitrary.

Besides the previous criterion for a positive correlation between the observations and the predictions, there is also a criterion for the form of this relationship. The ideal regression line of $y=x$ indicates that the predicted data match the observed data exactly. This is a line with a zero intercept and a regression coefficient that equals 1. When this line is approximated it means that the observed presence on average is equal to the predicted presence.

Orthogonal regression, also known as major axis regression, was used to fit a linear relationship between observed and predicted data across clusters. This regression method minimizes the sum of squared residuals which are found by orthogonal projection of the (x, y) points onto the regression line. The rationale for this is that in ordinary regression it is assumed that the independent variable is under the control of the investigator (Laws and Archie, 1981). However for field data such as ours this is often not the case and then an alternative is offered by orthogonal regression.

The scores for the regression intercept a and the regression coefficient b were calculated as given below. Again the cut-off values are rather arbitrary.

3 a = regression intercept, L1a = lower limit 95% confidence interval of intercept, L2a = upper limit 95% confidence interval of intercept

Score: If the lower limit, L1a is larger than 0, score = 0
Else if the upper limit, L2a is smaller than 0, score = 0
Else score = 2

When 0 lies within the 95% confidence interval of the regression intercept, than the validation score for the intercept is high.

4 b = regression coefficient, L1b = lower limit 95% confidence interval of regression coefficient, L2b = upper limit 95% confidence interval of regression coefficient

Score: If the lower limit, L1b is larger than 1, score = 0
Else if the upper limit, L2b is smaller than 1, score = 0
Else score = 2

When 1 lies within the 95% confidence interval of the regression coefficient, than the validation score for the coefficient is high.

Finally a total validation score was calculated as follows:

Validation score = Score WeightedDeviation (ad 1) + Score Correlation (ad 2) + Score Regression intercept (ad 4) + Score Regression coefficient (ad 4)

Thus the total score ranges from 0-8, where 0 indicates a bad relation between observed and modelled probability of occurrence and 8 a very good relation between observed and modelled probability of occurrence.

4.2 Validation results for predicted POO

In this paragraph the observed presences are compared to the predicted probability of occurrence (POO).

%Deviation

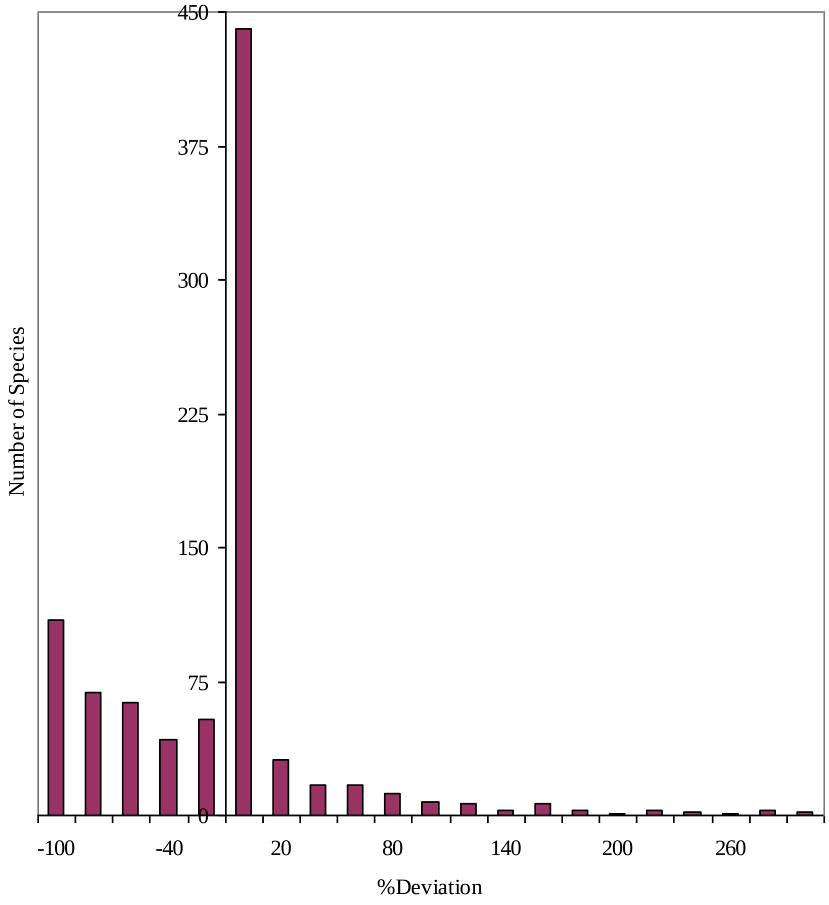


Figure 14. Distribution of %Deviation between total observed and total predicted POO for each species (>300%: 16 species). 441 species have a %Deviation between 0% and 20%.

The percentage deviation for a species is calculated as the relative value of the difference between the predicted POO summed for all clusters and the observed POO summed for all clusters, see equation [4]. 283 species show a %Deviation between -100% and -20%. A %Deviation of -20% means that the total predicted POO is 80% of the total observed presence (Figure 14). For example, it was recorded in 1000 relevés, but summed over all clusters it was found to be present in only 800 relevés. 54 species have a %Deviation between -20% and 0%, and species 441 species have a %Deviation between 0% and 20%.

Correlation

Figure 15 shows that most species have a high correlation between total observed presence and total predicted POO; 555 species had a correlation ≥ 0.7. With 502 degrees of freedom, a correlation coefficient of 0.088 is significant ($p<0.05$). This corresponds with an R^2 of 0.0077 (paragraph 2.6). For 29 of the 911 species the correlation was not significant (Figure 16).

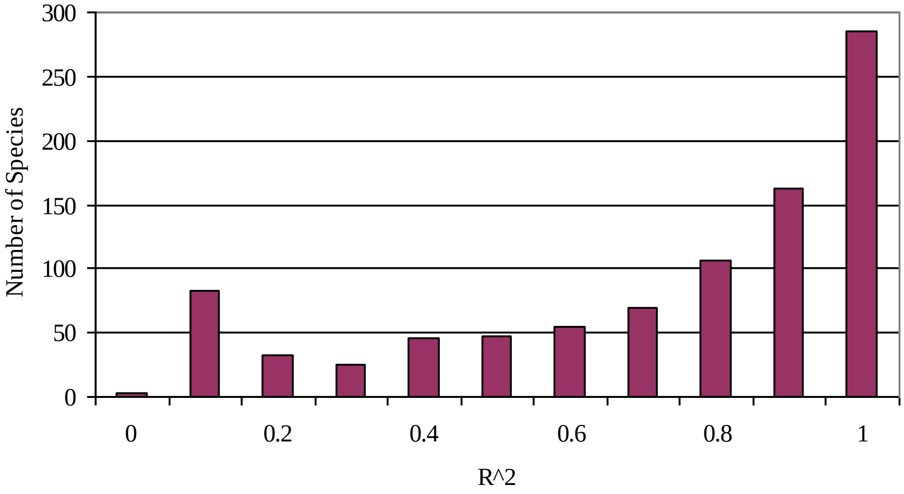


Figure 15. Distribution of Pearson correlation (R^2) between total observed presence and total predicted POO for the modelled species.

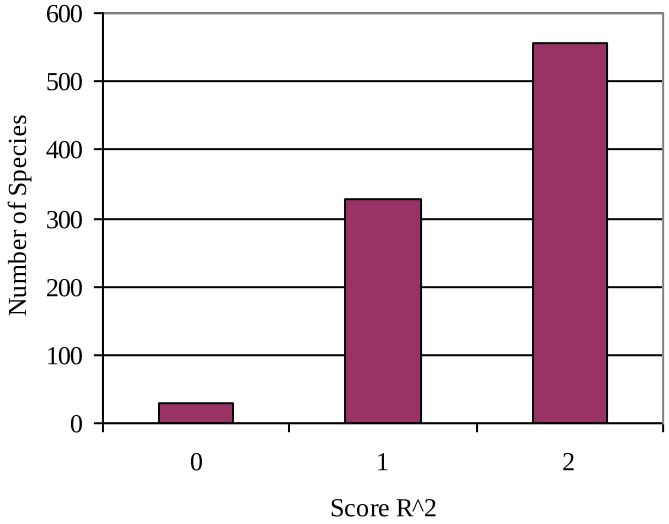


Figure 16. Number of species per correlation score (0: $R^2<0.007744$, 1: $R^2>=0.007744$ and $R^2<0.7$, 2: $R^2>= 0.7$).

Regression Analysis Intercept

Regression equations were derived per species to describe the relation between observed and predicted presences. Per species the resemblance of the relation between observed and predicted presences for each cluster to a zero intercept and a 1:1 relation (regression analysis) was evaluated.

In most cases the regression intercept was significantly higher than zero (465 species). Only some equations had a regression intercept significantly lower than zero (60 species). 386 species had a regression intercept not significantly different from 0 (Figure 17), agreeing with the ideal 1:1 line.

A small confidence interval indicates fairly consistent data values; a large confidence interval indicates a great deal of variability in the data. Figure 18 shows that most confidence interval ranges, calculated for the regression intercept (paragraph 2.6) are rather small, i.e. between 0 and 1, which indicates that the estimates of the intercept may be rather precise. For some species, however, the confidence interval is very large (seven species have an intercept confidence interval above 10), which indicates that these estimations may be very imprecise.

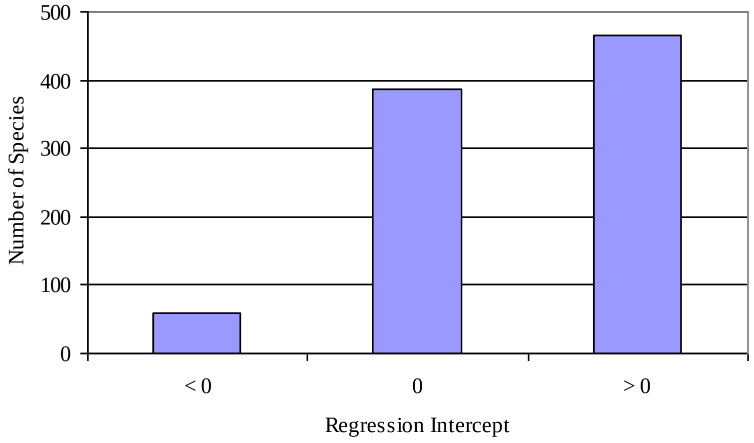


Figure 17. Number of Species for which the estimated intercept of the regression is significantly smaller than zero, not significantly different from zero or significantly larger than zero.

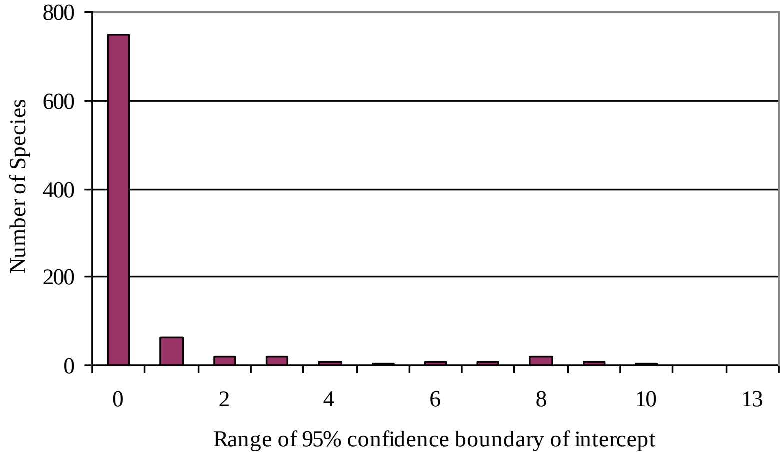


Figure 18. Ranges of 95% confidence intervals of the intercept using regression of predicted presence to observed presence for each cluster.

Regression Analysis Slope

The regression slope was in most cases significantly lower than the ideal value of 1 (736 species, Figure 19), and only in a few cases not significantly different from 1. A regression equation with an intercept larger than 0 and a slope lower than 1 (this occurred for 400 species) will in most cases result in an overestimation of the occurrence of rare species and an underestimation of the occurrence of more common species.

Most confidence interval ranges for the regression slope are very small; between 0 and 0.1, only thirteen species have an interval greater than 1 (Figure 20). This indicates that the estimations of the slope may be very precise.

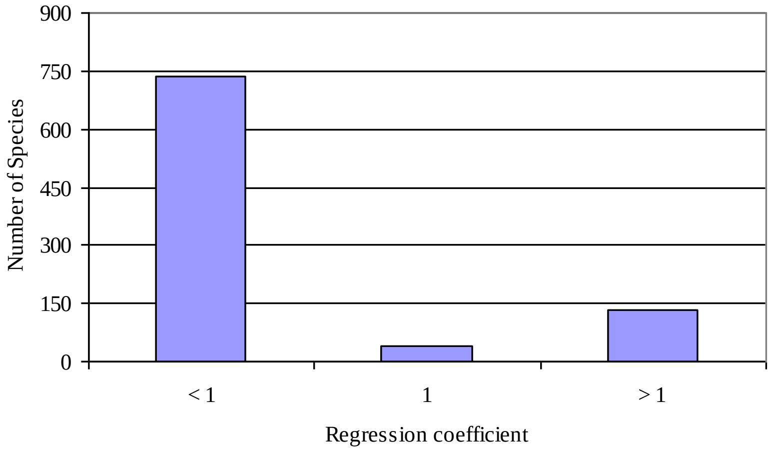


Figure 19. Number of species for which the estimated slope of the regression is significantly smaller than 1, not significantly different from 1 or significantly higher than 1

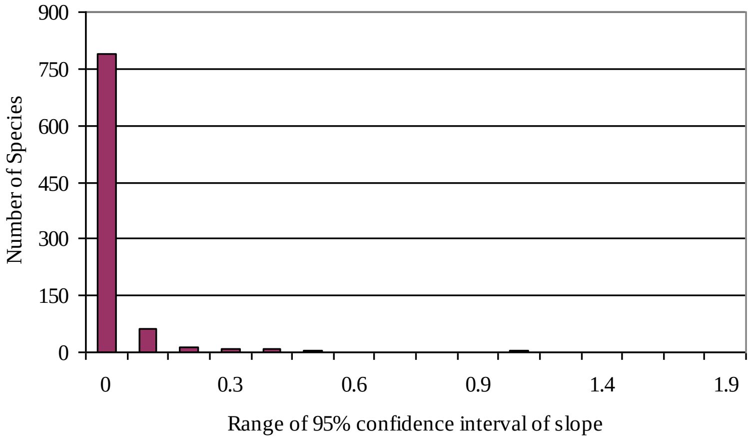


Figure 20. Ranges of 95% confidence intervals of the slope using regression of predicted presence to observed presence for each cluster. Six species with a range larger than 2 are not shown.

Total validation score

453 species received a total validation score of 0 to 3, 431 species received a total validation score of 4 to 7 (of which 256 species had a score of 4, Figure 21). Only 27 species reached the maximum score of 8 points (see table 8 for the species). Thus lower scores are more abundant than higher scores. Overall the scores are not too good; indicating that many species do not meet the criteria. The species that have the maximum score do not seem to

have much in common; rare as well as common species are present, and trees as well as forbs. There is however only one moss species that reaches the maximum score.

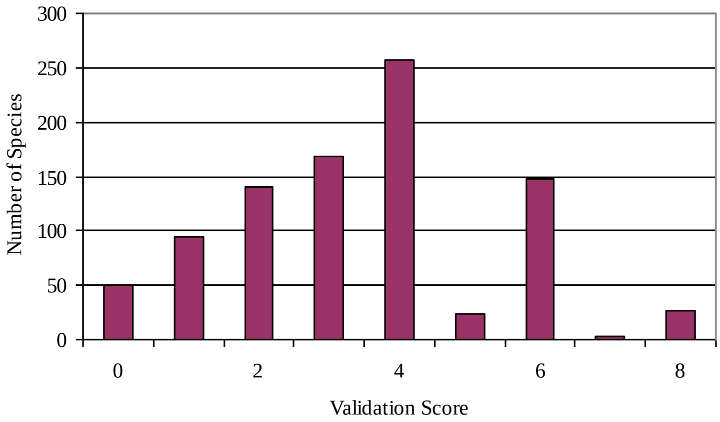


Figure 21. Distribution of total validation scores over the species that were modelled.

Table 8. Species with maximum validation score.

Taxon name	**Dutch name**
Brachypodium pinnatum	Gevinde kortsteel
Cakile maritima	Zeeraket
Calluna vulgaris	Struikhei
Atriplex prostrata	Spiesmelde
Corynephorus canescens	Buntgras
Erodium glutinosum	Kleverige reigersbek
Gentiana pneumonanthe	Klokjesgentiaan
Hippuris vulgaris	Lidsteng
Juncus alpinoarticulatus subsp. atricapillus	Duinrus s.s.
Lobelia dortmanna	Waterlobelia
Matricaria maritima	Reukeloze kamille
Melampyrum pratense	Hengel
Myosotis ramosissima	Ruw vergeet-mij-nietje
Populus alba	Witte abeel
Populus tremula	Ratelpopulier
Potamogeton polygonifolius	Duizendknoopfonteinkruid
Quercus petraea	Wintereik
Rhynchospora fusca	Bruine snavelbies
Saxifraga tridactylites	Kandelaartje
Festuca ovina subsp. tenuifolia	Fijn schapegras
Dactylorhiza majalis	Brede orchis
Hydrodictyon reticulatum	Waternetje
Chara globularis	Breekbaar kransblad
Chara vulgaris	Gewoon kransblad
Chara species	Kranswier
Nitella flexilis	Buigzaam glanswier
Senecio jacobaea	Jakobskruiskruid s.l.

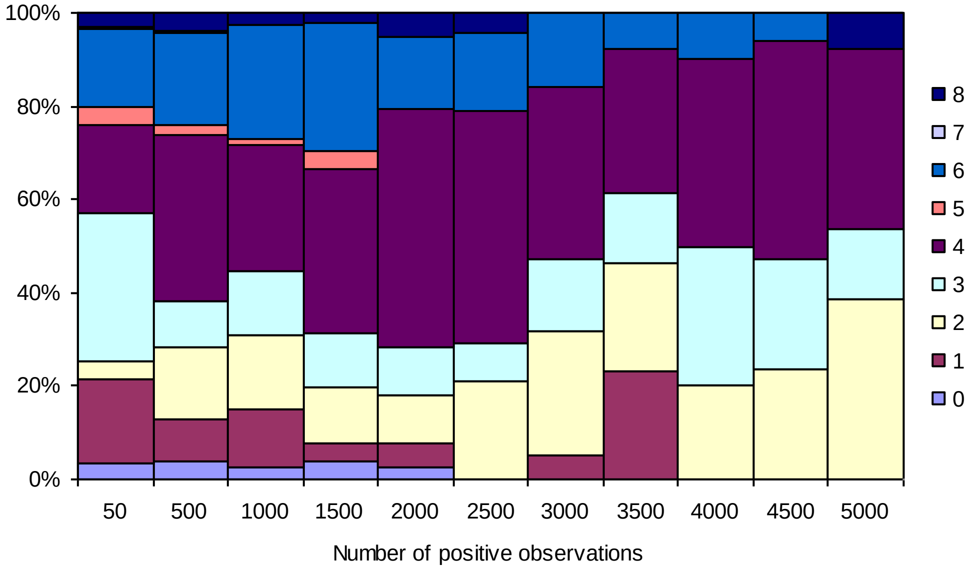
Validation score graphs

In the following 8 figures the relative distribution of the validation scores is represented as a function of the following previously mentioned variables:

1. Number of positive observations

2. Number of positive observations (lower presences)
3. Maximum probability of occurrence
4. Percentage deviance explained
5. Ĉ value Hosmer-Lemeshow test
6. Optimal probability of occurrence
7. Maximal value of the Kappa statistic
8. Number of model terms

At 50 to 500 positive observations the scores 1-6 and 8 are distributed rather evenly (Figure 22). Above 4000 positive observations there are no more scores of 0, 1 and 5 and only a few of 6 and 8. Thus common species mostly have total validation scores of 2-4.



pos obs	50	500	1000	1500	2000	2500	3000	3500	4000	4500	5000
# species	423	134	81	51	39	24	19	13	10	17	13

Figure 22. Distribution of total validation scores as a function of the number of positive observations in the dataset (# positive observations > 5000: 87 species).

At a low number of positive observations, the relative distribution of low scores (0 t/m 3) decrease and high scores (4, 6 en 8) increase as a function of positive observations (Figure 23). Thus rare species tend to have lower total validation scores. This is probably caused by lack of data, making the fitting of a proper regression model difficult. The minimum number of positive observations to obtain a reasonable fit seems to lie somewhere between 300 and 400. This minimum number is probably dependent on the species, but also of the composition of the calibration dataset. The chance that important circumstances for the species occurrence are lacking is larger when less data are available. However, the overrepresentation of certain (abiotic) circumstances may also lead to less good regression models.

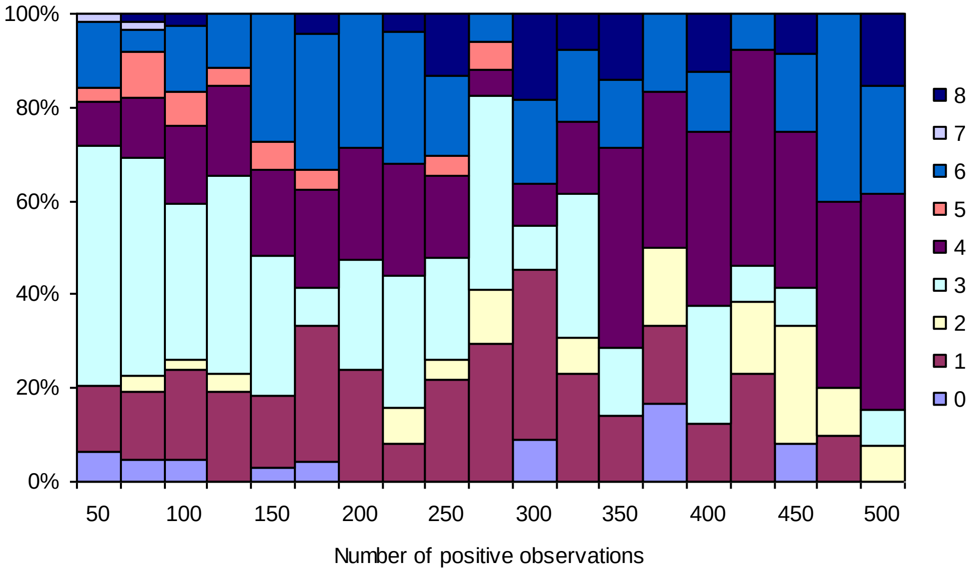
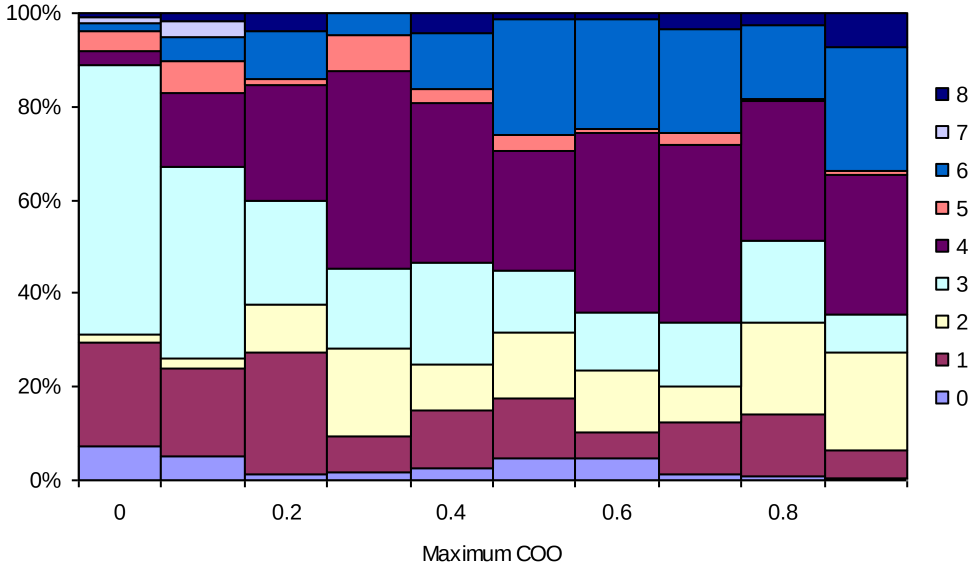


Figure 23. Distribution of validation scores at lower number of positive observations.

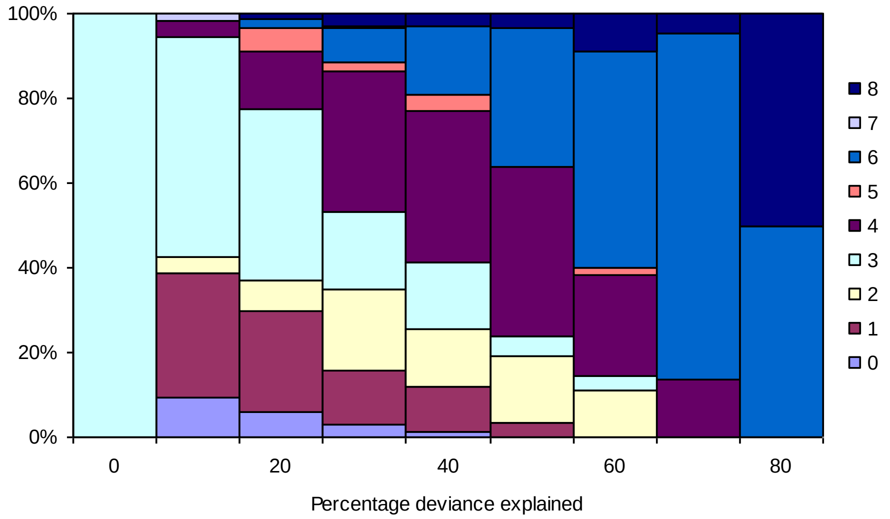
As the maximum POO increases, low validation scores decrease and high validation scores increase (Figure 24). Species with a high maximum probability of occurrence are probably the more common species, so the trend in this graph is comparable to the trend in the previous graphs: common species have higher total validation scores than rare species.



Max POO	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
# species	99	58	77	64	73	85	89	89	121	156

Figure 24. Maximum probability of occurrence plotted against the total validation score.

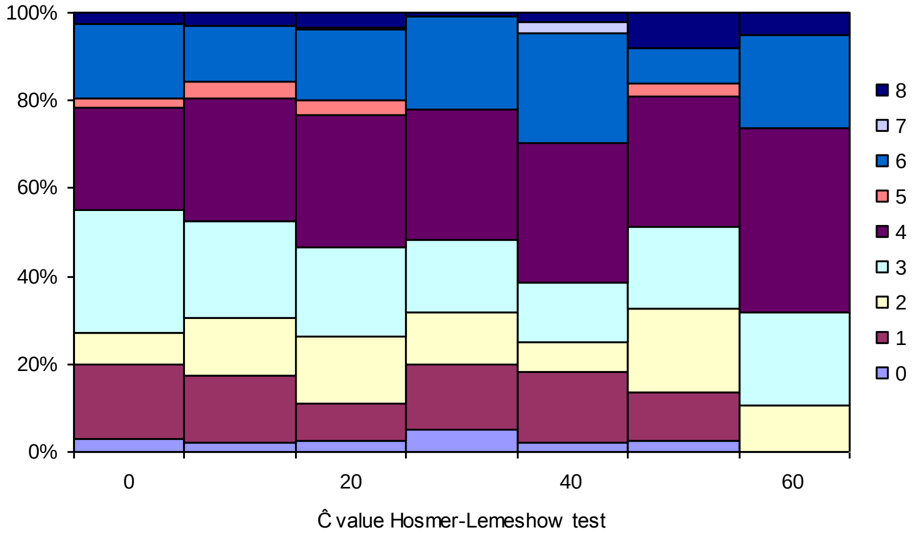
As percentage deviance explained increases, low validation scores decrease and high validation scores increase (Figure 25). So the percentage deviance explained is a good indicator of the total validation score of the species regression models.



PercDevExpl	0	10	20	30	40	50	60	70	80
# species	6	54	167	252	232	121	55	22	2

Figure 25. Percentage Deviance Explained plotted against the total validation score.

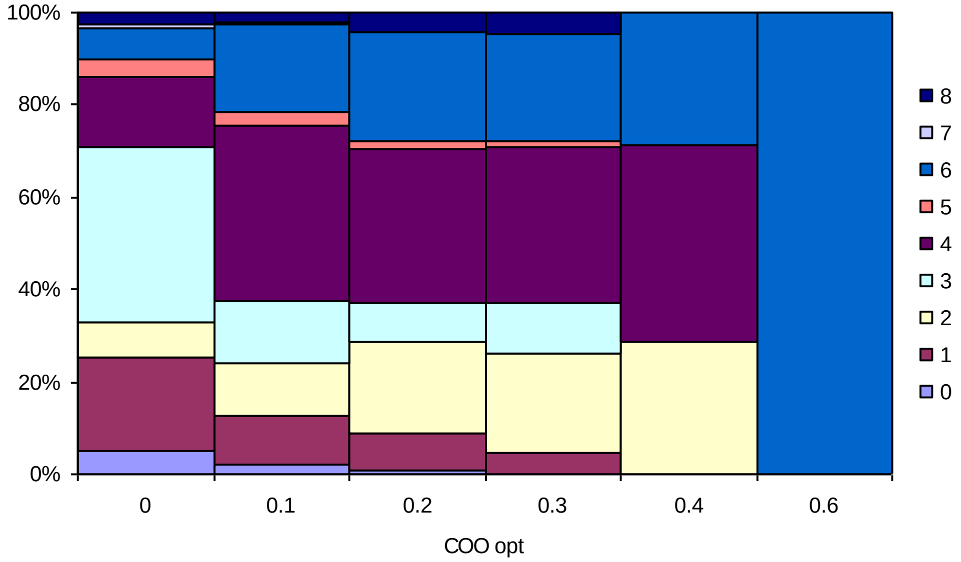
From class 70 onwards, the number of species per class is very low (≤7); therefore these are not shown in the graph (Figure 26). In this graph there is not a very clear trend. One could conclude that the total validations scores slightly increase as the Ĉ value increases.



Ĉ value	0	10	20	30	40	50	60
# species	165	321	184	95	44	37	19

Figure 26. Chi-criterion Hosmer-Lemeshow, Ĉ value >70: 46 species

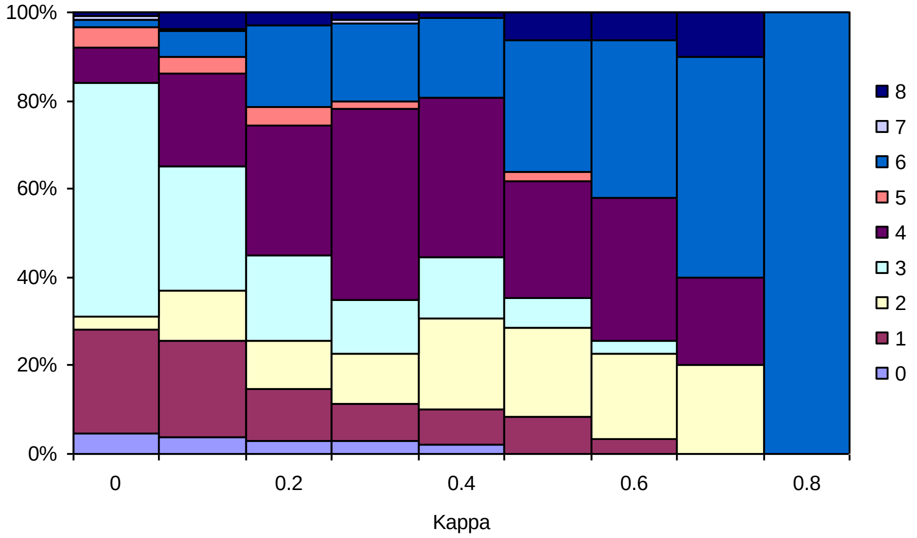
As the critical POO increases, low validation scores decrease and higher validation scores increase (Figure 27). Species with a high maximum probability of occurrence (pMaxExp) might be more common species, so the trend in this graph might be compared to the trend in the previous graphs: common species have higher total validation scores than rare species. Here also the relation between the total validation score of the model performance and the single indicator, the critical POO, is quite strong.



POO opt	0	0.1	0.2	0.3	0.4	0.6
# species	339	274	218	65	14	1

Figure 27. Critical POO plotted against the total validation score.

At higher Kappas, there are relatively higher validation scores (Figure 28). Again, this is in line with the total validation score.



kappa	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8
# species	125	160	169	160	150	94	31	20	2

Figure 28. Kappa plotted against total validation score.

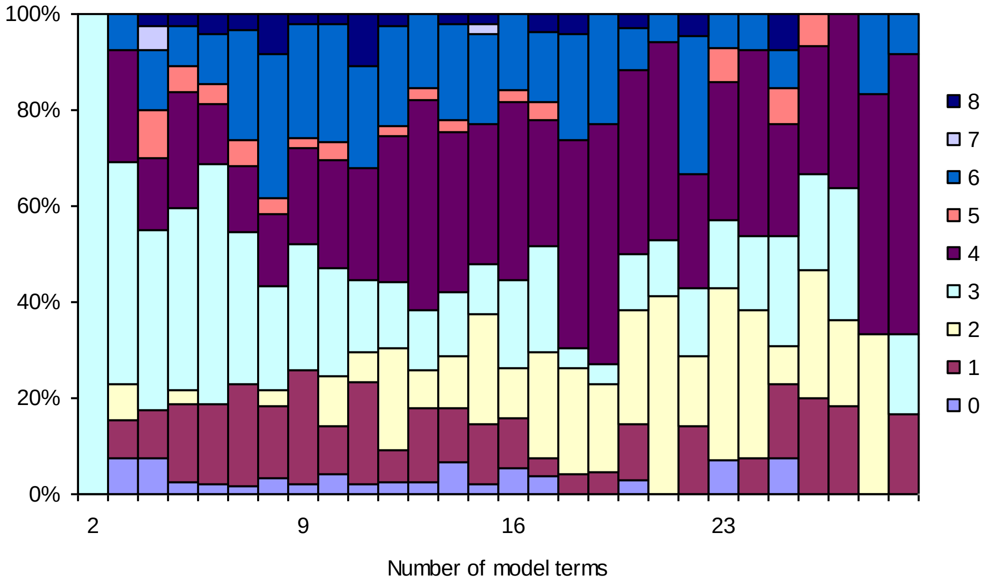


Figure 29. Number of model terms in the response equations plotted against total validation score.

Figure 29 shows the number of model terms that were included in the response equations, plotted against the number of species. The higher validation scores (5, 6, 7 and 8) are mostly found at a lower number of model terms (4 to 25, Figure 29). Many model terms do not necessarily improve the performance of the model. It could be interesting to test whether setting a maximum to the included model terms would improve the performance of the models. As the number of species with models containing more than 29 terms is low, these were omitted from the figure (30 to 51 model terms: 67 species).

4.3 Validation results for predicted presence

In this paragraph the observed presences are compared to the predicted presences using the critical POO. Since the methods for cross-validation used in this paragraph are the same as those used in the previous paragraph, the explanations in this paragraph will be somewhat shorter.

%Deviation

59 species have a %Deviation between -20% and 0%, and 70 species have a %Deviation between 0% and 20% (Figure 31). A %Deviation of -20% means e.g. that it was recorded in 1000 relevés, but summed over all clusters it was found to be present in only 800 relevés.

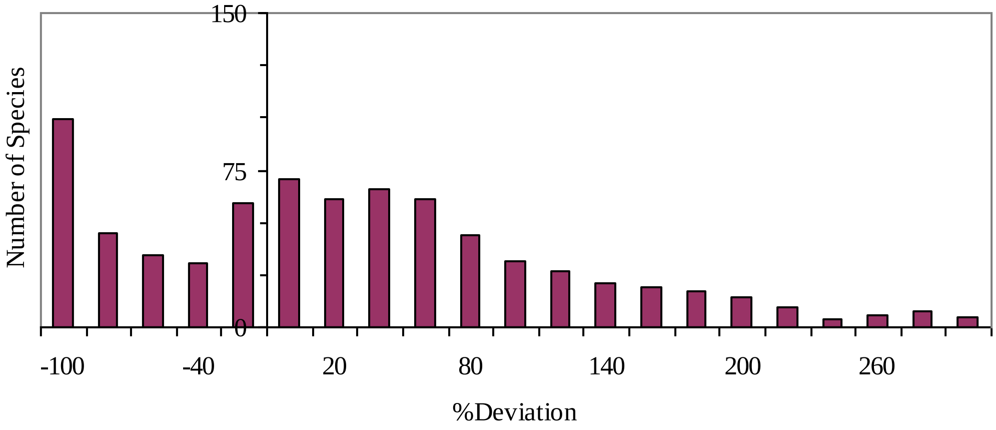


Figure 31. Distribution of %Deviation between total observed and total predicted presence for each species (>300%: 100 species).

Correlation

Figure 32 shows that most species have a high correlation between total observed presence and total predicted presence; 397 species had a correlation ≥ 0.7. With 502 degrees of freedom, a correlation coefficient of 0.088 is significant ($p<0.05$). This corresponds with an R^2 of about 0.0077 (paragraph 2.6). Of 61 species the correlation was not significant (Figure 33).

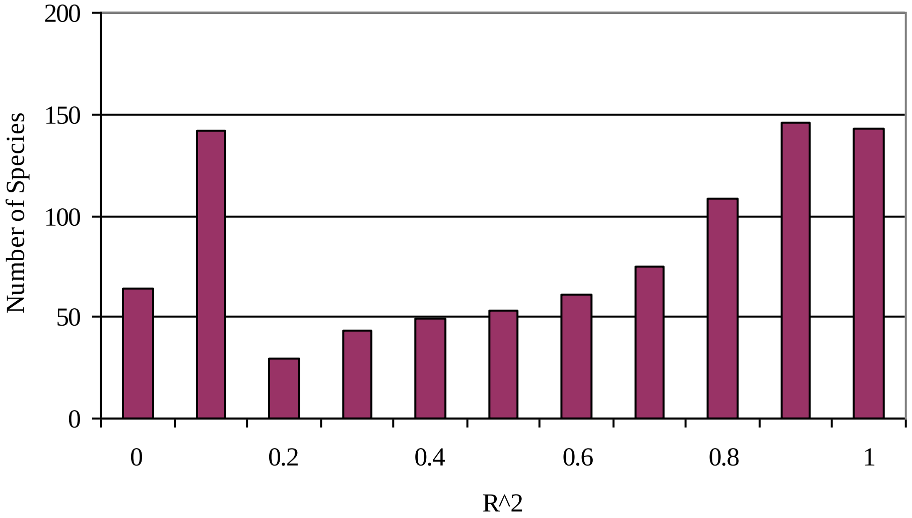


Figure 32. Distribution of Pearson correlation (R^2) between total observed presence and total predicted POO for the modelled species.

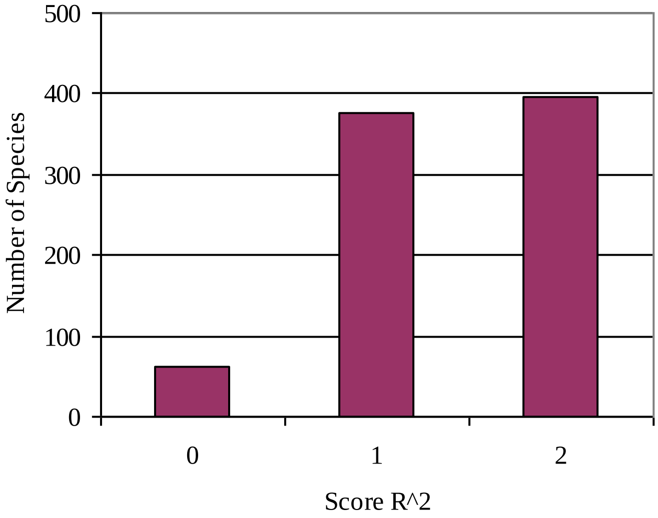


Figure 33. Number of species per correlation score (0: $R^2<0.007744$, 1: $R^2>=0.007744$ and $R^2<0.7$, 2: $R^2>= 0.7$)

Regression Analysis Intercept

For most species the regression intercept was significantly lower than zero (486), few models had a regression intercept significantly larger than zero (42 species), 306 species had a regression intercept not significantly different from zero (Figure 34).

Most confidence intervals for the regression intercept are rather small; between 0 and 1 (Figure 35). This indicates that these estimations of the intercept may be rather precise.
A considerable amount of species has a larger confidence interval, between 1 and 3 and some species have an even larger confidence interval (27 species have an intercept confidence interval above 10. This indicates that these estimations may be very imprecise.

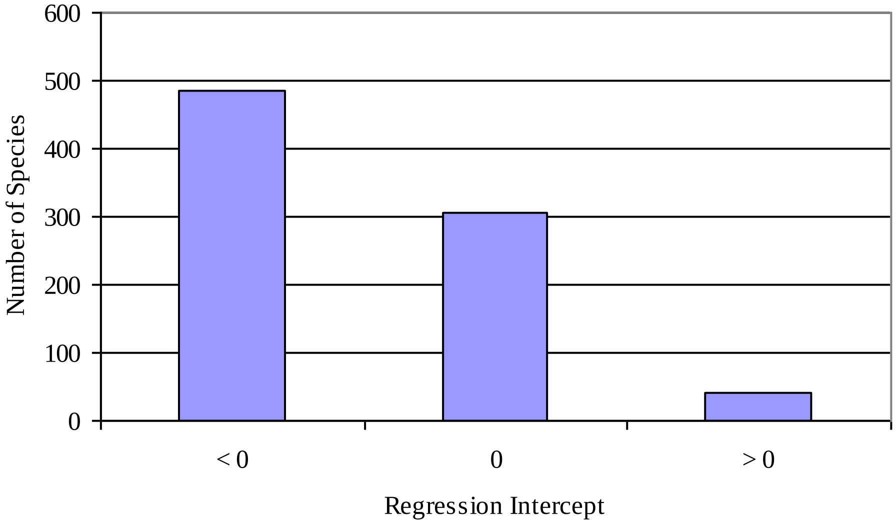


Figure 34. Number of species for which the estimated intercept of the regression is significantly smaller than 0, not significantly different from 0 or significantly higher than 0.

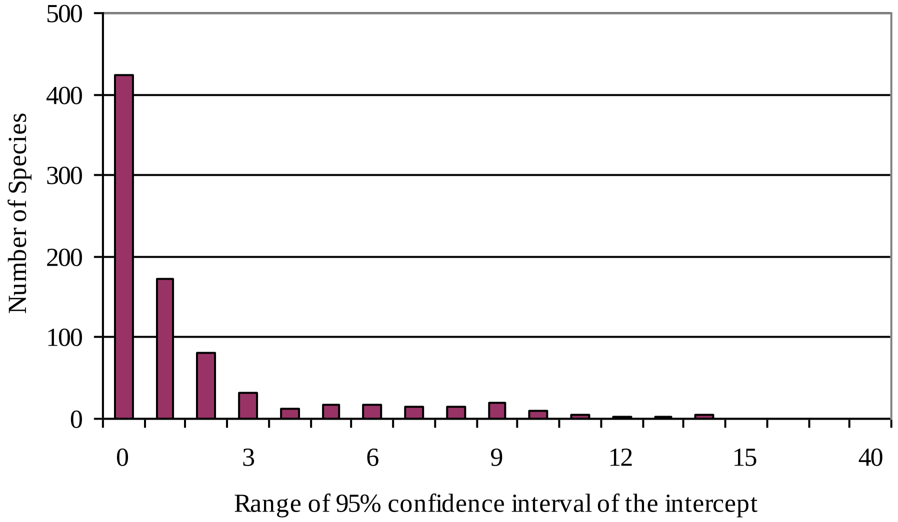


Figure 35. Ranges of 95% confidence intervals of the intercept using regression of predicted presence to observed presence for each cluster (Model II).

Regression analysis slope

The regression slope was in most cases larger than 1 and only in a few cases close to 1 (Figure 36). As described above the regression intercept was in most cases significantly smaller than 0. This is in contrast to the predicted POO, where the regression intercept was mostly significantly larger than 0 and the slope was mostly smaller than 1.

A regression equation with an intercept smaller than 0 and a slope larger than 1 may result in an underestimation of the occurrence of rare species, and in an overestimation of the occurrence of more common species. This is the case for 604 species. So the occurrence of many (common) species may be overestimated by MOVE4, compared to its own calibration set, when using the predicted presence.

Most confidence interval ranges for the regression slope are very small; between 0 and 0.1. This indicates that these estimations of the slope may be very precise. A considerable amount of species has a confidence interval between 0.1 and 0.3 and 49 species have an interval greater than 1 (Figure 37), which means that these estimations may be less precise.

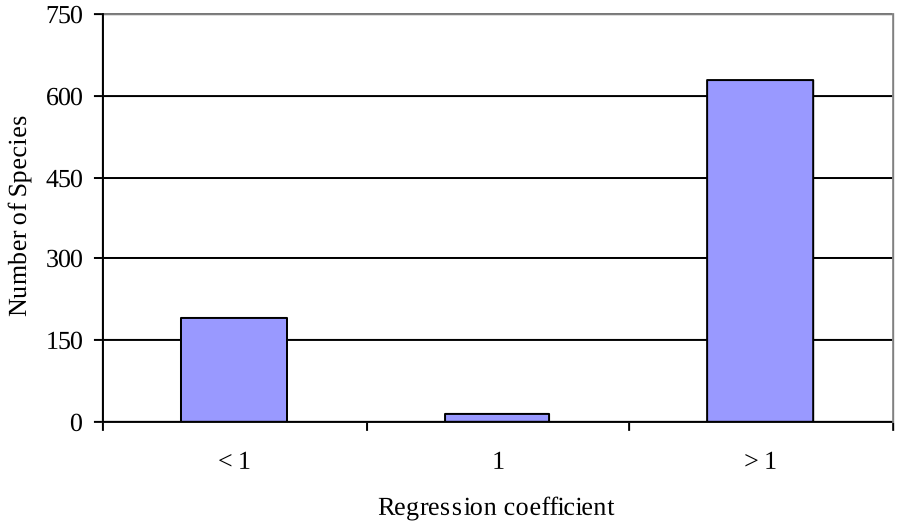


Figure 36. Number of species for which the estimated slope of the regression is significantly smaller than 1, not significantly different from 1 or significantly higher than 1.

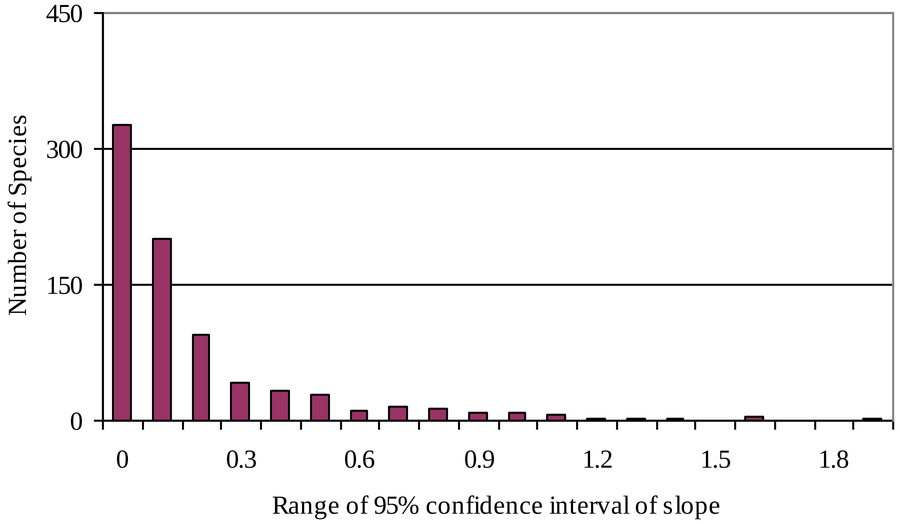


Figure 37. Ranges of 95% confidence intervals of the slope using regression of predicted presence to observed presence for each cluster (Model II). (>2: 26 species)

Total validation score

645 species received a total validation score of 0 to 3, 189 species received a total validation score of 4 to 7 (appendix 5). None of the species reached the maximum score of 8 points (Figure 38). The lower scores are much more abundant than higher scores. Overall the scores are not too good; indicating that many species do not meet the criteria.

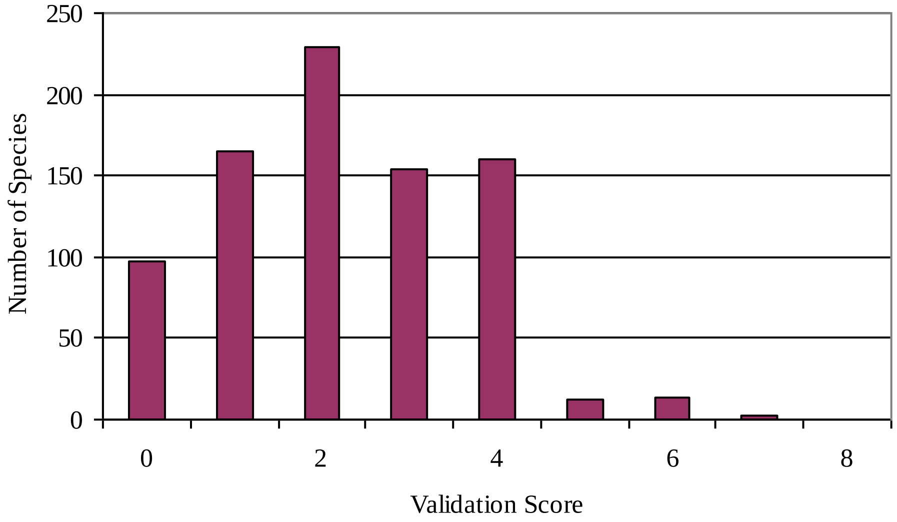
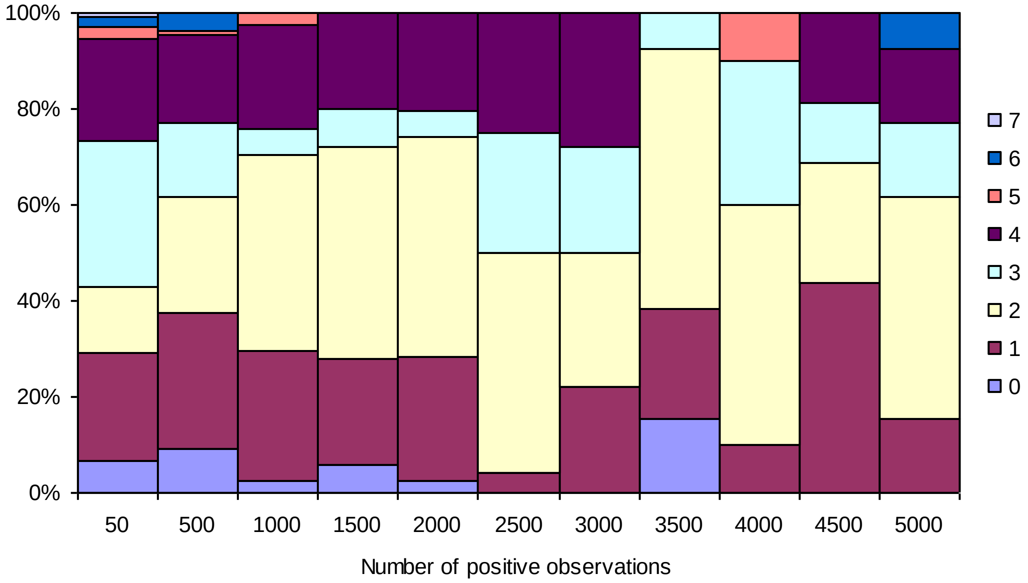


Figure 38. Distribution of total validation scores over the modelled species.

Validation score graphs

At 50 to 500 positive observations the scores 1-6 and 8 are distributed rather evenly (Figure 39). Above 4000 positive observations there are no more scores of 0 and only a few of 5 and 6. Thus common species mostly have total validation scores of 1-4.



pos obs	50	500	1000	1500	2000	2500	3000	3500	4000	4500	5000
# species	355	131	78	50	39	24	18	13	10	16	13

Figure 39. Distribution of total validation scores as a function of the number of positive observations in the dataset (# positive observations > 5000: 87 species).

Figure 40 shows the number of positive observations up to 500 observations, plotted against the total validation score. There does not seem to be a clear trend in this graph.

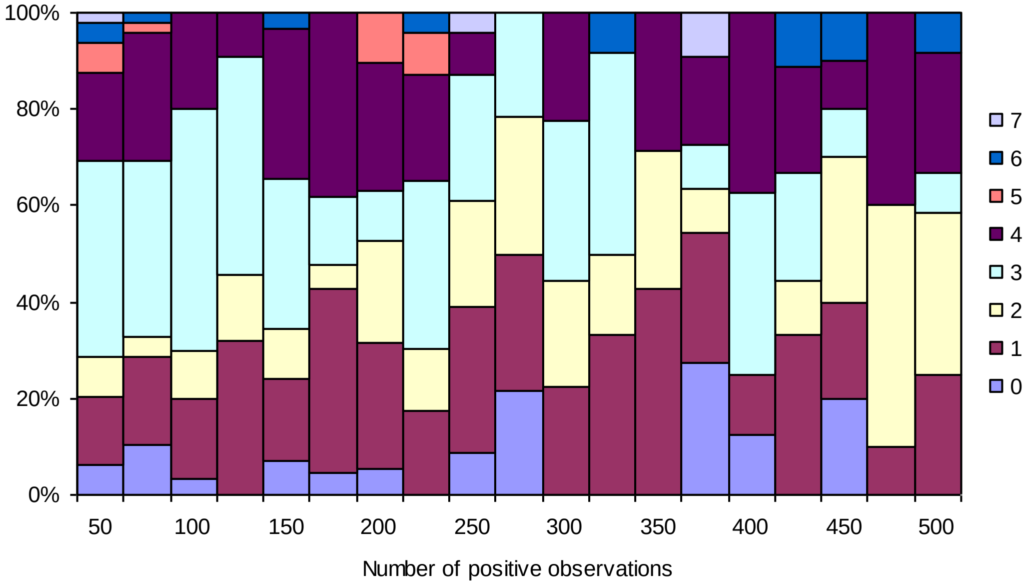
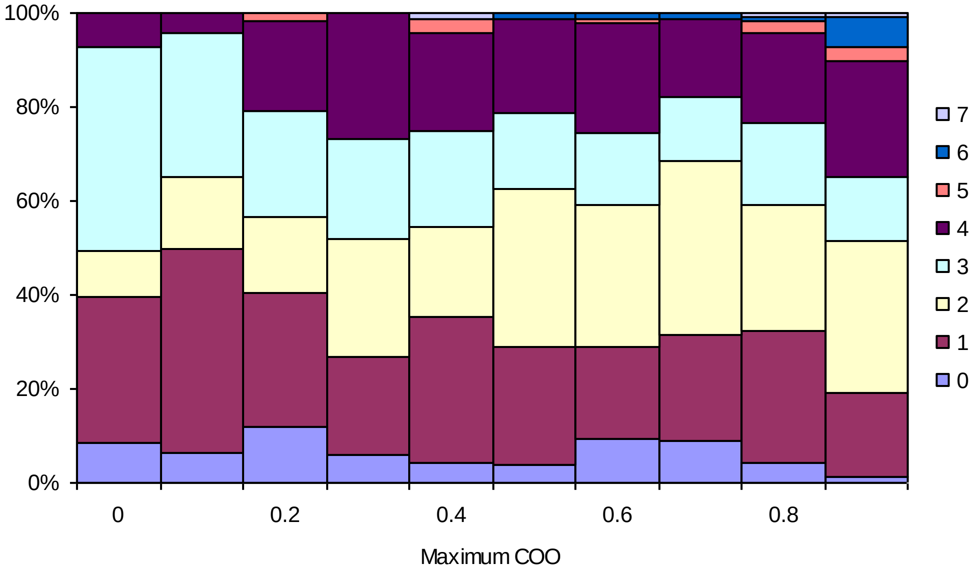


Figure 40. Distribution of total validation scores at lower number of positive observations.

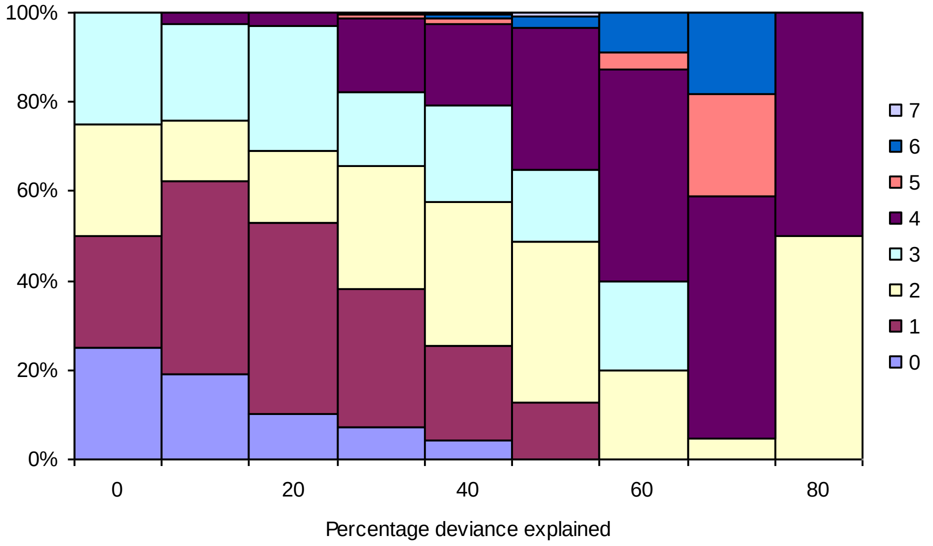
As the maximum POO increases, low validation scores decrease and high validation scores increase (Figure 41). Species with a high maximum probability of occurrence are probably the more common species, so the trend in this graph is comparable to the trend in the previous graphs: common species have higher total validation scores than rare species.



Max POO	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
# species	71	46	67	52	68	80	86	89	120	155

Figure 41. Maximum probability of occurrence plotted against the total validation score.

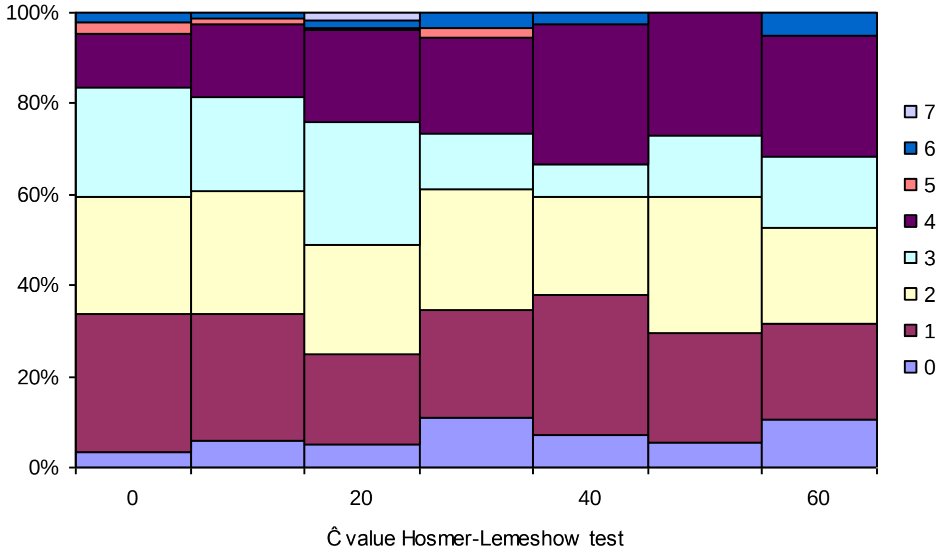
As the percentage deviance explained increases, low validation scores decrease and high validation scores increase (Figure 42). So the percentage deviance explained is a good indicator of the total validation score of the species regression models.



PercDevExpl	0	10	20	30	40	50	60	70	80
# species	4	37	145	230	220	119	55	22	2

Figure 42. Percentage Deviance Explained plotted against the total validation score.

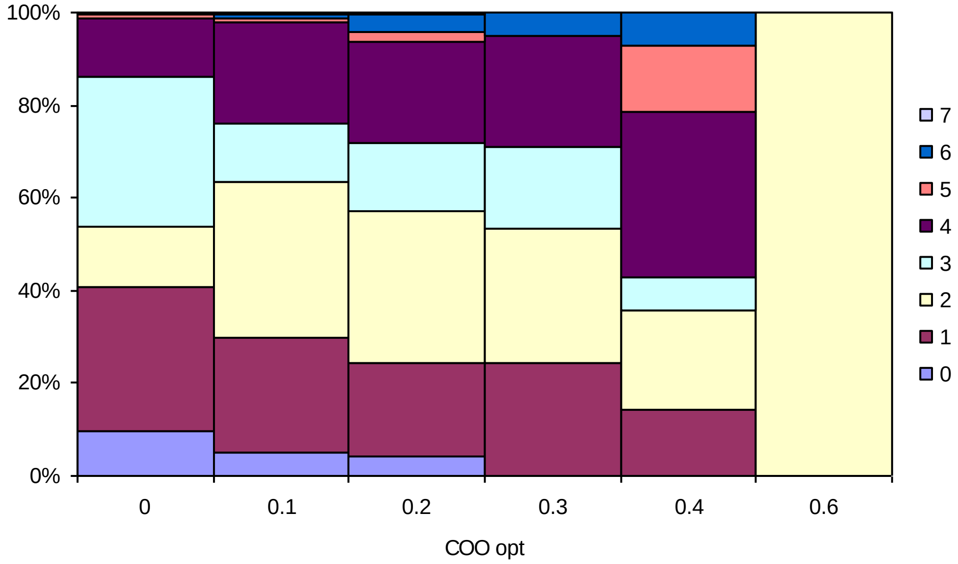
From class 70 onwards, the number of species per class is very low (≤6); therefore these are not shown in the graph (Figure 43).



Ĉ value	0	10	20	30	40	50	60
# species	146	283	176	90	42	37	19

Figure 43. Chi-criterion Hosmer-Lemeshow, Ĉ value >70: 41 species

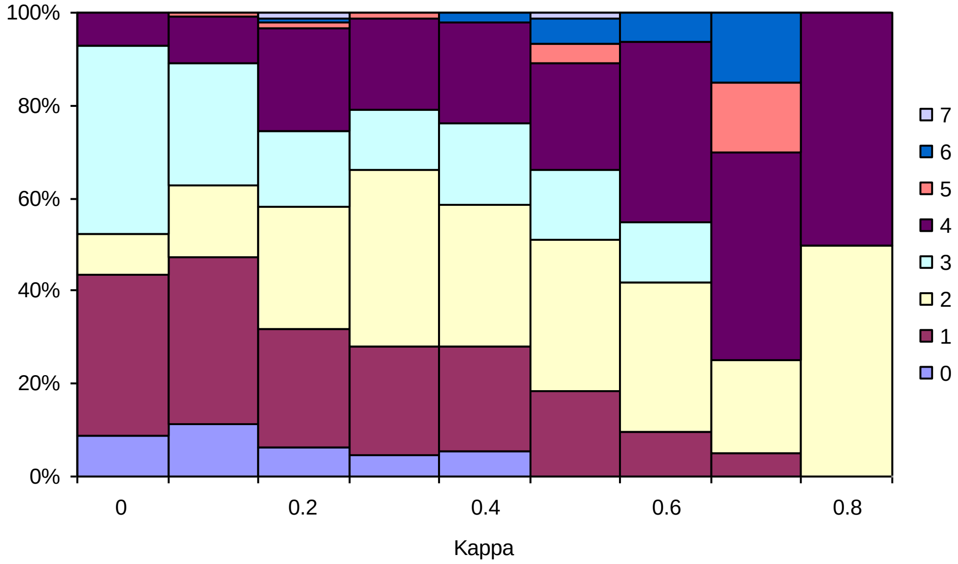
As the critical POO increases, low validation scores decrease and higher validation scores increase (Figure 44). Here also the relation between the total validation score of the model performance and the single indicator, the critical POO, is quite strong.



POO opt	0	0.1	0.2	0.3	0.4	0.6
# species	278	261	218	62	14	1

Figure 44. Critical POO plotted against the total validation score.

At higher Kappas there are relatively more high validation scores (Figure 45). Again, this is in line with the total validation score.



Kappa	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8
# species	101	131	157	157	143	92	31	20	2

Figure 45. Kappa plotted against total validation score.

The small percentage of higher validation scores (5, 6 and 7) are mostly found at a lower number of model terms (4 to 17, Figure 46). As the number of species with models containing more than 29 terms is low, these were omitted from the figure (30 to 51 model terms: 67 species).

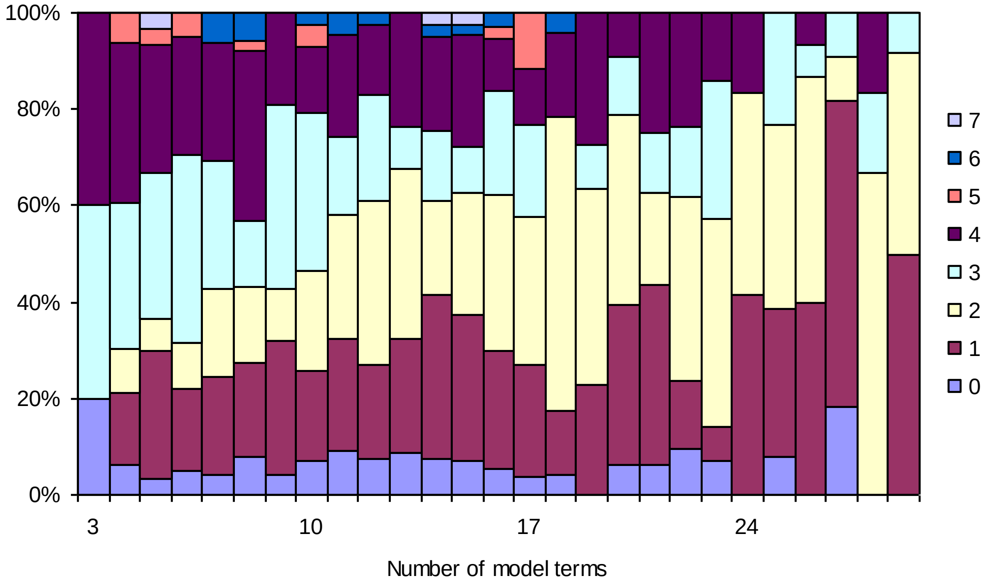


Figure 46. Number of model terms in the response equations plotted against total validation score.

5 Discussion and conclusion

The model MOVE4 was calibrated for over 900 plant species, thus allowing for the prediction of the probability of occurrence of a major part of the Dutch flora. One of the underlying assumptions within the method is that the relevés are evenly distributed and selected randomly in the field. However, when relevés are known to be taken at hotspots of biodiversity and undisturbed sites, than models derived from such data may be biased. To avoid this effect, one should test whether the combination of environmental factors presented to the model is well within the multidimensional space of the data from which the model was derived. As the model results may be used for calculations for nature areas and potential biodiversity hotspots in particular, this bias may be nullified.

For each species a logistic regression model was selected with plant species specific model terms. The number of terms in the individual species models ranged from less than 10 to over 30 terms. The percentage deviance explained of the species regression models ranges between 4.5% and 83.0% (all highly statistically significant; $p<0.001$, median: 39.1%, average 39.8%). This percentage deviance explained is not very high; however, considering the type of data (vegetation relevés) it is acceptable. The average percentage deviance explained is slightly higher than in the previous version MOVE3.2 (39.3%). The average maximum Probability Of Occurrence (POO) is higher (0.47 for MOVE3.2 to 0.56 for MOVE4). 30% of the species have a maximum POO higher than 0.80; for MOVE3.2 this was only 14%. This indicates that a larger group of species has a higher chance to be present under certain circumstances in the applied database. In other words, a greater specificity of the models was enabled by the alternative specification of the categorical variables as dummies, and/or by the greater number of possible model terms offered.

It must be taken into account that the current set of models under the name of MOVE4 is not just a re-computed set of models from the same data as used in MOVE 3 (actually MOVE 3.2 as described in Bakkenes *et al.*, 2002a,b). One environmental variable was skipped from the models (the combined potential affected fraction due to heavy metals) and the categorical variables vegetation structure and geographical region were transformed into binary variables. Thus a direct comparison between percentages deviance explained does not show the complete picture. Because of the great uncertainty in the estimates of the potential effects of heavy metals and the use of binary variables, we feel that the interpretability of the models is improved. In the case of *Eriophorum angustatum* (as exemplified in Bakkenes *et al*, 2002) only geographical regions Fenland and Sea clay area and the vegetation types Heathland and Grassland were included in the model in MOVE4. Thus spurious results, in this case for the geographical region Hills in MOVE 3.2, are prevented in MOVE4.

The automatic stepwise regression method allows for the inclusion of quadratic terms in the equation irrespective of whether the corresponding linear terms were or were not present in the equation already. This did not limit the models to having only one of each for a specific environmental variable. The environmental variables that could have both linear and quadratic terms in the regression equations, the Ellenberg values for nitrogen, moisture, acidity and salinity, were included in respectively 580, 642, 504 and 303 models with both a linear and a quadratic term.

In deriving the response models, an adapted Bayesian Information Criterion (BIC) was used, allowing for significant variables in the model equations only. Furthermore, the stepwise selection procedure was started from a so-called empty model, the first step in the selection procedure being the addition of a variable resulting in the biggest, significant increase in de deviance explained. Bakkenes et al. (2002) show that this model term selection procedure results in models with a relatively low number of model terms in the regression equations. However, the distribution of the number of model terms for each model (ranging from 2 to 51, see also Figure 29) shows that the method applied for MOVE4 did lead to models with an ample number of model terms, thus allowing for an ecological interpretation of the model results.

The predicted POO can be interpreted as 0 (absent) or 1 (present) by using the critical POO which is found by maximizing the Kappa statistic. The Kappa method is widely used (and discussed) for decision making in problems of binary value or categorical representations, the so-called 2x2 tables or contingency tables, both in environmental sciences and health sciences (Fielding and Bell, 1997; Hirzel *et al.*, 2006; Couto, 2003; Sim and Wright, 2005; Warrens, 2010; Hoehler, 2000; Van Wieringen and Van den Heuvel, 2005). Its use in determining an optimal or critical PPO value is however rather new and needs further research. An alternative could be the use of so-called ROC curves, although this would also imply finding a critical value solely determined by some statistic. The absence/presence results of MOVE4 are used by the plant dispersal model DIMO. However, since DIMO is a stochastic model, it might as well use the POO itself instead of absence/presence data. The confidence intervals around the calculated POO, as produced by MOVE4, may be taken into account in future versions of DIMO.For the interpretation of POO values in terms of a species being present under certain environmental circumstances, a back-translation to presence/absence data is still needed.

Validation of logistic regression models is notably difficult especially for absence/presence, i.e. 0/1, data (e.g. Pearce & Ferrier 2000, Austin 2007, Mouton *et al.* 2010). In this report a number of validation statistics are tried. The Hosmer-Lemeshow test was dismissed as the sole test for goodness-of-fit as the 560 models that were poor according to this test included models that had high percentages of deviance explained, both for species with high, low and intermediate number of presences, and showed good fits to observed frequencies when proportions of presence were plotted against Ellenberg f, r, n and s. Furthermore, the total validation score, albeit biased by including correlated indicators, showed no clear relation with the Hosmer-Lemeshow test statistic (Figure 26). This validation exercise was done by re-substitution, i.e. the same data were used to fit and to validate the models. Generally re-substitution result in too optimistic conclusion with respect to the performance of a model. It is better to validate the fitted model on an independent dataset, such as in Wamelink et al. (2005), or to use cross validation. This is especially important when the dataset is small as compared to the number of possible variables. As such MOVE4 does not seem to need cross validation as the size of the dataset is huge. However for absence/presence data the relevant dataset size is the number of absences (if small) or the number of presences (if small). The minimum number of presences is set to 50 which is of the same order of magnitude as the number of possible variables. For rare species automatic selection procedures of variables are vulnerable because variables can be selected by pure chance and not because they have a real relationship with the response. So cross validation is especially important for the more rare species.

In retrospection, the validation measures used are not flawless. The weighting of equal proportional differences more heavily with common species can be discussed. The correlation between observed and predicted counts across the clusters falsely assumes a homogeneous variance. Moreover for rare species many clusters will have very low counts and this might have a large impact on the correlation. Because of these difficulties no final conclusions can be drawn with respect to validation of the models. However, considering all validation measures, predicted POO seems to be a better representation of the observed data than the predicted presence. This can be expected as the Kappa statistic forms the link to the original presence/absence data and therefore imposes an additional constraint.

It may be clear that validating is indeed difficult and the used validation measures are merely to be regarded as a first effort and not as the final option. In future modelling and validation more attention has to be paid on this topic.

The tests carried out only reflect the predicted probability of occurrence of one species only. However MOVE4 is often used to predict the changes in the probability of occurrence of species assemblages as a result of changes in abiotic circumstances. This is not yet tested or validated. It may be clear that such validation is highly desirable. As a starting point new collected data can be used. Since MOVE4 is only calibrated on data from before 1990, the model could be validated on data collected after 1990.

In total one may conclude that the model performance, as far as tested is adequate (but not more than that) and can be used to model plant species occurrence. Several possible causes are discussed here. The biggest gain of MOVE4 compared to the earlier version is most likely the better statistical foundation, the exclusion of the effect of heavy metals, the splitting of the multi-value categorical variables for vegetation type and geographical region, and the calculation of the model uncertainty. It is surprising that despite the better model foundation, statistical basis and transparency the differences with the earlier version of MOVE are not larger in terms of the average percentage deviance explained, although a significant higher number of models reach higher maximum predicted POO. Apparently, the current method of logistic regression is not able to find much more resolution within the current dataset and the current definition of the environmental variables. Additional investigation into the performance of the species models might reveal ways in which model performance can be improved.

A definite improvement of MOVE4 compared with the 3.2 version is that the geographical regions (FGR) and the vegetation type (veg) effects are now modelled as separate binary variables. This makes the model results simpler and more interpretable and avoids spurious results for regions or types that are not relevant to the spatial distribution of the species.

A factor that might influence the performance of the models is the exclusion of appropriate vegetation types for some species. As a number of the species studied can be characterised as purely aquatic, such as species of the genera *Najas*, *Potamogeton* and *Utricularia* that may be restricted to specific water conditions (Wortelboer, 2010) the addition of water types may improve the models for these species considerably. Furthermore, special care should be taken when calculating the effects of changes in the environment variables on the distribution of these species as the environmental variables should be estimated in a different way for water plants (i.e. nutrient concentrations in the water entering the site from upstream) than for terrestrial plants (i.e. nutrient availability in the soil derived from local nitrogen deposition).

At the moment Ellenberg values are used in MOVE4 as an indication of the abiotic soil circumstances. The model P2E is needed to convert nitrogen availability, pH and moisture from SMART2-SUMO2 to Ellenberg values for MOVE4. The unexplained system variation (USV) of P2E contributes much to the uncertainty of the model chain of 'The Nature planner' (Schouwenberg *et al.,* 2000). This source of uncertainty could be removed from the model chain when SMART2 and MOVE use the same variables. This could be achieved by replacing Ellenberg indicator values by indicator values based on measurements as described by Wamelink *et al.* (2005). Since the model is normally run in combination with SMART2 (-

SUMO2), this may influence the performance of the whole model chain tremendously. This would also most likely improve the results of the validation of MOVE4, when carried out.

Plants compete with each other for nutrients, light and moisture. Management influences the availability of these factors. Mowing and grazing decrease the amount of nutrients in the system. Moreover, by removal of the above ground biomass the light conditions change. Furthermore, management has a direct influence on plant species occurrence. The growth form and the regeneration ability of a species determine the sensitivity for management damage (Oosterbeek *et al.*, 1997). These relations indicate the importance of using management variables in MOVE4. The variables for vegetation structure (*veg*1 to *veg*5) give a global indication of the management. Direct and indirect effects of management are expressed, as well as effects of other factors like natural succession. The incorporation of a more direct measure of the influence of management on the species occurrence in MOVE4 may be advisable, which also may improve model performance. When the vegetation term in the regression model would be split up in more vegetation types, as suggested above, this may also overcome partly the effect of many management influences. However, this may not fully cover for the management intensity, which also influences the probability of occurrence.

The sometimes rather poor performance of MOVE4, and in general a model based on presence/absence data of plants, may be due to the fact that the presence of a seed bank and events of dispersal are neglected. MOVE4 only predicts the chance that a species will occur based on environmental variables. Whether a species will be present in reality (and also in the data set used for model building or in an independent data set) depends for an important part on seed bank and seed dispersal effects. Although the abiotic circumstances may be suitable for a certain species on a certain location, it may still not be present in reality when the dispersal capacity of the species is insufficient to reach that location with suitable habitat or when the species cannot re-emerge from the seed bank. This will give a poor result when the model is validated. To overcome this problem the dispersal model DIMO was developed. It is advisable not to validate MOVE4 without also applying DIMO to correct for the effects mentioned above, depending on the application of MOVE4. There are also applications of MOVE4 where the combination of DIMO is not important and for those applications MOVE4 can be validated on its own.

The correlation between *n* (nutrient availability) and *r* (acidity) in the data set was high (r^2 = 0.78). Also from other field data and experiments it is well known that there is a correlation between *n* and *r*, but normally not strong enough to justify the use of only one of the variables in the kind of regression analyses as is used here. Another reason to use both *n* and *r* is that in many cases it is desirable to be able to discuss the effects of *n* (nutrients, eutrophication) as well as the effects of *r* (pH). One should be very careful in drawing conclusions, though, as part of the variability in model results is now attributed to differences in nutrient availability but may be caused by differences in acidity, and vice versa.

In 2004, Berendse and Wolff carried out a scientific audit on MOVE3.2. They recommended testing the suitability of the models to predict the future distribution of plant species. Their proposal was to derive response models from data from the years prior to the last 10 years and to use these models to predict probabilities of occurrence for the last 10 years. The predictions can then be compared with available data. Thus, a true validation of the model would be possible. This test has not been carried out yet

References

Alkemade J.R.M., J. Wiertz en J.B. Latour, 1996. Kalibratie van Ellenbergs milieu-indicatiegetallen aan werkelijk gemeten bodemfactoren. Rapport nr. 711901016. RIVM, Bilthoven.

Austin, M., 2007. Species distribution models and ecological theory: A critical assessment and some possible new approaches. ecological modelling 200: 1–19.

Bakkenes, M., D. de Zwart and J.R.M. Alkemade, 2002. MOVE nationaal Model voor de Vegetatie versie 3.2: Achtergronden en analyse van modelvarianten. RIVM rapport 408657006, RIVM, Bilthoven. Including appendices in report 408657010 by the same authors.

Bio, A.M.F., 2000. Does vegetation suit our models? Data and model assumptions and the assessment of species distribution in space. PhD Thesis, Utrecht University, Utrecht.

Couto, P., 2003. Assessing the accuracy of spatial simulation models. Ecologicl Modelling 167, 181-198.

De Heer, M., R. Alkemade, M. Bakkenes, M. van Esbroek, A. Hinsberg, D. de Zwart, 2000. MOVE, nationaal model voor de vegetatie, versie 3. De kans op voorkomen van ca. 900 plantensoorten als functie van 7 omgevingsvariabelen. RIVM rapport 408657002, RIVM, Bilthoven.

Dobben, H.F. van, 1993. Vegetation as a monitor for deposition of nitrogen and acidity. Proefschrift RUU, Utrecht.

Dobben, H.F. van and W. de Vries, 2001. Relatie tussen vegetatie en abiotische factoren in het meetnet vitaliteit en verdroging; een statistische studie op grond van waarnemingen in 200 opstanden in 1995 en 1996. Alterra-rapport 406, Alterra, Wageningen.

Ellenberg, H., H.E. Weber, R. Dull, V. Wirth, W. Werner, and D. Paulissen, 1991. Zeigerwerte von Pflanzen in Mitteleuropa. Scripta Geobot 18:1–248.

Ertsen, A.C.D., J.R.M. Alkemade and M. Wassen, 1998. Calibrating Ellenberg indicator values for moisture, acidity, nutrient availability and salinity in the Netherlands. Plant Ecology 135: 113-124.

Ertsen, A. C. D. and F. G. Wortelboer, 2002. Ristori 2001: Responsmodellen voor aquatische systemen s' Hertogenbosch, The Netherlands:. Royal Haskoning. Report 38931/R0325/DE/DenB.

Fielding, A.H. and J.F. Bell, 1997. A review of methods for the assessment of prediction errors in conservation presence/absence models. Environmental Conservation 24, 38-49.

Heer, M. de, R. Alkemade, M. Bakkenes, M. van Esbroek, A. Hinsberg, D. de Zwart, 2000. MOVE, nationaal model voor de vegetatie, versie 3. De kans op voorkomen van ca. 900 plantensoorten als functie van 7 omgevingsvariabelen. RIVM rapport 408657002, RIVM, Bilthoven.

Heikkinen, R., 1996. Predicting patterns of vascular plant species richness with composite variables: a meso-scale study in Finnish Lapland. Vegetatio 126:151–165.

Hirzel, A.H., Le Lay, G., Helfer, V., Randin, C. and Guisan, A., 2006. Evaluating the ability of habitat suitability models to predict species presences. Ecological Modelling 199, 142-152.

Hoehler, F.K., 2000. Bias and prevalence effects on kappa viewed in terms of sensitivity and specificity. Journal of Epidemiology 53, 499-503.

Hoek, D.C.J. van der, M. Bakkenes and J.R.M. Alkemade, 2000. Natuurwaardering in de NATUURPLANNER. Toepassing voor de VIJNO. RIVM rapport 408657004, RIVM, Bilthoven.

Hoek, D.C.J. van der, M. Bakkenes, 2007. Natuurplanner 3.0: Beschrijving en handleiding. MNP Rapport 500067002/2007, Milieu- en Natuurplanbureau, Bilthoven.

Hosmer, D. W. and S. Lemeshow, 1989. Applied Logistic Regression. John Wiley & Sons, Inc., New York.

Kalton, G. and D. Kasprzyk, 1982. "Imputing for missing survey responses", American Statistical Association, Proceedings of the Section on Survey Research Methods, pp.22-31.

Kros, J., 2002. Evaluation of biogeochemical models at local and regional scale. PhD thesis. Alterra, Wageningen.

Landis, J.R. and G.G. Koch, 1977. The Measurement of Observer Agreement for Categorical Data. Biometrics 33(1):159-174.

Laws, E.A. and J.W. Archie, 1981. Appropriate use of regression analysis in marine biology. Marine Biology 65(1): 13-16.

McCullagh, P. and J.A. Nelder, 1989. Generalized Linear Models, Second Edition. Chapman & Hall, London.

MNP, 2006. Natuurbalans 2006. MNP Rapport 500402001, MNP, Bilthoven.

MNP, 2007. Natuurbalans 2007. MNP Rapport 500402005, MNP, Bilthoven.

Mouton, A.M., B. De Baets, P.L.M. Goethals. 2010. Ecological relevance of performance criteria for species distribution models. Ecological Modelling 221: 1995–2002.

Oosterbeek, J.G.B., J.R.M. ALkemade, J. Wiertz, H.F. van Dobben and G.W.W. Wamelink, 1997. Het modelleren van de effecten van natuurbeheer ten behoeve van MOVE. RIVM rapport 715001006. RIVM, Bilthoven.

Oude Voshaar, J.H., 1995. Oude Voshaar, Statistiek voor Onderzoekers, Wageningen Pers, Wageningen.

Pastoors M.J.H., 1992. Landelijk grondwater model; conceptuele modelbeschrijving. RIVM rapport 714305004, RIVM, Bilthoven.

PBL, 2008. Natuurbalans 2008. PBL Rapport 500402008, PBL, Bilthoven.

Pearce, J. and S. Ferrier 2000. Evaluating the predictive performance of habitat models developed using logistic regression. Ecological Modelling 133; 225–245

Rijt, van de C.W.C.J., L. Hazelhoff, and C.W.P.M. Blom, 1996. Vegetation zonation in a former tidal area: A vegetation-type response model based on DCA and logistic regression using GIS. Journal of Vegetation Scienc. 7: 505–518.

Runhaar, J., J.P.M. Witte and R.H.G. Jongman, 1994. Ellenberg-indicatiewaarden: verbetering met reciprocal averaging? Landschap 11(1):41–47.

Runhaar, J., J.R.M. Alkemade, S.M. Hennekens, J. Wiertz en M. van 't Zelfde, 2002. Afstemming biotische responsmodules DEMNAT-SMART/MOVE. Rapport 408657008, RIVM, Bilthoven.

Schaminée, J.H.J., A.H.F. Stortelder, and V. Westhoff, 1995. De Vegetatie van Nederland. 1. Inleiding tot de plantensociologie: grondslagen, methoden en toepassingen. Opulus Press, Uppsala, SE.

Schouwenberg, E.P.A.G., H. Houweling, M.J.W. Jansen, J. Kros and J.P. Mol-Dijkstra, 2000. Uncertainty propagation in model chains: a case study in nature conservancy. Alterra rapport 1. Alterra, Wageningen.

Sim, J. and Wright, C.C., 2005. The kappa statistic in reliability studies: use, interpretation, and sample size requirements. Physical Therapy 85, 257-268.

Svanholm, H., H. Starklint, H. J. G. Gundersen, J. Fabricius, H. Barlebo, S. Olsen 1989. Reproducibility of histomorphologic diagnoses with special reference to the kappa statistic. APMIS 97: 689–698.

Swartzman, G., C.H. Huang and S. Kaluzny, 1992. Spatial analysis of Berring Sea groundfish survey data using generalised additive-models, Can. J. Fish. Aquat. Sci. 49:1366–1378.

Verboom, J., R. Foppen, P. Chardon, P. Opdam and P. Luttikhuizen, 2001. Introducing the key patch approach for habitat networks with persistent populations: an example for marshland birds. Biological conservation 100:89-101.

Wamelink, G.W.W. and H.F. van Dobben, 2003. Validity and uncertainty of Ellenberg indicator values. Basic and applied ecology, 4:515-523.

Wamelink, G.W.W, P.W.Goedhart, , H.F van Dobben, and F.Berendse, 2005. Plant species as predictors of soil pH: replacing expert judgement by measurements. Journal of vegetation science 16:461-470.

Wamelink, G.W.W., 2007. Simulation of vegetation dynamics as affected by nitrogen deposition. PhD thesis. Alterra, Wageningen.

Wamelink, G.W.W., R.M. Winkler, and F.G. Wortelboer, 2009. User documentation MOVE4 v1.0. Wageningen, Alterra. In prep.

Warrens, M.J., 2010. Inequalities between kappa and kappa-like statistics for kxk tables. Psychometrika, 75, 176-185.

Wiertz, J., J. van Dijk, and J.B. Latour, 1992. De MOVE vegetatiemodule: De kans op voorkomen van 700 plantensoorten als functie van vocht, pH, nutriënten en zout. Rapport 92/24. IBN-DLO, Wageningen, NL.

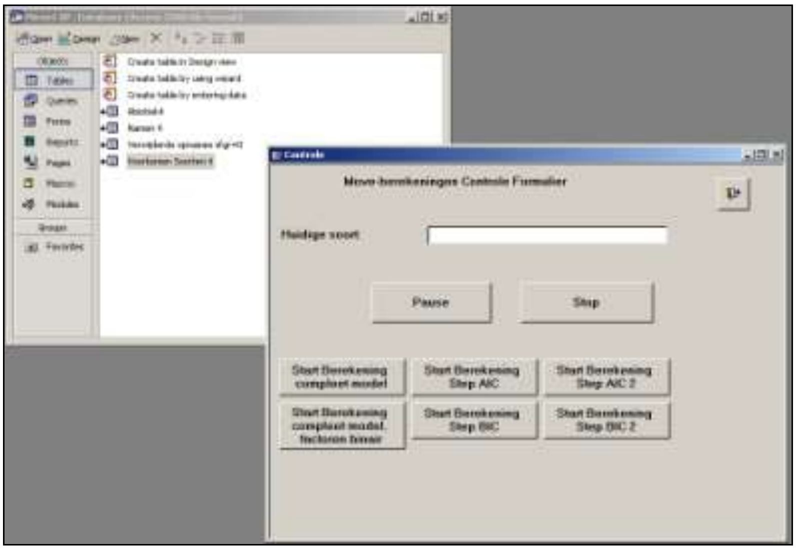
Wortelboer, R., Rosenboom, R., Kragt, F., Ligtvoet, W., Van Gaalen, F., Knoop, J., Cleij, P., Van Puijenbroek, P., Janse, J. en Alkemade, R., 2010. Ecologische effectberekeningen voor de 2e nationale Natuurverkenning. Aquatische systemen. Planbureau voor de Leefomgeving, Bilthoven. Internal report. 147pp.

Yee, T.W. and N.D. Mitchell, 1991. Generalized additive models in plant ecology. Journal of Vegetation Science 2:587-602.

Appendix 1 Procedure voor Berekenen van responsiemodellen (Access + SPlus)

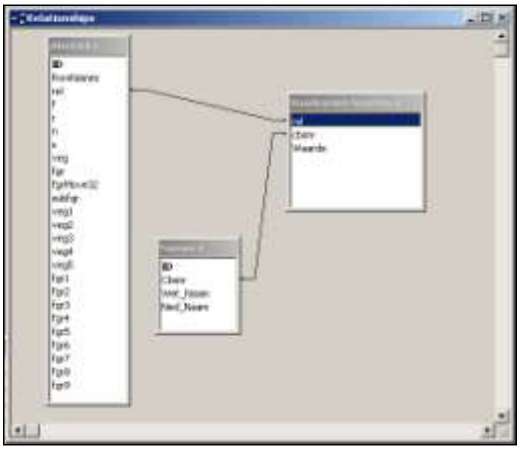
Vanuit een Access-database wordt de procedure voor het bepalen van de responsiemodellen aangestuurd. Hiervoor worden 3 (gelinkte) tabellen gebruikt (zie Figuur 1.1):

1. Namen 4; bevat de soortnamen van een voor-selectie van de planten (914 records). Velden: Id, cbsnr, Wet_naam, Ned_naam.
2. Abiotiek4; bevat de omgevingskenmerken van de opnames (108826 records). Velden: o.a. id, rel, *f*, *r*, *n*, en *s*.
3. Voorkomen Soorten 4; bevat de relatie tussen de opnames en het voorkomen van de planten (allen aanwezigheid; 1779759 records). Velden: rel, cbsnr, Waarde.



Figuur 1.1. Tabellen en controle-formulier voor berekenen van responsiemodellen in Move 4

De tabellen worden gekoppeld middels de velden rel (relevé-nummer of opnamenummer) en cbsnr (uniek nummer toegekend aan de planten door het CBS). Zie Figuur 1.2.



Figuur 1.2. Relaties tabellen Move 4

De selectie welke planten doorgerekend moeten worden, wordt bepaald in de Query ‘Selecteer soorten voor berekeningen’.

Het Controle-formulier geeft toegang tot het gebruiken van verschillende methoden voor het berekenen van de responsiemodellen. Tijdens de berekeningen wordt getoond voor welke soort de berekening wordt uitgevoerd. SPlus gebruikt nog al wat rekentijd en geheugen bij het berekenen van de MOVE-regressies. De Pause/Continue knop maakt het gemakkelijk om tussendoor de berekeningen tijdelijk te onderbreken. De Stop-knop stopt de berekeningen helemaal. Via de Query ‘Selecteer soorten voor berekeningen’ kan gemakkelijk een andere soort-set doorgerekend worden.

Er wordt gebruik gemaakt van 2 Splus-scripts om zo de uitvoer-files te initialiseren. Het format van de uitvoer wordt mede bepaald door het type van het ‘model’waar de statistische berekeningen mee uitgevoerd worden (zie ‘ModelType’ in listing 1.3). ‘Variable’ wordt gebruikt bij modellen die voor elke soort anders kunnen zijn en waarbij dus het model per soort een verschillend aantal termen kan bevatten (b.v. bij stapsgewijze regressie). ‘Fixed’ wordt gebruikt als alle modellen eenzelfde vaste set van termen bevatten (gewonde regressie). Voor het achteraf kunnen interpreteren van de resultaten is dit onderscheid essentieel. Er wordt ook bij het inlezen van de SPlus-resultaten gebruik van gemaakt.

De abiotiek-gegevens bevinden zich in een SPlus Dataframe ‘Abiotiek4’. Het voorkomen van de soorten wordt in de Access-procedure naar een tijdelijke file weggeschreven en in het SPlus-script ingelezen.

De uitvoerbestanden van de SPlus-berekeningen worden gespecificeerd in het SPlus-script, maar ook in de aanstuur-routine in Access.
De uitvoer van de SPlus-berekeningen bestaat uit 6 bestanden:

1. Bestand met de resultaten van de regressie-berekening: aantal waarnemingen, significantie van de regressie, percentage verklaarde variantie, afbreekcriterium, enz. Als naam voor dit bestand wordt de naam van het model gebruikt (b.v. Move4stepBIC2.txt).
2. Bestand met de waarden, standard error en t-waarde van de afzonderlijke coefficienten in de modellen. Naam: b.v. Move4stepBIC2_Coef.txt. Fixed model: een keer alle coefficiënten (namen van de modeltermen) verder alleen de coëfficiënt-waarden. Variable model: bij elk model (elke soort) het aantal coefficienten en de namen van de coefficienten (in specifieke volgorde), gevolgd door de coefficient-waarden.
3. Bestand met de correlaties tussen de termen van het model. Naam: b.v. Move4stepBIC2_Cor.txt. Volgorde van de getallen correspondeert met de volgorde van de coefficienten in het model (zie bij 2).
4. Bestand met de covarianties van de termen in het model. Naam: b.v. Move4stepBIC2_Cov.txt. Volgorde van de getallen correspondeert met de volgorde van de coefficienten in het model (zie bij 2).
5. Bestand met de kans op voorkomen volgens het berekende model voor alle soorten en alle opnames. Naam: b.v. Move4stepBIC2_pred_fit.txt.
6. Bestand met de standard errors van de berekende kans op voorkomen volgens de berekende modellen voor alle soorten en all opnames. Naam: b.v. Move4stepBIC2_pred_se.txt.

De bestanden met de berekende fits en de berekende standard errors kunnen bij een groot aantal opnames erg groot worden. In dit geval is het raadzaam de schrijf-commando’s in het Splus-script uit te commentariëren.

Listing 1.1. Start-script voor berekeningen in Splus. Hier wordt de Splus-functie Move4stepBIC2 aangeroepen met parameter begin=1.

```
# Splus script voor MOVE-regressies
# Michel Bakkenes, juli/augustus 1999
# Aangepast Rick Wortelboer, oktober 2004
       options(width = 100000)
       options(object.size = 50000000000)
       #options(keep = NULL)
       print(c("Reading inv.dat"))
       #rm(planttemp)
       planttemp <- scan("D:\\temp\\inv.dat")
       dx <- planttemp
       print(length(planttemp))

       # Model dat gebruikt wordt om verbetering tov MOVE2 te analyseren
       # (zelfde invoer variabelen als MOVE2 model (f, r en n)
       # analyseren van de bimodalen modellen voor model a (basis model)
       begin <- 1
       Move4stepBIC2(planttemp,begin)


       print("Ready")

       q()
```

Listing 1.2. Doorstart script voor berekeningen in Splus. Hier wordt de Splus-functie Move4stepBIC2 aangeroepen met parameter begin=0.

```
# Splus script voor MOVE-regressies
# Michel Bakkenes, juli/augustus 1999
# Aangepast Rick Wortelboer, oktober 2004

       options(width = 100000)
       options(object.size = 50000000000)
       #options(keep = NULL)
       print(c("Reading inv.dat"))
       #rm(planttemp)
       planttemp <- scan("D:\\temp\\inv.dat")
       dx <- planttemp
       print(length(planttemp))

       # Model dat gebruikt wordt om verbetering tov MOVE2 te analyseren
       # (zelfde invoer variabelen als MOVE2 model (f, r en n)
       # analyseren van de bimodalen modellen voor model a (basis model)
       begin <- 0
       Move4stepBIC2(planttemp,begin)

       print("Ready")

       q()
```

Listing 1.3. Splus-script voor het berekenen van Move-responsiemodellen van hogere planten m.b.v. stapsgewijze logistische regressie.

```
function(data, begin)
{
#  Move4 stepBIC model 2
#    uitgebreid met kwadratische Ellenberg-s, afzonderlijke vegetatietypen en
afzonderlijke fgr's
#  interacties tussen vegetatietypen en ellenberg-waarden en fgr's en ellenberg-
waarden
#  Rick Wortelboer, RIVM/MNP/NLB
#  Versie 23-3-2004
#  - KovOpt berekend
#  aanpassen van namen uitvoer-bestanden
#
       OutFileName <- "Move4stepBIC2.txt"
```

```
        OutFileNamePredFit <- "Move4stepBIC2_pred_fit.txt"
        OutFileNamePredSE <- "Move4stepBIC2_pred_se.txt"
        OutFileNameCoef <- "Move4stepBIC2_Coef.txt"
        OutFileNameCor <- "Move4stepBIC2_Cor.txt"
        OutFileNameCov <- "Move4stepBIC2_Cov.txt"
        ModelType <- "Variable"
        assign("elbr", Abiotiek4$r, where = 0)
        assign("elbn", Abiotiek4$n, where = 0)
        assign("elbf", Abiotiek4$f, where = 0)
        assign("elbs", Abiotiek4$s, where = 0)
        assign("veg", Abiotiek4$veg, where = 0)
        assign("veg1", Abiotiek4$veg1, where = 0)
        assign("veg2", Abiotiek4$veg2, where = 0)
        assign("veg3", Abiotiek4$veg3, where = 0)
        assign("veg4", Abiotiek4$veg4, where = 0)
        assign("veg5", Abiotiek4$veg5, where = 0)
        assign("fgr", Abiotiek4$fgr, where = 0)
        assign("fgr1", Abiotiek4$fgr1, where = 0)
        assign("fgr2", Abiotiek4$fgr2, where = 0)
        assign("fgr3", Abiotiek4$fgr3, where = 0)
        assign("fgr4", Abiotiek4$fgr4, where = 0)
        assign("fgr5", Abiotiek4$fgr5, where = 0)
        assign("fgr6", Abiotiek4$fgr6, where = 0)
        assign("fgr7", Abiotiek4$fgr7, where = 0)
        assign("fgr8", Abiotiek4$fgr8, where = 0)
        assign("fgr9", Abiotiek4$fgr9, where = 0)
        dum <- !is.na(elbr) & !is.na(elbn) & !is.na(elbf) & !is.na(elbs
               ) & !is.na(fgr1) & !is.na(fgr2) & !is.na(fgr3) & !is.na(
               fgr4) & !is.na(fgr5) & !is.na(fgr6) & !is.na(fgr7) & !
               is.na(fgr8) & !is.na(fgr9) & !is.na(veg1) & !is.na(veg2
               ) & !is.na(veg3) & !is.na(veg5) & !is.na(veg5)
        assign("Orgs", planttemp, where = 0)
        OrgVol <- Orgs[dum]    #              assign("OrgVol", OrgVol, where = 0)
#aantal <- sum(OrgVol)
        print(c("lengte Orgvol=", length(OrgVol)))
        print(c("Som Orgvol=", sum(OrgVol)))  #if(aantal > 2) {
        assign("OrgVol", OrgVol, where = 0)
        #       f0  <- glm(OrgVol ~ 1, family = binomial, maxit = 30, na.action =
na.omit)       #
        print(c("Calculating volledig model (abiotiek2) glm's"))
        d1 <- data[dum]
        f0 <- glm(OrgVol ~ 1, family = binomial, maxit = 30, na.action
                = na.omit)
        f1 <- stepBIC.glm(f0,  ~ elbr + elbr^2 + elbn + elbn^2 + elbf +
               elbf^2 + elbs + elbs^2 + fgr1 + fgr2 + fgr3 + fgr4 +
               fgr5 + fgr6 + fgr7 + fgr8 + fgr9 + veg1 + veg2 + veg3 +
               veg4 + veg5 + elbr * elbn + elbr * elbf + elbn * elbf +
               veg1 * elbr + veg1 * elbr^2 + veg2 * elbr + veg2 * elbr^
               2 + veg3 * elbr + veg3 * elbr^2 + veg4 * elbr + veg4 *
               elbr^2 + veg5 * elbr + veg5 * elbr^2 + veg1 * elbn +
               veg1 * elbn^2 + veg2 * elbn + veg2 * elbn^2 + veg3 *
               elbn + veg3 * elbn^2 + veg4 * elbn + veg4 * elbn^2 +
               veg5 * elbn + veg5 * elbn^2 + veg1 * elbf + veg1 * elbf^
               2 + veg2 * elbf + veg2 * elbf^2 + veg3 * elbf + veg3 *
               elbf^2 + veg4 * elbf + veg4 * elbf^2 + veg5 * elbf +
               veg5 * elbf^2 + veg1 * elbs + veg1 * elbs^2 + veg2 *
               elbs + veg2 * elbs^2 + veg3 * elbs + veg3 * elbs^2 +
               veg4 * elbs + veg4 * elbs^2 + veg5 * elbs + veg5 * elbs^
               2 + fgr1 * elbr + fgr1 * elbr^2 + fgr2 * elbr + fgr2 *
               elbr^2 + fgr3 * elbr + fgr3 * elbr^2 + fgr4 * elbr +
               fgr4 * elbr^2 + fgr5 * elbr + fgr5 * elbr^2 + fgr6 *
               elbr + fgr6 * elbr^2 + fgr7 * elbr + fgr7 * elbr^2 +
               fgr8 * elbr + fgr8 * elbr^2 + fgr9 * elbr + fgr9 * elbr^
               2 + fgr1 * elbn + fgr1 * elbn^2 + fgr2 * elbn + fgr2 *
               elbn^2 + fgr3 * elbn + fgr3 * elbn^2 + fgr4 * elbn +
               fgr4 * elbn^2 + fgr5 * elbn + fgr5 * elbn^2 + fgr6 *
               elbn + fgr6 * elbn^2 + fgr7 * elbn + fgr7 * elbn^2 +
               fgr8 * elbn + fgr8 * elbn^2 + fgr9 * elbn + fgr9 * elbn^
               2 + fgr1 * elbf + fgr1 * elbf^2 + fgr2 * elbf + fgr2 *
               elbf^2 + fgr3 * elbf + fgr3 * elbf^2 + fgr4 * elbf +
```

```
                fgr4 * elbf^2 + fgr5 * elbf + fgr5 * elbf^2 + fgr6 *
                elbf + fgr6 * elbf^2 + fgr7 * elbf + fgr7 * elbf^2 +
                fgr8 * elbf + fgr8 * elbf^2 + fgr9 * elbf + fgr9 * elbf^
                2 + fgr1 * elbs + fgr1 * elbs^2 + fgr2 * elbs + fgr2 *
                elbs^2 + fgr3 * elbs + fgr3 * elbs^2 + fgr4 * elbs +
                fgr4 * elbs^2 + fgr5 * elbs + fgr5 * elbs^2 + fgr6 *
                elbs + fgr6 * elbs^2 + fgr7 * elbs + fgr7 * elbs^2 +
                fgr8 * elbs + fgr8 * elbs^2 + fgr9 * elbs + fgr9 * elbs^
                2, direction = "both", trace = F)
        poswrn <- sum(d1)
        totwrn <- length(d1)
        pred <- predict(f1, type = "respons", se = T)
        dev0 <- f0$dev
        dev1 <- f1$dev
        verschil <- dev0 - dev1
        vrgr <- f0$df - f1$df
        dfdev0 <- f0$df
        dfdev1 <- f1$df
        if(vrgr > 0) {
                pdev <- 1 - pchisq(verschil, vrgr)
        }
        else {
                pdev <- 1
        }
        percverkl <- (verschil/dev0) * 100
        aantal <- poswrn
        hos <- hosana3(d1, f1$fit)
        KappaCrit <- 1 #KAPPA-best fit
        KovCrit <- 1
        MaxFit <- max(f1$fit)
        if(length(f1$coefficients) > 1) {
                KappaMax <- 0
                NAantal <- length(d1)
                for(j in 1:100) {
                        K <- MaxFit/100 * j
                        coefa <- sum(d1 * ifelse(f1$fit > K, 1, 0))
                        coefb <-  - sum((d1 - 1) * ifelse(f1$fit > K, 1,
                                0))
                        coefc <- sum(d1 * ifelse(f1$fit < K, 1, 0))
                        coefd <-  - sum((d1 - 1) * ifelse(f1$fit < K, 1,
                                0))
                        KappaVar <- ((coefa + coefc) * (coefa + coefb) + (
                                coefb + coefd) * (coefc + coefd))/
                                NAantal
                        KappaHuidig <- ((coefa + coefd) - KappaVar)/(
                                NAantal - KappaVar)
                        if(KappaMax < KappaHuidig) {
                                KappaMax <- KappaHuidig
                                KovCrit <- K
                        }
                }
                KappaCrit <- KappaMax
        }
        model.summary <- summary(f1)
        write(c(aantal, max(f1$fit), dev0, dfdev0, dev1, dfdev1,
                verschil, dfdev0 - dfdev1, pdev, round(percverkl, 3),
                hos[1], hos[2], KovCrit, KappaCrit), OutFileName,
                ncolumns = 200, append = T)
        if(ModelType == "Fixed") {
                if(begin == 1) {
                        write(c("NumberOfVariables", length(
                                model.summary$coefficients)/3),
                                OutFileNameCoef, ncolumns = 3, append
                                 = T)
                        write(c("NumberOfVariables", length(
                                model.summary$coefficients)/3),
                                OutFileNameCor, ncolumns = 3, append =
                                T)
                        write(c("NumberOfVariables", length(
                                model.summary$coefficients)/3),
```

```
                                OutFileNameCov, ncolumns = 3, append =
                                T)
                }
                for(k in 1:length(f1$coefficients)) {
                        write(names(f1$coefficients[k]),
                                OutFileNameCoef, ncolumns = 200, append
                                 = T)
                        write(names(f1$coefficients[k]), OutFileNameCor,
                                ncolumns = 200, append = T)
                        write(names(f1$coefficients[k]), OutFileNameCov,
                                ncolumns = 200, append = T)
                }
        }
        else {
# ModelType == "Variable"
                print(f1$coefficients)
                write(c("NumberOfVariables", length(model.summary$
                        coefficients)/3), OutFileNameCoef, ncolumns = 3,
                        append = T)
                for(k in 1:length(f1$coefficients)) {
                        write(names(f1$coefficients[k]),
                                OutFileNameCoef, ncolumns = 3, append
                                 = T)
                }
        }
        write(model.summary$coefficients, OutFileNameCoef, ncolumns =
                length(model.summary$coefficients)/3, append = T)
        write(model.summary$correlation, OutFileNameCor, ncolumns =
                length(model.summary$coefficients)/3, append = T)
        write(model.summary$cov.unscaled, OutFileNameCov, ncolumns =
                length(model.summary$coefficients)/3, append = T)
}
```

Listing 1.3b. SPlus-script met de functies stepBIC, addBIC en dropBIC

SPlus Function stepBIC_glm:

```
function(object, scope, scale, direction = c("both", "backward",
        "forward"), trace = T, keep = NULL, steps = 1000, control =
        NULL, ...)
{
        if(missing(direction))
                direction <- "both"
        else direction <- match.arg(direction)
        sub.assign <- function(terms, assign)
        {
                a <- attributes(terms)
                tl <- a$term.labels
                if(a$intercept)
                        tl <- c(names(assign)[1], tl)
                asgn <- assign[tl]
                poi <- 0
                for(i in tl) {
                        la <- length(asgn[[i]])
                        asgn[[i]] <- seq(poi + 1, poi + la)
                        poi <- poi + la
                }
                asgn
        }
        re.arrange <- function(keep)
        {
                namr <- names(k1 <- keep[[1]])
                namc <- names(keep)
                nc <- length(keep)
                nr <- length(k1)
                array(unlist(keep, recursive = F), c(nr, nc), list(namr,
                        namc))
        }
        make.step <- function(models, fit, scale, object)
        {
                change <- sapply(models, "[[", "change")
```

```
                rd <- sapply(models, "[[", "deviance")
                dd <- c(NA, diff(rd))
                rdf <- sapply(models, "[[", "df.resid")
                ddf <- c(NA, diff(rdf))
                BIC <- sapply(models, "[[", "BIC")
                heading <- c("Stepwise Model Path \nAnalysis of Deviance Table",
                        "\nInitial Model:", deparse(as.vector(formula(
                        object))), "\nFinal Model:", deparse(as.vector(
                        formula(fit))), "\n")
                aod <- data.frame(Step = change, Df = ddf, Deviance =
                        dd, "Resid. Df" = rdf, "Resid. Dev" = rd, BIC
                         = BIC, check.names = F)
                attr(aod, "heading") <- heading
                attr(aod, "class") <- c("anova", "data.frame")
                fit$anova <- aod
                fit
        }
        backward <- direction == "both" | direction == "backward"
        forward <- direction == "both" | direction == "forward"
        if(missing(scope)) {
                fdrop <- numeric(0)
                fadd <- NULL
        }
        else {
                if(is.list(scope)) {
                        fdrop <- if(!is.null(fdrop <- scope$lower))
                                attr(terms(update.formula(object,
                                fdrop)), "factor") else numeric(0)
                        fadd <- if(!is.null(fadd <- scope$upper)) attr(
                                terms(update.formula(object, fadd)),
                                "factor")
                }
                else {
                        fadd <- if(!is.null(fadd <- scope)) attr(terms(
                                update.formula(object, scope)),
                                "factor")
                        fdrop <- numeric(0)
                }
        }
        if(is.null(fadd)) {
                backward <- T
                forward <- F
        }
        m <- model.frame(object)
        obconts <- object$contrasts
        objectcall <- object$call     #build the big model matrix
        if(forward) {
                add.rhs <- paste(dimnames(fadd)[[2]], collapse = "+")
                add.rhs <- eval(parse(text = paste("~ . +", add.rhs)))
                new.form <- update.formula(object, add.rhs, evaluate =
                        F)
                fc <- objectcall
                Terms <- terms(new.form)
                fc$formula <- Terms
                fobject <- list(call = fc)
                class(fobject) <- class(object)
                m <- model.frame(fobject)
                x <- model.matrix(Terms, m, contrasts = obconts)
        }
        else {
                Terms <- object$terms
                x <- model.matrix(Terms, m, contrasts = obconts)
        }
        Asgn <- attr(x, "assign")     #from glm.fit
        a <- attributes(m)
        y <- model.extract(m, "response")
        w <- model.extract(m, "weights")
        if(is.null(w))
                w <- rep(1, nrow(m))
        offset <- attr(Terms, "offset")
```

```
        if(is.null(offset))
                offset <- 0
        else offset <- m[, offset]
        family <- family(object)
        obj.control <- object$control
        if(is.null(obj.control))
                obj.control <- glm.control()
        if(!is.null(control))
                obj.control[names(control)] <- control
        control <- obj.control
        models <- vector("list", steps)
        if(!is.null(keep)) {
                keep.list <- vector("list", steps)
                nv <- 1
        }
        n <- length(object$fitted)
        if(missing(scale)) {
                famname <- family$family["name"]
                scale <- switch(famname,
                        Poisson = 1,
                        Binomial = 1,
                        deviance(object)/object$df.resid)
        }
        fit <- object
        cf <- attributes(coef(object))        #check if any terms have zero df
        if(cf$singular) {
                TT <- !match(TL <- attr(object$terms, "term.labels"),
                        names(cf$assign), F)
                if(any(TT)) {
                        upd <- eval(parse(text = paste(c(".~.", TL[TT]),
                                collapse = "-")))
                        fit <- update(fit, upd)
                }
        }
        n <- length(fit$fitted)
        # adjust number of observations without missing data
        bBIC <- deviance(fit) + log(n) * (n - fit$df.resid) * scale
        nm <- 1
        Terms <- fit$terms
        if(trace)
                cat("Start:  BIC=", format(round(bBIC, 4)), "\n",
                        deparse(as.vector(formula(fit))), "\n\n")
        models[[nm]] <- list(deviance = deviance(fit), df.resid = fit$
                df.resid, change = "", BIC = bBIC)
        if(!is.null(keep))
                keep.list[[nm]] <- keep(fit, bBIC)
        BIC <- bBIC + 1
        while(bBIC < BIC & steps > 0) {
                steps <- steps - 1
                BIC <- bBIC
                bfit <- fit
                ffac <- attr(Terms, "factor")
                scope <- factor.scope(ffac, list(add = fadd, drop =
                        fdrop))
                aod <- NULL
                change <- NULL
                if(backward && (ndrop <- length(scope$drop))) {
                        aod <- dropBIC.lm(fit, scope$drop, scale)
                        if(trace)
                                print(aod)
                        change <- rep("-", ndrop + 1)
                }
                if(forward && (nadd <- length(scope$add))) {
                        aodf <- addBIC.lm(fit, scope$add, scale, x = x)
                        if(trace)
                                print(aodf)
                        change <- c(change, rep("+", nadd + 1))
                        if(is.null(aod))
                                aod <- aodf
                        else {
```

```
                                ncaod <- dim(aod)[1]
                                aod[seq(ncaod + 1, ncaod + nadd + 1),
                                  ] <- aodf
                        }
                }
                if(is.null(aod))
                        break
                o <- order(aod[, "sign"], aod[, "Cp"])[1]
                if(o[1] == 1)
                        break
                change <- paste(change[o], dimnames(aod)[[1]][o])
                if(trace)
                        print(change)
                Terms <- terms(update(formula(fit), eval(parse(text =
                        paste("~ .", change)))))
                attr(Terms, "formula") <- rebld.formula(Terms)
                asgn <- sub.assign(Terms, Asgn)
                tx <- x[, unlist(Asgn[names(asgn)]), drop = F]
                attr(tx, "assign") <- asgn
                newfit <- glm.fit(tx, y, w, NULL, offset, family,
                        control$maxit, control$epsilon, null.dev = T)
                n <- length(newfit$fitted)
        # adjust number of observations without missing data
                bBIC <- deviance(newfit) + log(n) * (n - newfit$
                        df.resid) * scale
                if(bBIC >= BIC)
                        break
                if(trace)
                        cat("\nStep:  BIC=", format(round(bBIC, 4)),
                                "\n", deparse(as.vector(formula(Terms))
                                ), "\n\n")
                nm <- nm + 1
                models[[nm]] <- list(deviance = deviance(newfit),
                        df.resid = newfit$df.resid, change = change,
                        BIC = bBIC)
                fit <- c(newfit, list(x = tx, terms = Terms, formula =
                        attr(Terms, "formula")))
                oc <- objectcall
                oc$formula <- as.vector(fit$formula)
                fit$call <- oc
                fit$family <- object$family
                class(fit) <- class(object)
                if(!is.null(keep))
                        keep.list[[nm]] <- keep(fit, bBIC)
        }
        if(!is.null(keep))
                fit$keep <- re.arrange(keep.list[seq(nm)])
        make.step(models = models[seq(nm)], fit, scale, object)
}
```

SPlus Function addBIC_lm:

```
function(object, scope = . ~ ., scale, keep, x = NULL)
{
        add.all <- function(qr, effect, x, assgn, tol = .Machine$
                double.eps^0.5)
        {
                dx <- dim(x)
                n <- as.integer(dx[1])
                p <- as.integer(dx[2])
                pold <- as.integer(length(qr$pivot))
                number <- as.integer(length(assgn))
                lngth <- as.integer(sapply(assgn, length))
                start <- as.integer(sapply(assgn, "[[", 1))
                .Fortran("addall",
                        x = x,
                        n,
                        p,
                        start,
                        lngth,
```

```
                        number,
                        effect,
                        effects = matrix(0, n, number),
                        pivot = integer(p),
                        rank = integer(number),
                        chisq = double(number),
                        pold,
                        qr$qr,
                        qr$qraux,
                        qr$pivot,
                        qr$rank,
                        double(3 * p + n),
                        tol)[c("x", "effects", "chisq", "pivot", "rank"
                        )]
        }
        p <- length(object$coef)
        if(!is.character(scope))
                scope <- add.scope(object, update.formula(object, scope,
                        evaluate = F))
        if(!length(scope))
                stop("no terms in scope for adding to object")
        if(is.null(x)) {
# when called iteratively x can be known
# need to do the following since the scope might be a character vector of term
labels
                add.rhs <- paste(scope, collapse = "+")
                add.rhs <- eval(parse(text = paste("~ . +", add.rhs)))
                new.form <- update.formula(object, add.rhs, evaluate =
                        F)
                fc <- object$call
                Terms <- terms(new.form)
                fc$formula <- Terms
                fob <- list(call = fc)
                class(fob) <- class(object)
                m <- model.frame(fob)
                x <- model.matrix(Terms, m, contrasts = object$
                        contrasts)
        }
        cnames <- dimnames(x)[[2]]
        iswt <- !is.null(wt <- object$weights)
        if(iswt)
                x <- x * sqrt(wt)
        n <- dim(x)[[1]]
        asgn <- attr(x, "assign")
        tl <- names(asgn)
        if(!all(match(scope, tl, F)))
                stop("scope is not a subset of term labels of the supplied x"
                        )
        xasgn <- unlist(asgn[names(object$assign)])
        asgn <- asgn[scope]
        k <- length(scope)
        rdf <- object$df.resid
        chisq <- deviance.lm(object)
        if(missing(scale))
                scale <- chisq/rdf
        if(!missing(keep)) {
                max.keep <- c("coefficients", "fitted", "residuals",
                        "x.residuals", "effects", "R")
                if(is.logical(keep) && keep)
                        keep <- max.keep
                else {
                        if(!all(match(keep, max.keep, F)))
                                stop(paste(
                                  "Can only keep one or more of: \"",
                                  paste(max.keep, collapse = "\", \""),
                                  "\"", sep = ""))
                }
                fv <- predict(object)
                y <- object$residuals + fv
                if(iswt) {
```

```
                        if(any(wt == 0))
                                stop(
                                  "\"keep\" not allowed when some of the weights are
zero"
                                  )
                        wt <- sqrt(wt)
                }
        }
        else keep <- character(0)
        xr <- match("x.residuals", keep, F)
        value <- array(vector("list", 6 * k), c(k, 6), list(scope, c(
                "coefficients", "fitted", "residuals", "x.residuals",
                "effects", "R")))
        if(length(ef <- object$effects) < n)
                stop("function only currently defined for methods that compute
effects"
                        )
        dfs <- double(k)
        chis <- double(k)
        R <- object$R
        if(length(xasgn))
                oldx <- x[, xasgn, drop = F]
        else stop("need a term or an intercept in initial model")
        qR <- object$qr
        if(is.null(qR)) {
                qR <- qr(oldx)
        }
        if(xr) {
                xresid <- qr.resid(qR, x[, unlist(asgn)])
                if(iswt)
                        xresid <- xresid/wt
        }
        newx <- x[, unlist(asgn), drop = F]
        newnames <- cnames[unlist(asgn)]
        TT <- rep(F, (length(cnames)))
        TT[unlist(asgn)] <- T
        asgn <- assign.sub(asgn, TT)
        addall <- add.all(qR, ef, newx, asgn)
        chis <- addall$chis
        dfs <- addall$rank
        if(length(keep)) {
                pivot <- addall$pivot
                newqr <- addall$x
                effects <- addall$effects
                Rnames <- dimnames(R)[[1]]
                oldrank <- object$rank
                oldns <- !is.na(object$coef)
                oldns <- seq(oldns)[oldns]
                for(i in 1:k) {
                        asgni <- asgn[[i]]
                        ranki <- dfs[i]
                        pivoti <- pivot[asgni]
                        pi <- length(asgni)
                        if(xr) {
                                value[[i, 4]] <- xresid[, asgni]
                        }
                        newrank <- oldrank + ranki
                        goodcols <- seq(newrank)
                        nR <- newqr[oldrank + seq(pi), asgni, drop = F]
                        nR[lower.tri(nR)] <- 0
                        nR <- rbind(newqr[seq(oldrank), pivoti, drop =
                                F], nR)
                        rnames <- c(Rnames, newnames[pivoti])
                        r <- array(0, c(p + pi, p + pi), list(rnames,
                                rnames))
                        r[seq(p), seq(p)] <- R
                        r[seq(oldrank + pi), seq(pi) + p] <- nR
        #reshuffle R if it was rank defficient
                        if(Tr <- p - oldrank)
                                r <- r[, c(seq(oldrank), seq(pi) + p,
```

```
                                  seq(Tr) + oldrank)]
                        efi <- effects[, i]
                        names(efi) <- c(dimnames(r)[[2]][goodcols], rep(
                                "", n - newrank))
                        attr(r, "rank") <- newrank
                        class(r) <- "upper"
                        bi <- backsolve(r[goodcols, goodcols], efi[
                                goodcols])     #now slot in the NAs
                        if(Tr | (pi - ranki)) {
                                Bi <- rep(NA, p + pi)
                                names(Bi) <- c(cnames[xasgn], newnames[
                                  asgni])
                                if(ranki) {
                                  TT <- (pivoti - asgni[1] + 1)[seq(
                                    ranki)]
                                  Bi[c(oldns, p + TT)] <- bi
                                  fvi <- cbind(oldx[, oldns, drop = F],
                                    newx[, asgni[TT]]) %*% bi
                                }
                                else {
                                  Bi[seq(p)] <- object$coef
                                  fvi <- object$fitted
                                }
                                bi <- Bi
                        }
                        else fvi <- cbind(oldx, newx[, asgni]) %*% bi
                        if(iswt)
                                fvi <- fvi/wt
                        value[i, -4] <- list(bi, fvi, y - fvi, efi, r)
                }
        }
        scope <- c("<none>", scope)
        dfs <- c(0, dfs)
        chis <- c(chisq, chis)
        bics <- chis + log(n) * (n - rdf + dfs) * scale
        dfs[1] <- NA
        pDev <- 1 - pchisq(c(NA, chis[1] - chis[-1]), dfs)
        pDev[1] <- 0
        sign <- ifelse(pDev <= 0.05, 0, 1)
        aod <- data.frame(Df = dfs, "Sum of Sq" = c(NA, chis[1] - chis[
                -1
                ]), pDev = pDev, sign = sign, RSS = chis, Cp = bics,
                row.names = scope, check.names = F)
        head <- c("Single term additions", "\nModel:", deparse(
                as.vector(formula(object))))
        if(!missing(scale))
                head <- c(head, paste("\nscale: ", format(scale), "\n")
                        )
        class(aod) <- c("anova", "data.frame")
        attr(aod, "heading") <- head
        if(length(keep))
                list(anova = aod, keep = structure(value[, keep, drop
                         = F], class = "matrix"))
        else aod
}
```

SPlus Function dropBIC:

```
function(object, scope, scale, keep, all.cols = T)
{
        b <- coef(object)
        cnames <- labels(b)
        singular <- attr(b, "singular")
        p <- length(b)
        x <- model.matrix(object)
        iswt <- !is.null(wt <- object$weights)
        if(iswt) {
                x <- x * sqrt(wt)
        }
        n <- dim(x)[[1]]
```

```
        asgn <- attr(x, "assign")
        tl <- attr(object$terms, "term.labels")
        if(missing(scope))
                scope <- drop.scope(object)
        else {
                if(!is.character(scope))
                        scope <- attr(terms(update.formula(object,
                                scope)), "term.labels")
                if(!all(match(scope, tl, F)))
                        stop("scope is not a subset of term labels")
        }
        asgn <- asgn[scope]
        k <- length(scope)
        rdf <- object$df.resid
        chisq <- deviance.lm(object)
        if(missing(scale))
                scale <- chisq/rdf
        if(!missing(keep)) {
                max.keep <- c("coefficients", "fitted", "residuals",
                        "x.residuals", "effects", "R")
                if(is.logical(keep) && keep)
                        keep <- max.keep
                else {
                        if(!all(match(keep, max.keep, F)))
                                stop(paste(
                                  "Can only keep one or more of: \"",
                                  paste(max.keep, collapse = "\", \""),
                                  "\"", sep = ""))
                }
                fv <- predict(object)
                y <- object$residuals + fv
                if(iswt) {
                        if(any(wt == 0))
                                stop(
                                  "\"keep\" not allowed when some of the weights are
zero"
                                  )
                        wt <- sqrt(wt)
                }
        }
        else keep <- character(0)
        xr <- match("x.residuals", keep, F)
        value <- array(vector("list", 6 * k), c(k, 6), list(scope, c(
                "coefficients", "fitted", "residuals", "x.residuals",
                "effects", "R")))
        if(length(ef <- object$effects) < n)
                stop("function only currently defined for methods that compute
effects"
                        )
        dfs <- double(k)
        chis <- double(k)
        if(singular) {
                y <- (object$residuals + predict(object))
                na.coef <- (1:length(object$coefficients))[!is.na(
                        object$coefficients)]
                if(iswt) {
                        if(!length(keep))
                                wt <- sqrt(wt)
                        y <- y * wt
                }
                rank <- object$rank
                for(i in 1:k) {
                        ii <- asgn[[i]]         #brute force method
                        if(all.cols)
                                jj <- setdiff(seq(ncol(x)), ii)
                        else jj <- setdiff(na.coef, ii)
                        z <- lm.fit.qr(x[, jj, drop = F], y, singular
                                 = T, qr = xr)
                        efi <- z$effects
                        dfs[i] <- rank - (ranki <- z$rank)
```

```
                        chis[i] <- sum(efi[ - seq(ranki)]^2)
                        if(length(keep)) {
                                fvi <- z$fitted
                                res <- z$residuals
                                if(iswt) {
                                  fvi <- fvi/wt
                                  res <- res/wt
                                }
                                value[i, -4] <- list(z$coef, fvi, res,
                                  efi, z$R)
                                if(xr) {
                                  xres <- qr.resid(z$qr, x[, ii, drop
                                     = F])
                                  if(iswt)
                                    xres <- xres/wt
                                  value[[i, 4]] <- xres
                                }
                        }
                }
        }
        else {
                R <- object$R
                R <- array(R, dim(R), dimnames(R))
                if(xr) {
                        xk <- array(0, dim(x), dimnames(x))
                        xk[1:p,  ] <- R
                }
                else xk <- array(ef, c(n, 1), list(dimnames(x)[[1]],
                                NULL))
                for(i in 1:k) {
                        ii <- asgn[[i]]
                        pii <- length(ii)
                        dfs[i] <- pii
                        if(xr) {
                                xi <- xk[, c(1, ii)]
                                xi[, 1] <- ef
                        }
                        else xi <- xk
                        r <- R
                        pi <- 1:(p - pii)
                        pp <-  - p
                        for(j in rev(ii)) {
                                z <- delcol(r, xi, j)
                                r <- z[[1]][pp,  ]
                                xi <- z[[2]]
                                pp <- pp + 1
                        }
                        efi <- xi[, 1]
                        chis[i] <- sum(efi[ - pi]^2)
                        if(length(keep)) {
# compute it all, even though all may not be reqd
                                bi <- as.matrix(backsolve(r, xi[pi,  ])
                                  )
                                dimnames(bi)[[1]] <- cnames[ - ii]
                                fvi <- x[,  - ii, drop = F] %*% bi
                                if(iswt)
                                  fvi <- fvi/wt
                                names(efi)[] <- ""
                                names(efi)[pi] <- cnames[ - ii]
                                value[i, -4] <- list(bi[, 1], fvi[, 1],
                                  y - fvi[, 1], efi, r)
                                if(xr) {
                                  xres <- x[, ii] - fvi[, -1]
                                  if(iswt)
                                    xres <- xres/wt
                                  value[[i, 4]] <- xres
                                }
                        }
                }
        }
```

```
        scope <- c("<none>", scope)
        dfs <- c(0, dfs)
        chis <- c(chisq, chis)
        bics <- chis + log(n) * (n - rdf - dfs) * scale
        dfs[1] <- NA
        pDev <- pchisq(c(NA, chis[-1] - chis[1]), dfs)
        pDev[1] <- 0
        sign <- ifelse(pDev <= 0.05, 0, 1)
        aod <- data.frame(Df = dfs, "Sum of Sq" = c(NA, chis[-1] - chis[
                1]), pDev = pDev, sign = sign, RSS = chis, Cp = bics,
                row.names = scope, check.names = F)
        head <- c("Single term deletions", "\nModel:", deparse(
                as.vector(formula(object))))
        if(!missing(scale))
                head <- c(head, paste("\nscale: ", format(scale), "\n")
                        )
        class(aod) <- c("anova", "data.frame")
        attr(aod, "heading") <- head
        if(length(keep))
                list(anova = aod, keep = structure(value[, keep, drop
                         = F], class = "matrix"))
        else aod
}
```

Listing 1.4. Procedures voor sequentieel berekenen van de responsiemodellen met SPlus

```
Option Compare Database
Option Explicit
Const ModelNaamBasis = "Move4"
Const MethodeFixed = 1
Const MethodeFixedBinair = 11
Const MethodeStepAIC = 2
Const MethodeStepBIC = 3
Const MethodeStepAIC2 = 4
Const MethodeStepBIC2 = 5
Const AbiotiekTabelNaam = "Abiotiek4"
Const TempDirectory = "D:\temp"
Const MoveOpnamesDirectory = "D:\Projekten\Graadmeters & Ecologische
Modellering\Move\Opnames4"
Const ScriptDir = "D:\"
Const OutputDir = "C:\SplusData"
Const SplusProgrammCall = "C:\Program Files\sp2000\cmd\SPLUS.exe"
Const SplusCommandArgs = " S_ELMHOST=131.224.138.58 S_PROJ=C:\splusdata /BATCH "


Function BerekenCompleetModel()
    Bereken ModelNaamBasis & "Compleet3", MethodeFixed, "Selecteer soorten voor
berekeningen"
End Function

Function BerekenCompleetModelBinair()
    Bereken ModelNaamBasis & "Compleet3Bin", MethodeFixedBinair, "Selecteer
soorten voor berekeningen"
End Function

Function BerekenStepAIC()
    Bereken ModelNaamBasis & "stepAIC", MethodeStepAIC, "Selecteer soorten voor
berekeningen"
End Function

Function BerekenStepAIC2()
    Bereken ModelNaamBasis & "stepAIC2", MethodeStepAIC2, "Selecteer soorten voor
berekeningen"
End Function

Function BerekenStepBIC()
    Bereken ModelNaamBasis & "stepBIC", MethodeStepBIC, "Selecteer soorten voor
berekeningen"
End Function
```

```
Function BerekenStepBIC2()
    Bereken ModelNaamBasis & "stepBIC2", MethodeStepBIC2, "Selecteer soorten voor
berekeningen"
End Function


Function Bereken(ModelNaam As String, Methode As Integer, SoortenTabelNaam As
String)
    Dim SoortSet As DAO.Recordset
    Dim sql As String
    Dim filenum As Long
    Dim RelSet As DAO.Recordset
    Dim ModelType As String
    Dim SplusCommandLine As String

    SplusCommandLine = Chr(34) & SplusProgrammCall & Chr(34) & SplusCommandArgs

    Select Case Methode
      Case MethodeFixed: ModelType = "Fixed"
      Case Else:         ModelType = "Variable"
    End Select

    sql = "SELECT Id " & _
          "FROM [" & AbiotiekTabelNaam & "] " & _
          "ORDER BY Id;"
    Set RelSet = CurrentDb.OpenRecordset(sql)

    ' Initialize files
    filenum = FreeFile
    Open OutputDir & "\" & ModelNaam & ".txt" For Output As #filenum
    Print #filenum, "Soort", "Aantal", "pMaxExp", "Dev0", "dfDev0", "Dev1",
"dfDev1", "DeltDev", "dfDeltDev", "pDev", _
      "PercDevVerkl", "ChiHos", "pChiHos", "KovOpt", "kappa"
    Close #filenum
    Open OutputDir & "\" & ModelNaam & "_pred_fit.txt" For Output As #filenum
    Print #filenum, "Soort"; " ";
    While Not RelSet.EOF
      Print #filenum, RelSet!Id & " ";
      RelSet.MoveNext
    Wend
    Print #filenum,
    Close #filenum
    Open OutputDir & "\" & ModelNaam & "_pred_se.txt" For Output As #filenum
    Print #filenum, "Soort"; " ";
    RelSet.MoveFirst
    While Not RelSet.EOF
      Print #filenum, RelSet!Id & " ";
      RelSet.MoveNext
    Wend
    Print #filenum,
    Close #filenum
    RelSet.Close

    Open OutputDir & "\" & ModelNaam & "_Coef.txt" For Output As #filenum
    Print #filenum, "Coefficients"
    Print #filenum, "ModelType" & " " & ModelType
    Close #filenum
    Open OutputDir & "\" & ModelNaam & "_Cor.txt" For Output As #filenum
    Print #filenum, "Correlations"
    Print #filenum, "ModelType" & " " & ModelType
    Close #filenum
    Open OutputDir & "\" & ModelNaam & "_Cov.txt" For Output As #filenum
    Print #filenum, "CovariancesUnscaled"
    Print #filenum, "ModelType" & " " & ModelType
    Close #filenum

    Set SoortSet = CurrentDb.OpenRecordset(SoortenTabelNaam)
    Dim EersteSoort As Integer
    EersteSoort = True
```

```
    SysCmd acSysCmdInitMeter, "Bereken Responsiemodellen: ",
AantalRecords(SoortSet)
    DoPause = False
    DoStop = False
    While Not SoortSet.EOF

      FileCopy MoveOpnamesDirectory & "\srt_" & SoortSet!Cbsnr & ".ou",
TempDirectory & "\inv.dat"

      On Error Resume Next
          Forms("Controle")![txtHuidigeSoort] = SoortSet!Cbsnr
      DoEvents
      On Error GoTo 0

      filenum = FreeFile
      Open OutputDir & "\" & ModelNaam & ".txt" For Append As #filenum
      Print #filenum, SoortSet!Cbsnr & " ";
      Close #filenum
      Open OutputDir & "\" & ModelNaam & "_pred_fit.txt" For Append As #filenum
      Print #filenum, SoortSet!Cbsnr & " ";
      Close #filenum
      Open OutputDir & "\" & ModelNaam & "_pred_se.txt" For Append As #filenum
      Print #filenum, SoortSet!Cbsnr & " ";
      Close #filenum
      Open OutputDir & "\" & ModelNaam & "_Coef.txt" For Append As #filenum
      Print #filenum, SoortSet!Cbsnr
      Close #filenum
      Open OutputDir & "\" & ModelNaam & "_Cor.txt" For Append As #filenum
      Print #filenum, SoortSet!Cbsnr
      Close #filenum
      Open OutputDir & "\" & ModelNaam & "_Cov.txt" For Append As #filenum
      Print #filenum, SoortSet!Cbsnr
      Close #filenum

      If EersteSoort Then
        Select Case Methode
          Case MethodeFixed:
                   RunApplicationChecked SplusCommandLine & " " & ScriptDir &
"\Move4_Compleet3_begin.ssc " & TempDirectory & " \outc3.txt " & TempDirectory & "
\error.txt", "" & TempDirectory & " \outc3.txt", "Ready"
          Case MethodeFixedBinair:
                   RunApplicationChecked SplusCommandLine & " " & ScriptDir &
"\Move4_Compleet3Bin_begin.ssc " & TempDirectory & " \outc3.txt " & TempDirectory
& " \error.txt", "" & TempDirectory & " \outc3.txt", "Ready"
          Case MethodeStepAIC:
                   RunApplicationTimed SplusCommandLine & " " & ScriptDir &
"\Move4_stepAIC_begin.ssc", 300000
          Case MethodeStepBIC:
                   RunApplicationTimed SplusCommandLine & " " & ScriptDir &
"\Move4_stepBIC_begin.ssc", 300000
          Case MethodeStepAIC2:
                   RunApplicationChecked SplusCommandLine & " " & ScriptDir &
"\Move4_stepAIC2_begin.ssc " & TempDirectory & " \outAIC2.txt " & TempDirectory &
" \error.txt", "" & TempDirectory & " \outAIC2.txt", "Ready"
          Case MethodeStepBIC2:
                   RunApplicationChecked SplusCommandLine & " " & ScriptDir &
"\Move4_stepBIC2_begin.ssc " & TempDirectory & " \outBIC2_" & SoortSet!Cbsnr &
".txt " & TempDirectory & " \error.txt", "" & TempDirectory & " \outBIC2_" &
SoortSet!Cbsnr & ".txt", "Ready"
        End Select
      Else
        Select Case Methode
          Case MethodeFixed:
                   RunApplicationChecked SplusCommandLine & " " & ScriptDir &
"\Move4_Compleet3.ssc " & TempDirectory & " \outc3.txt " & TempDirectory & "
\error.txt", "" & TempDirectory & " \outc3.txt", "Ready"
          Case MethodeFixedBinair:
                   RunApplicationChecked SplusCommandLine & " " & ScriptDir &
"\Move4_Compleet3Bin.ssc " & TempDirectory & " \outc3.txt " & TempDirectory & "
\error.txt", "" & TempDirectory & " \outc3.txt", "Ready"
```

```
          Case MethodeStepAIC:
                   RunApplicationTimed SplusCommandLine & " " & ScriptDir &
"\Move4_stepAIC.ssc", 300000
          Case MethodeStepBIC:
                   RunApplicationTimed SplusCommandLine & " " & ScriptDir &
"\Move4_stepBIC.ssc", 300000
          Case MethodeStepAIC2:
                   RunApplicationChecked SplusCommandLine & " " & ScriptDir &
"\Move4_stepAIC2.ssc " & TempDirectory & " \outAIC2.txt " & TempDirectory & "
\error.txt", "" & TempDirectory & " \outAIC2.txt", "Ready"
          Case MethodeStepBIC2:
                   'RunApplicationChecked SplusCommandLine & " " & ScriptDir &
"\Move4_stepBIC2.ssc " & TempDirectory & " \outBIC2.txt " & TempDirectory & "
\error.txt", "" & TempDirectory & " \outBIC2.txt", "Ready"
                   RunApplicationChecked SplusCommandLine & " " & ScriptDir &
"\Move4_stepBIC2.ssc " & TempDirectory & " \outBIC2_" & SoortSet!Cbsnr & ".txt " &
TempDirectory & " \error.txt", "" & TempDirectory & " \outBIC2_" & SoortSet!Cbsnr
& ".txt", "Ready"
        End Select
      End If

      EersteSoort = False
      'SysCmd acSysCmdUpdateMeter, SoortSet.AbsolutePosition + 1
      DoEvents
      Dim i As Integer
      'For i = 1 To 30000
      '  DoEvents
      'Next i

      If DoPause Or DoStop Then
        If DoStop Then
          SoortSet.MoveLast
        Else
          Forms("Controle")![txtHuidigeSoort] = SoortSet!Cbsnr & "; Paused"
          While DoPause And Not DoStop
            DoEvents
          Wend
          If DoStop Then
            SoortSet.MoveLast
          End If
        End If
      End If

      SoortSet.MoveNext

      DoEvents
    Wend
    SoortSet.Close
    SysCmd acSysCmdRemoveMeter

End Function
```

Listing 1.5. Procedure voor starten van applicaties vanuit Access

```
Option Compare Database
Option Explicit

   Private Type STARTUPINFO
      cb As Long
      lpReserved As String
      lpDesktop As String
      lpTitle As String
      dwX As Long
      dwY As Long
      dwXSize As Long
      dwYSize As Long
      dwXCountChars As Long
      dwYCountChars As Long
      dwFillAttribute As Long
      dwFlags As Long
```

```
      wShowWindow As Integer
      cbReserved2 As Integer
      lpReserved2 As Long
      hStdInput As Long
      hStdOutput As Long
      hStdError As Long
   End Type

   Private Type PROCESS_INFORMATION
      hProcess As Long
      hThread As Long
      dwProcessID As Long
      dwThreadID As Long
   End Type

   Private Declare Function WaitForSingleObject Lib "kernel32" (ByVal _
      hHandle As Long, ByVal dwMilliseconds As Long) As Long

   Private Declare Function CreateProcessA Lib "kernel32" (ByVal _
      lpApplicationName As String, ByVal lpCommandLine As String, ByVal _
      lpProcessAttributes As Long, ByVal lpThreadAttributes As Long, _
      ByVal bInheritHandles As Long, ByVal dwCreationFlags As Long, _
      ByVal lpEnvironment As Long, ByVal lpCurrentDirectory As String, _
      lpStartupInfo As STARTUPINFO, lpProcessInformation As _
      PROCESS_INFORMATION) As Long

   Private Declare Function CloseHandle Lib "kernel32" _
      (ByVal hObject As Long) As Long

   Private Declare Function GetExitCodeProcess Lib "kernel32" _
      (ByVal hProcess As Long, lpExitCode As Long) As Long

   Private Const NORMAL_PRIORITY_CLASS = &H20&
   Private Const INFINITE = -1&
   Private Const CheckInterval = 10000

Public Function RunApplicationChecked(CommandLine As String, FileName As String,
CheckString As String)
    ExecCmdChecked CommandLine, FileName, CheckString
End Function

Public Function ExecCmdChecked(cmdline$, CheckFileName As String, CheckString As
String)
      Dim proc As PROCESS_INFORMATION
      Dim start As STARTUPINFO
      Dim ret As Long
      Dim ProcessEnded As Integer, iWaitTime As Long
      Dim Begintime As Long

      ProcessEnded = False
      If Len(Dir(CheckFileName)) > 0 Then
        Begintime = Timer
        On Error Resume Next
        Kill CheckFileName
        While Err > 0 And Len(Dir(CheckFileName)) > 0
          While Timer > Begintime And Timer - Begintime < CheckInterval / 1000
            DoEvents
          Wend
          Kill CheckFileName
        Wend
      End If
      DoEvents
      ' Initialize the STARTUPINFO structure:
      start.cb = Len(start)
      start.wShowWindow = 2

      ' Start the shelled application:
      Shell cmdline, vbMinimizedNoFocus

      While Not ProcessEnded
```

```
        ' wait a while
        Begintime = Timer
        While Timer > Begintime And Timer - Begintime < CheckInterval / 1000
          DoEvents
          DoEvents
        Wend
        ' Check file contents
        ProcessEnded = CheckStringInFile(CheckFileName, CheckString)
      Wend

      Call GetExitCodeProcess(proc.hProcess, ret&)
      Call CloseHandle(proc.hThread)
      Call CloseHandle(proc.hProcess)
      ExecCmdChecked = ret&
End Function

Function CheckStringInFile(FileName As String, CheckString As String) As Integer
    Dim filenum As Integer, str As String, Found As Integer
    filenum = FreeFile
    Found = False
    On Error Resume Next
    Open FileName For Input As filenum
    If Err = 0 Then
      While Not EOF(filenum) And Not Found
        Input #filenum, str
        If InStr(str, CheckString) > 0 Then
          Found = True
        End If
      Wend
      Close #filenum
    End If
    On Error GoTo 0
    CheckStringInFile = Found
End Function

Public Function RunApplicationTimed(CommandLine As String, msTime As Long)
    ExecCmdTimed CommandLine, msTime
End Function

Public Function ExecCmdTimed(cmdline$, msTime As Long)
      Dim proc As PROCESS_INFORMATION
      Dim start As STARTUPINFO
      Dim ret As Long

      ' Initialize the STARTUPINFO structure:
      start.cb = Len(start)

      ' Start the shelled application:
      ret& = CreateProcessA(vbNullString, cmdline$, 0&, 0&, 1&, _
                            NORMAL_PRIORITY_CLASS, 0&, vbNullString, start, proc)

      ' Wait for the shelled application to finish:
      ret& = WaitForSingleObject(proc.hProcess, msTime)

      Call GetExitCodeProcess(proc.hProcess, ret&)
      Call CloseHandle(proc.hThread)
      Call CloseHandle(proc.hProcess)
      ExecCmdTimed = ret&
End Function
```

Appendix 2 Procedure voor inlezen van SPlus-resultaten

Een database is opgezet om de resultaten van SPlus in te lezen en op te slaan, en wel zodanig dat hiermee de resultaten van de regressie-berekeningen eenduidig vastliggen en er vervolgens op een gestandaardiseerde wijze modelberekeningen uitgevoerd kunnen worden.

De procedure VerwerkResultaten leest de SPus-resultaten in in de diverse tabellen. Als argument wordt de naam van een initialisatiebestand meegegeven (Listing 2.1). De bestanden 'Move4stepBIC2_Taxa.txt' en 'Move4stepBIC2_Variabelen.txt' zijn niet door SPlus gegenereerd, maar moeten door de gerbuiker apart aangeleverd worden. Het bestand 'Move4stepBIC2_Taxa.txt' betreft een lijst met TaxonNr (nummer wat verder in de bestanden en berekeningen van Move 4 gebruikt wordt voor het identificeren van de soorten; type: long), Taxon-code (code die SPlus genereert uit de aangeboden taxonid's; type: string) en TaxonNaam (uitgebreide soortnaam). Het bestand 'Move4stepBIC2_Variabelen.txt' betreft een lijst met VariabeleNr (nummer wat verder in de bestanden en berekeningen van Move 4 gebruikt wordt voor het identificeren van de variabelen; type: long), VariabeleCode (naam van de variabele in het SPlus-model; type: string) , VariabeleNaam (lange naam van de variabele; type: string), VariabeleOmschrijving (omschrijving van de variabele; type: string), VariabeleEenheid (eenheid van de variabele zoals aangeboden aan het SPlus-model), VariabeleType (type van de variabele: zontinue, binair, factor), VariabeleTransformatie (benodigde transformatie die is toegepast op de variabele zoals aangeboden aan het SPlus-model; -, log of log+1).

De verwerkte resultaten worden in een aparte database opgeslagen. De afzonderlijke functies staan vermeld in Listing 2.4 - 2.6.

Listing 2.1. Initialisatiebestand

```
[Algemeen]
Model          = Move Terrestrisch Versie 4.0 BIC
Opmerkingen    = Berekend door Rick Wortelboer Mei-Juni 2004
[Files]
ResultatenDatabase                                                              =
D:\Projekten\Graadmeters\Move\Modellen\StepBIC2\Move4StepBIC2.mdb
Statistiek     = D:\Projekten\Graadmeters\Move\Modellen\StepBIC2\Move4stepBIC2.txt
Taxa                                                                            =
D:\Projekten\Graadmeters\Move\Modellen\StepBIC2\Move4stepBIC2_Taxa.txt
Variabelen                                                                      =
D:\Projekten\Graadmeters\Move\Modellen\StepBIC2\Move4stepBIC2_Variabelen.txt
Coefficienten                                                                   =
D:\Projekten\Graadmeters\Move\Modellen\StepBIC2\Move4stepBIC2_Coef.txt
Correlaties                                                                     =
D:\Projekten\Graadmeters\Move\Modellen\StepBIC2\Move4stepBIC2_Cor.txt
Covarianties                                                                    =
D:\Projekten\Graadmeters\Move\Modellen\StepBIC2\Move4stepBIC2_Cov.txt
```

Listing 2.2. Voorbeeld van een taxa-bestand

```
TaxonNr,TaxonCode,TaxonNaam
1,1,Acer campestre
2,2,Acer pseudoplatanus
3,4,Achillea millefolium
4,5,Achillea ptarmica
5,7,Acorus calamus
6,8,Actaea spicata
```

Listing 2.3. Voorbeeld van een Variabelen-bestand voor Move

```
VariabeleNr,VariabeleCode,VariabeleNaam,VariabeleOmschrijving,VariabeleEenheid,Var
iabeleType,VariabeleTransformatie
1,f,f,vochtgetal,-,continue,-
2,r,r,zuurgetal,-,continue,-
3,n,n,stikstofgetal,-,continue,-
4,s,s,zoutgetal,-,continue,-
5,veg1,veg1,loofbos,binair,-
6,veg2,veg2,grasland,binair,-
7,veg3,veg3,heide,binair,-
8,veg4,veg4,dennenbos,binair,-
9,veg5,veg5,sparrenbos,binair,-
10,fgr1,fgr1,Heuvelland,binair,-
11,fgr2,fgr2,Hogere zandgronden,binair,-
12,fgr3,fgr3,Rivierengebied,binair,-
13,fgr4,fgr4,Laagveengebied,binair,-
14,fgr5,fgr5,Zeekleigebied,binair,-
15,fgr6,fgr6,Duingebied,binair,-
16,fgr7,fgr7,Afgesloten Zeearmen,binair,-
17,fgr8,fgr8,Getijdengebied,binair,-
18,fgr9,fgr9,Noordzee,binair,-
```

Listing 2.4. Procedures voor verwerken van SPlus-resultaten (1)

```
Option Compare Database
Option Explicit

Dim ModelNaam As String
Dim ModelOpmerkingen As String
Dim ResultatenDatabaseNaam As String
Dim StatistiekBestandsnaam As String
Dim TaxaBestandsnaam As String
Dim VariabelenBestandsnaam As String
Dim CoefficientenBestandsnaam As String
Dim CoefficientenRuwBestandsnaam As String
Dim CorrelatiesBestandsnaam As String
Dim CovariantiesBestandsnaam As String

Function VerwerkResultaten(StuurBestandnaam As String)
    Dim sql As String

    If Len(StuurBestandnaam) = 0 Then
      MsgBox "Geen naam van een stuurbestand opgegeven"
    ElseIf Len(Dir(StuurBestandnaam)) = 0 Then
      MsgBox "Stuurbestand '" & StuurBestandnaam & "' bestaat niet"
    Else

      ' Verwerk Stuurbestand
      ProcessIniFile StuurBestandnaam

      ' Maak database
      If Len(Dir(ResultatenDatabaseNaam)) > 0 Then Kill ResultatenDatabaseNaam
      CreateDatabase ResultatenDatabaseNaam, dbLangGeneral
      ' Maak Tabellen aan
      DoCmd.TransferDatabase acExport, "Microsoft Access", ResultatenDatabaseNaam,
acTable, "ModelInfo Template", "ModelInfo"
      RefreshLink "Statistiek", ResultatenDatabaseNaam, "ModelInfo"
      DoCmd.TransferDatabase acExport, "Microsoft Access", ResultatenDatabaseNaam,
acTable, "Taxa Template", "Taxa"
      RefreshLink "Statistiek", ResultatenDatabaseNaam, "Taxa"
      DoCmd.TransferDatabase acExport, "Microsoft Access", ResultatenDatabaseNaam,
acTable, "Variabelen Template", "Variabelen"
      RefreshLink "Statistiek", ResultatenDatabaseNaam, "Variabelen"
      DoCmd.TransferDatabase acExport, "Microsoft Access", ResultatenDatabaseNaam,
acTable, "Statistiek Template", "Statistiek"
      RefreshLink "Statistiek", ResultatenDatabaseNaam, "Statistiek"
      DoCmd.TransferDatabase acExport, "Microsoft Access", ResultatenDatabaseNaam,
acTable, "CoefficientenRuw Template", "CoefficientenRuw"
      RefreshLink "Statistiek", ResultatenDatabaseNaam, "CoefficientenRuw"
      DoCmd.TransferDatabase acExport, "Microsoft Access", ResultatenDatabaseNaam,
acTable, "Coefficienten Template", "Coefficienten"
```

```
      RefreshLink "Statistiek", ResultatenDatabaseNaam, "Coefficienten"
      DoCmd.TransferDatabase acExport, "Microsoft Access", ResultatenDatabaseNaam,
acTable, "Correlaties Template", "Correlaties"
      RefreshLink "Statistiek", ResultatenDatabaseNaam, "Correlaties"
      DoCmd.TransferDatabase acExport, "Microsoft Access", ResultatenDatabaseNaam,
acTable, "Covarianties Template", "Covarianties"
      RefreshLink "Statistiek", ResultatenDatabaseNaam, "Covarianties"

      ' Lees ModelInfo
      Dim tmpSet As Recordset
      WisTabel "ModelInfo"
      Set tmpSet = CurrentDb.OpenRecordset("ModelInfo")
      tmpSet.AddNew
      tmpSet!ModelNaam = ModelNaam
      tmpSet!Opmerkingen = ModelOpmerkingen
      tmpSet!datum = Now
      tmpSet.Update

      ' Lees Statistiek
      LeesStatistiek "Statistiek", StatistiekBestandsnaam
      ' Lees Taxa
      LeesTaxa "Taxa", TaxaBestandsnaam
      '  Lees  Variabelen  als  bestand  aanwezig  is,  anders:  haal  ze  uit  de
coefficienten-file
      LeesVariabelen "Variabelen", VariabelenBestandsnaam

      ' Lees coefficienten en verwerk ruwe coefficienten
      VerwerkCoefficienten CoefficientenBestandsnaam

      ' Lees Correlaties
      LeesCorrelaties "Correlaties", CorrelatiesBestandsnaam

      ' Lees Covarianties
      LeesCovarianties "Covarianties", CovariantiesBestandsnaam

      ' Verwijder links naar tabellen
      DeleteLink "ModelInfo"
      DeleteLink "Taxa"
      DeleteLink "Variabelen"
      DeleteLink "Statistiek"
      DeleteLink "Coefficienten"
      DeleteLink "CoefficientenRuw"
      DeleteLink "Correlaties"
      DeleteLink "Covarianties"

    End If

End Function

Function ProcessIniFile(IniFileName As String)
On Error GoTo Error_ProcessIniFile

    Dim filenum As Long, arg1 As String, arg2 As String, StrLine As String
    Dim narg As Long, strArray(100) As String

    ModelNaam = GetIniFileItem(IniFileName, "Algemeen", "Model")
    ModelOpmerkingen = GetIniFileItem(IniFileName, "Algemeen", "Opmerkingen")
    ResultatenDatabaseNaam       =       GetIniFileItem(IniFileName,       "Files",
"ResultatenDatabase ")
    StatistiekBestandsnaam = GetIniFileItem(IniFileName, "Files", "Statistiek ")
    TaxaBestandsnaam = GetIniFileItem(IniFileName, "Files", "Taxa ")
    VariabelenBestandsnaam = GetIniFileItem(IniFileName, "Files", "Variabelen ")
    CoefficientenBestandsnaam      =      GetIniFileItem(IniFileName,      "Files",
"Coefficienten ")
    CorrelatiesBestandsnaam = GetIniFileItem(IniFileName, "Files", "Correlaties ")
    CovariantiesBestandsnaam  =  GetIniFileItem(IniFileName,  "Files",  "Covarianties
")

    ' Check input
    If Len(ModelNaam) = 0 Or _
       Len(ModelOpmerkingen) = 0 Or _
       Len(ResultatenDatabaseNaam) = 0 Or _
```

```
       Len(StatistiekBestandsnaam) = 0 Or _
       Len(TaxaBestandsnaam) = 0 Or _
       Len(VariabelenBestandsnaam) = 0 Or _
       Len(CoefficientenBestandsnaam) = 0 Or _
       Len(CorrelatiesBestandsnaam) = 0 Or _
       Len(CovariantiesBestandsnaam) = 0 Then
       MsgBox "Error in initialization file: No or incorrect filename specified"
      End
    End If

Einde_ProcessIniFile:
    Close #filenum
    Exit Function

Error_ProcessIniFile:
    MsgBox "Error in function ProcessIniFile; err=" & Err & ": " & Err.Description
    Resume Einde_ProcessIniFile

End Function

Function LeesStatistiek(TabelNaam As String, FileNaam As String)
    If Not LoadTable(TabelNaam, FileNaam, " ") Then
      MsgBox "Loading file '" & FileNaam & "' failed."
      'GoTo Error_RunJob
    End If

End Function

Function LeesTaxa(TabelNaam As String, FileNaam As String)
    If Not LoadTable(TabelNaam, FileNaam, ",") Then
      MsgBox "Loading file '" & FileNaam & "' failed."
      'GoTo Error_RunJob
    End If
End Function

Function LeesVariabelen(TabelNaam As String, FileNaam As String)
    If Not LoadTable(TabelNaam, FileNaam, ",") Then
      MsgBox "Loading file '" & FileNaam & "' failed."
      'GoTo Error_RunJob
    End If
End Function

Function LoadTable(TableName As String, InputFileName As String, Delim As String)
As Integer
On Error GoTo 0
    Dim iField As Integer, sql As String, dumval
    Dim strFieldNames As String
    Dim FieldArray(100) As String, FieldIndex(100) As Integer, nFieldNames As
Integer
    Dim tmpSet As Recordset
    Const tmpInputTableName = "tmp Input"

    DoCmd.TransferText acLinkDelim, , tmpInputTableName, InputFileName, True
    CurrentDb.TableDefs.Refresh
    DBEngine.Idle dbRefreshCache
    DoEvents
    ' Test for one or more fields
    ' if more fields: delimination worked apparently well: test existence of all
fields needed and load them
    If CurrentDb.TableDefs(tmpInputTableName).Fields.Count > 1 Then
      For iField = 0 To CurrentDb.TableDefs(TableName).Fields.Count - 1
        On Error Resume Next
        dumval =
CurrentDb.TableDefs(tmpInputTableName).Fields(CurrentDb.TableDefs(TableName).Field
s(iField).Name).Value
        If Err > 0 Then
          MsgBox "Error: Field '" &
CurrentDb.TableDefs(TableName).Fields(iField).Name & "' missing from table '" &
tmpInputTableName & "'"
          GoTo Error_LoadTable
        End If
        On Error GoTo 0
```

```
      Next iField
      ' Fields checked and present
      For iField = 0 To CurrentDb.TableDefs(TableName).Fields.Count - 1
        strFieldNames         =         strFieldNames         &         ",["         &
CurrentDb.TableDefs(TableName).Fields(iField).Name & "]"
      Next iField
      strFieldNames = Mid(strFieldNames, 2)
      sql = "INSERT INTO [" & TableName & "] ( " & strFieldNames & " ) " & _
            "SELECT " & strFieldNames & " " & _
            "FROM [" & tmpInputTableName & "];"
      CurrentDb.Execute sql
    Else   ' fields not deliminated
      ' Reload file without headings
      DoCmd.DeleteObject acTable, tmpInputTableName
      CurrentDb.TableDefs.Refresh
      DBEngine.Idle dbRefreshCache
      DoEvents
      DoCmd.TransferText acLinkDelim, , tmpInputTableName, InputFileName, False
      CurrentDb.TableDefs.Refresh
      DBEngine.Idle dbRefreshCache
      DoEvents

      Dim SourceSet As Recordset, TargetSet As Recordset
      Set SourceSet = CurrentDb.OpenRecordset(tmpInputTableName)
      strFieldNames  =  SourceSet.Fields(0).Value    '  get  fieldnames  form  first
record
      For iField = 0 To CurrentDb.TableDefs(TableName).Fields.Count - 1
        FieldIndex(iField)              =              GetWordIndex(strFieldNames,
CurrentDb.TableDefs(TableName).Fields(iField).Name, Delim)
        If FieldIndex(iField) = 0 Then
          MsgBox             "Error:             Field             '"             &
CurrentDb.TableDefs(TableName).Fields(iField).Name  &  "'  missing  from  file  '"  &
InputFileName & "'"
          GoTo Error_LoadTable
        End If
      Next iField
      Set TargetSet = CurrentDb.OpenRecordset(TableName)

      SourceSet.MoveNext   ' skip heading
      While Not SourceSet.EOF
        If Not IsNull(SourceSet(0)) Then
        nFieldNames = SplitStringArg(FieldArray, SourceSet(0), Delim)
        TargetSet.AddNew
        For iField = 0 To CurrentDb.TableDefs(TableName).Fields.Count - 1
          If TargetSet(iField).Type = dbText Then
            TargetSet(iField).Value   =   IIf(Len(FieldArray(iField   +   1))   >   0,
FieldArray(iField + 1), Null)
          Else
            If iField < nFieldNames Then
              TargetSet(iField).Value   =   IIf(Len(FieldArray(iField   +   1))   >   0,
CDbl(FieldArray(iField + 1)), Null)
            End If
          End If
        Next iField
        TargetSet.Update
        End If
        SourceSet.MoveNext
      Wend
      SourceSet.Close
      TargetSet.Close

    End If
    LoadTable = True

End_LoadTable:
    DoCmd.DeleteObject acTable, tmpInputTableName
    Exit Function

Error_LoadTable:
    LoadTable = False
    GoTo End_LoadTable
```

```
End Function


Function VerwerkCoefficienten(CoefFilenaam As String, Optional DoeMaakVariabelen
As Integer = False)
    Dim sql As String

      ' Lees Coefficienten
      LeesCoefficienten "CoefficientenRuw", CoefFilenaam

      If DoeMaakVariabelen Then
        ' Voor een eerste slag kan het zinvol zijn om de variabelen uit het
coefficienten-bestand te halen
        sql = "SELECT CoefficientenRuw.VariabeleNaam,
IIf(Left([VariabeleNaam],1)='(',Mid([VariabeleNaam],2),[VariabeleNaam]) AS var1,
IIf(Right([Var1],1)=')',Left([Var1],Len([var1])-1),[var1]) AS var2,
IIf(Right([Var2],2)='^2',Left([Var2],Len([var2])-2),[var2]) AS var3,
IIf(Left([var3],2)='i(',Mid([var3],3),[var3]) AS var4, [var4] AS var5 " & _
              "FROM CoefficientenRuw " & _
              "WHERE (((CoefficientenRuw.VariabeleNaam) Not Like '*:*' And Not
(CoefficientenRuw.VariabeleNaam)='(Intercept)'));"
        On Error Resume Next
        Dim qd As QueryDef
        Set qd = CurrentDb.CreateQueryDef("tmp sql")
        qd.Close
        Set qd = CurrentDb.QueryDefs("tmp sql")
        On Error GoTo 0
        qd.sql = sql
        qd.Close
        CurrentDb.QueryDefs.Refresh
        DoEvents
        DoCmd.DeleteObject acTable, "tmp var"
        DoCmd.CopyObject , "tmp var", acTable, "tmp var Template"
        On Error GoTo 0
        sql = "INSERT INTO [tmp var] ( VariabeleCode, VariabeleNaam ) " & _
              "SELECT [tmp sql].var5, [tmp sql].var5 " & _
              "FROM [tmp sql] " & _
              "GROUP BY [tmp sql].var5, [tmp sql].var5;"
        CurrentDb.Execute sql
        sql = "INSERT INTO Variabelen ( VariabeleNr, VariabeleCode, VariabeleNaam
) " & _
              "SELECT [tmp var].VariabeleNr, [tmp var].VariabeleCode, [tmp
var].VariabeleNaam " & _
              "FROM [tmp var];"
        CurrentDb.Execute sql
      End If

      ' Zet ruwe resultaten coefficienten om
      AanpassenCoefficienten

End Function


Function AanpassenCoefficienten()
    Dim sql As String

    ' Intercepten
    sql = "INSERT INTO Coefficienten ( TaxonNr, TermType, Coeff, CoeffSE, VarNrA,
VarNrB, Volgorde ) " & _
          "SELECT Taxa.TaxonNr, 'c' AS TermType, CoefficientenRuw.Waarde,
CoefficientenRuw.SE, 0 AS VarNrA, 0 AS VarNrB, VolgordeNr " & _
          "FROM CoefficientenRuw INNER JOIN Taxa ON CoefficientenRuw.SoortNaam =
Taxa.TaxonCode " & _
          "WHERE (((CoefficientenRuw.VariabeleNaam)='(intercept)'));"
    CurrentDb.Execute sql

    ' Gelijke namen
    sql = "INSERT INTO Coefficienten ( TaxonNr, VarNrA, VarNrB, TermType, Coeff,
CoeffSE, Volgorde ) " & _
          "SELECT Taxa.TaxonNr, Variabelen.VariabeleNr, 0 AS VarNrB, 'cA' AS
TermType, CoefficientenRuw.Waarde, CoefficientenRuw.SE, VolgordeNr " & _
```

```
          "FROM (CoefficientenRuw INNER JOIN Variabelen ON
CoefficientenRuw.VariabeleNaam = Variabelen.VariabeleCode) INNER JOIN Taxa ON
CoefficientenRuw.SoortNaam = Taxa.TaxonCode;"
    CurrentDb.Execute sql

    ' Kwadratische termen
    sql = "INSERT INTO Coefficienten ( TermType, VarNrA, TaxonNr, Coeff, CoeffSE,
VarNrB, Volgorde ) " & _
          "SELECT 'cA2', Variabelen.VariabeleNr, Taxa.TaxonNr,
CoefficientenRuw.Waarde, CoefficientenRuw.SE, 0 AS VarNrB, VolgordeNr " & _
          "FROM Variabelen, CoefficientenRuw INNER JOIN Taxa ON
CoefficientenRuw.SoortNaam = Taxa.TaxonCode " & _
          "WHERE (((CoefficientenRuw.VariabeleNaam) Like '*^*' And
(CoefficientenRuw.VariabeleNaam) Not Like '*:*') AND
((Mid([CoefficientenRuw].[VariabeleNaam],3,Len([CoefficientenRuw].[Variabelenaam])
-5))=[Variabelen].[VariabeleCode]));"
    CurrentDb.Execute sql

    ' Interacties zonder kwadratische termen
    sql = "INSERT INTO Coefficienten ( TaxonNr, TermType, VarNrA, VarNrB, Coeff,
CoeffSE, Volgorde ) " & _
          "SELECT Taxa.TaxonNr, 'cAB' AS TermType, Variabelen.VariabeleNr AS
VarNrA, Variabelen_1.VariabeleNr AS VarNrB, CoefficientenRuw.Waarde,
CoefficientenRuw.SE, VolgordeNr " & _
          "FROM Variabelen, Variabelen AS Variabelen_1, CoefficientenRuw INNER
JOIN Taxa ON CoefficientenRuw.SoortNaam = Taxa.TaxonCode " & _
          "WHERE (((CoefficientenRuw.VariabeleNaam) Not Like '*^*' And
(CoefficientenRuw.VariabeleNaam) Like '*:*') AND
((Left([CoefficientenRuw].[VariabeleNaam],InStr([CoefficientenRuw].[Variabelenaam]
,':')-1))=[Variabelen].[VariabeleCode]) AND
((Mid([CoefficientenRuw].[VariabeleNaam],InStr([CoefficientenRuw].[Variabelenaam],
':')+1))=[Variabelen_1].[VariabeleCode]));"
    CurrentDb.Execute sql
    DoEvents

    ' Interacties met een kwadratische term links
    sql = "INSERT INTO Coefficienten ( TaxonNr, TermType, VarNrA, VarNrB, Coeff,
CoeffSE, Volgorde ) " & _
          "SELECT Taxa.TaxonNr, 'cA2B' AS TermType, Variabelen.VariabeleNr AS
VarNrA, Variabelen_1.VariabeleNr AS VarNrB, CoefficientenRuw.Waarde,
CoefficientenRuw.SE, VolgordeNr " & _
          "FROM Variabelen, Variabelen AS Variabelen_1, CoefficientenRuw INNER
JOIN Taxa ON CoefficientenRuw.SoortNaam = Taxa.TaxonCode " & _
          "WHERE (((CoefficientenRuw.VariabeleNaam) Like '*^2*:*') AND
((Mid(Left([CoefficientenRuw].[VariabeleNaam],InStr([CoefficientenRuw].[Variabelen
aam],':')-1),3,InStr([CoefficientenRuw].[Variabelenaam],'^')-
3))=[Variabelen].[VariabeleCode]) AND
((Mid([CoefficientenRuw].[VariabeleNaam],InStr([CoefficientenRuw].[Variabelenaam],
':')+1))=[Variabelen_1].[VariabeleCode]));"
    CurrentDb.Execute sql
    DoEvents

    ' Interacties met een kwadratische term rechts
    sql = "INSERT INTO Coefficienten ( TaxonNr, TermType, VarNrA, VarNrB, Coeff,
CoeffSE, Volgorde ) " & _
          "SELECT Taxa.TaxonNr, 'cAB2' AS TermType, Variabelen.VariabeleNr AS
VarNrA, Variabelen_1.VariabeleNr AS VarNrB, CoefficientenRuw.Waarde,
CoefficientenRuw.SE, VolgordeNr " & _
          "FROM Variabelen, Variabelen AS Variabelen_1, CoefficientenRuw INNER
JOIN Taxa ON CoefficientenRuw.SoortNaam = Taxa.TaxonCode " & _
          "WHERE (((CoefficientenRuw.VariabeleNaam) Like '*:*^2*' And
(CoefficientenRuw.VariabeleNaam) Not Like '*^2*:*^2*') AND
((Left([CoefficientenRuw].[VariabeleNaam],InStr([CoefficientenRuw].[Variabelenaam]
,':')-1))=[Variabelen].[VariabeleCode]) AND
((Mid(Mid([CoefficientenRuw].[VariabeleNaam],InStr([CoefficientenRuw].[Variabelena
am],':')+1),3,InStr([CoefficientenRuw].[Variabelenaam],'^')-
InStr([CoefficientenRuw].[Variabelenaam],':')-
3))=[Variabelen_1].[VariabeleCode]));"
    CurrentDb.Execute sql
    DoEvents

    ' Interacties met twee kwadratische termen
```

```
    sql = "INSERT INTO Coefficienten ( TaxonNr, TermType, VarNrA, VarNrB, Coeff,
CoeffSE, Volgorde ) " & _
          "SELECT Taxa.TaxonNr, 'cA2B2' AS TermType, Variabelen.VariabeleNr AS
VarNrA, Variabelen_1.VariabeleNr AS VarNrB, CoefficientenRuw.Waarde,
CoefficientenRuw.SE, VolgordeNr " & _
          "FROM Variabelen, Variabelen AS Variabelen_1, CoefficientenRuw INNER
JOIN Taxa ON CoefficientenRuw.SoortNaam = Taxa.TaxonCode " & _
          "WHERE (((CoefficientenRuw.VariabeleNaam) Like '*^2*:*^2*') AND
((Mid([CoefficientenRuw].[VariabeleNaam],3,InStr([CoefficientenRuw].[Variabelenaam
],'^')-3))=[Variabelen].[VariabeleCode]) AND
((Left(Mid([CoefficientenRuw].[VariabeleNaam],InStr([CoefficientenRuw].[Variabelen
aam],':')+3),InStr(InStr([CoefficientenRuw].[Variabelenaam],':')+1,[CoefficientenR
uw].[Variabelenaam],'^')-InStr([CoefficientenRuw].[Variabelenaam],':')-
3))=[Variabelen_1].[VariabeleCode]));"
    CurrentDb.Execute sql
    DoEvents

End Function

Function GetIniFileItem(IniFileName As String, Section As String, Item As String)
As String
    Dim filenum As Long
    Dim CurrentSection As String, CurrentItem As String
    Dim Found As Integer, tmpstr As String
    Dim SectionFound As Integer, SectionProcessed As Integer

    filenum = FreeFile
    Open IniFileName For Input As filenum
    CurrentSection = ""
    SectionFound = False
    SectionProcessed = False
    Found = False
    While Not EOF(filenum) And Not SectionProcessed And Not Found
      Input #filenum, tmpstr
      tmpstr = Trim(tmpstr)
      If Left(tmpstr, 1) = "[" And Right(tmpstr, 1) = "]" Then
        CurrentSection = Mid(tmpstr, 2, Len(tmpstr) - 2)
        If CurrentSection = Section Then
          SectionFound = True
        Else
          If SectionFound Then
            SectionProcessed = True
          End If
          SectionFound = False
        End If
      ElseIf SectionFound And Left(tmpstr, 2) <> "//" Then
        If InStr(tmpstr, "=") > 0 Then
          CurrentItem = Trim(Left(tmpstr, InStr(tmpstr, "=") - 1))
          If CurrentItem = Item Then
            Found = True
            GetIniFileItem = Trim(Mid(tmpstr, InStr(tmpstr, "=") + 1))
          End If
        End If
      End If
    Wend
    Close filenum
    If Not Found Then
      GetIniFileItem = ""
    End If
End Function

Function RefreshLink(TargetTableName As String, DatabaseName As String,
SourceTableName As String)
    DeleteLink TargetTableName
    DoCmd.TransferDatabase acLink, "Microsoft Access", DatabaseName, acTable,
SourceTableName, TargetTableName
End Function

Function DeleteLink(TableName As String)
    On Error Resume Next
    DoCmd.Close acTable, TableName, acSaveNo
    DoCmd.DeleteObject acTable, TableName
```

```
    CurrentDb.TableDefs.Refresh
    On Error GoTo 0
End Function
```

Listing 2.4. Procedures voor verwerken van SPlus-resultaten (2): coefficienten

```
Option Compare Database
Option Explicit
Const MaxAantalWoorden = 1000
Const MaxAantalVariabelen = 1000
Private Const UpdateIntervalMeter = 100

Function LeesCoefficienten(DoelTabelNaam As String, CoefFilenaam As String)
    Dim tset As Recordset
    Dim filenum As Integer
    Dim StrArr(MaxAantalWoorden) As String, StrArg As Long
    Dim str1 As String
    Dim ModelType As String, Soortnaam As String,
VariabeleNaam(MaxAantalVariabelen) As String, AantalVariabelen As Integer
    Dim CoeffWaarde(MaxAantalVariabelen) As Double, CoeffSE(MaxAantalVariabelen)
As Double, CoeffTWaarde(MaxAantalVariabelen) As Double
    Dim AantalWoorden As Long
    Dim i As Integer, iSoort As Long

    SysCmd acSysCmdInitMeter, "Inlezen coefficienten", 1

    WisTabel DoelTabelNaam

    Set tset = CurrentDb.OpenRecordset(DoelTabelNaam)

    filenum = FreeFile

    Open CoefFilenaam For Input As filenum

    Line Input #filenum, str1  ' file type: Coefficients
    If str1 <> "Coefficients" Then
        MsgBox "Bestand '" & CoefFilenaam & "' is geen coefficienten-bestand"
        Exit Function
    End If

    Line Input #filenum, str1  ' modeltype: Variable / Fixed
    AantalWoorden = SplitString(StrArr, str1)
    If AantalWoorden <> 2 Then
        MsgBox "Foute specificatie van modeltype op regel 2"
        Exit Function
    End If
    ModelType = StrArr(2)
    iSoort = 1
    While Not EOF(filenum)
      ' Lees Soortnaam
      Line Input #filenum, Soortnaam
      ' Lees aantal variabelen
      If Len(Soortnaam) > 0 Then
        If iSoort Mod UpdateIntervalMeter = 1 Then
          SysCmd acSysCmdInitMeter, "Inlezen coefficienten (" & ((iSoort \
UpdateIntervalMeter)) * UpdateIntervalMeter + 1 & "-" & ((iSoort \
UpdateIntervalMeter) + 1) * UpdateIntervalMeter & ": ", UpdateIntervalMeter
        End If
        Line Input #filenum, str1
        AantalWoorden = SplitString(StrArr, str1)
        AantalVariabelen = CInt(StrArr(2))
        If ModelType = "Fixed" And iSoort = 1 Then
          ' Voor 1 tot aantal variabelen: lees variabelenaam
          For i = 1 To AantalVariabelen
            Line Input #filenum, VariabeleNaam(i)
          Next i
        End If
        ' Lees Coefficientwaarden voor alle variabelen: regel 1
        Line Input #filenum, str1
        AantalWoorden = SplitString(StrArr, str1)
```

```
        For i = 1 To AantalVariabelen
          CoeffWaarde(i) = CDbl(StrArr(i))
        Next i
        ' Lees Standard Errors van Coefficientwaarden voor alle variabelen: regel
2
        Line Input #filenum, str1
        AantalWoorden = SplitString(StrArr, str1)
        For i = 1 To AantalVariabelen
          CoeffSE(i) = CDbl(StrArr(i))
        Next i
        ' Lees t-waarde van Coefficientwaarden voor alle variabelen: regel 3
        Line Input #filenum, str1
        AantalWoorden = SplitString(StrArr, str1)
        For i = 1 To AantalVariabelen
          CoeffTWaarde(i) = CDbl(StrArr(i))
        Next i
        ' Sla gegevens op in tabel DoelTabelNaam
        For i = 1 To AantalVariabelen
          tset.AddNew
          tset!Soortnaam = Trim(Soortnaam)
          tset!VolgordeNr = i
          tset!VariabeleNaam = Trim(VariabeleNaam(i))
          tset!Waarde = CoeffWaarde(i)
          tset!SE = CoeffSE(i)
          tset![t-waarde] = CoeffTWaarde(i)
          tset.Update
        Next i
        iSoort = iSoort + 1
        SysCmd acSysCmdUpdateMeter, iSoort Mod UpdateIntervalMeter
      End If ' len(Soortnaam) > 0
    Wend
    Close #filenum
    tset.Close
    SysCmd acSysCmdRemoveMeter

End Function
```

Listing 2.4. Procedures voor verwerken van SPlus-resultaten (3): correlaties en covarianties

```
Option Compare Database
Option Explicit
Const MaxAantalWoorden = 1000
Const MaxAantalVariabelen = 1000
Private Const UpdateIntervalMeter = 100

Function LeesCorrelaties(DoelTabelNaam As String, BronFileNaam As String)
    Dim tset As Recordset
    Dim filenum As Integer
    Dim StrArr(MaxAantalWoorden) As String, StrArg As Long
    Dim str1 As String
    Dim ModelType As String, Soortnaam As String, AantalVariabelen As Integer
    Dim CorWaarde(MaxAantalVariabelen, MaxAantalVariabelen) As Double
    Dim AantalWoorden As Long
    Dim i As Integer, j As Integer
    Dim iSoort As Long

    SysCmd acSysCmdInitMeter, "Inlezen correlaties in '" & DoelTabelNaam & "'", 1

    filenum = FreeFile

    Open BronFileNaam For Input As filenum

    Line Input #filenum, str1  ' file type: Coefficients
    If str1 <> "Correlations" Then
        MsgBox "Bestand '" & BronFileNaam & "' is geen correlatie-bestand"
        Exit Function
    End If

    WisTabel DoelTabelNaam

    Set tset = CurrentDb.OpenRecordset(DoelTabelNaam)
```

```
    Line Input #filenum, str1  ' modeltype: Variable / Fixed
    AantalWoorden = SplitString(StrArr, str1)
    If AantalWoorden <> 2 Then
        MsgBox "Foute specificatie van modeltype op regel 2"
        Exit Function
    End If
    ModelType = StrArr(2)
    iSoort = 1
    While Not EOF(filenum)
      ' Lees Soortnaam
      Line Input #filenum, Soortnaam
      If Len(Soortnaam) > 0 Then
        If iSoort Mod UpdateIntervalMeter = 1 Then
          SysCmd acSysCmdInitMeter, "Inlezen coefficienten (" & ((iSoort \
UpdateIntervalMeter)) * UpdateIntervalMeter + 1 & "-" & ((iSoort \
UpdateIntervalMeter) + 1) * UpdateIntervalMeter & ": ", UpdateIntervalMeter
        End If
        ' Lees aantal variabelen
        Line Input #filenum, str1
        AantalVariabelen = SplitString(StrArr, str1)
        'AantalVariabelen = CInt(StrArr(2))
        ' Voor 1 tot aantal variabelen: lees variabelenaam
        For i = 2 To AantalVariabelen
          Line Input #filenum, str1
          AantalWoorden = SplitString(StrArr, str1)
          For j = 1 To i - 1
            tset.AddNew
            tset!Soortnaam = Soortnaam
            tset!VariabeleNr1 = i
            tset!VariabeleNr2 = j
            tset!Waarde = CDbl(StrArr(j))
            tset.Update
            DoEvents
          Next j
        Next i
      End If
      iSoort = iSoort + 1
      SysCmd acSysCmdUpdateMeter, iSoort Mod UpdateIntervalMeter
    Wend
    Close #filenum
    tset.Close
    SysCmd acSysCmdRemoveMeter

End Function


Function LeesCovarianties(DoelTabelNaam As String, BronFileNaam As String)
    Dim tset As Recordset
    Dim filenum As Integer
    Dim StrArr(MaxAantalWoorden) As String, StrArg As Long
    Dim str1 As String
    Dim ModelType As String, Soortnaam As String, AantalVariabelen As Integer
    Dim CorWaarde(MaxAantalVariabelen, MaxAantalVariabelen) As Double
    Dim AantalWoorden As Long
    Dim i As Integer, j As Integer, iSoort As Long

    SysCmd acSysCmdInitMeter, "Inlezen covarianties in '" & DoelTabelNaam & "'", 1

    filenum = FreeFile

    Open BronFileNaam For Input As filenum

    Line Input #filenum, str1  ' file type: Coefficients
    If str1 <> "CovariancesUnscaled" Then
        MsgBox "Bestand '" & BronFileNaam & "' is geen covariantie-bestand"
        Exit Function
    End If


    WisTabel DoelTabelNaam

    Set tset = CurrentDb.OpenRecordset(DoelTabelNaam)
```

```
    Line Input #filenum, str1  ' modeltype: Variable / Fixed
    AantalWoorden = SplitString(StrArr, str1)
    If AantalWoorden <> 2 Then
        MsgBox "Foute specificatie van modeltype op regel 2"
        Exit Function
    End If
    ModelType = StrArr(2)
    iSoort = 1
    While Not EOF(filenum)
      ' Lees Soortnaam
      Line Input #filenum, Soortnaam
      Soortnaam = Trim(Soortnaam)
      If Len(Soortnaam) > 0 Then
        If iSoort Mod UpdateIntervalMeter = 1 Then
          SysCmd acSysCmdInitMeter, "Inlezen coefficienten (" & ((iSoort \
UpdateIntervalMeter)) * UpdateIntervalMeter + 1 & "-" & ((iSoort \
UpdateIntervalMeter) + 1) * UpdateIntervalMeter & ": ", UpdateIntervalMeter
        End If
      ' Lees aantal variabelen
        Line Input #filenum, str1
        AantalVariabelen = SplitString(StrArr, str1)
        ' Sla eerste waarde op
        tset.AddNew
        tset!Soortnaam = Soortnaam
        tset!CoeffNr1 = 1
        tset!CoeffNr2 = 1
        tset!Waarde = CDbl(StrArr(1))
        tset.Update
        For i = 2 To AantalVariabelen
          Line Input #filenum, str1
          AantalWoorden = SplitString(StrArr, str1)
          For j = 1 To i
            tset.AddNew
            tset!Soortnaam = Soortnaam
            tset!CoeffNr1 = i
            tset!CoeffNr2 = j
            tset!Waarde = CDbl(StrArr(j))
            tset.Update
            DoEvents
          Next j
        Next i
      End If
      iSoort = iSoort + 1
      SysCmd acSysCmdUpdateMeter, iSoort Mod UpdateIntervalMeter
    Wend
    Close #filenum
    tset.Close
    SysCmd acSysCmdRemoveMeter

End Function
```

Appendix 3 Procedure voor berekenen van Move-resultaten

Gebruikmakend van de opslagstructuur van de Splus-resultaten in een database (zie Appendix 2), zijn er procedures opgesteld om in Access de kans-op-voorkomen van soorten te berekenen. Hierbij wordt gebruikgemaakt van de functionaliteit van Queries in Access, wat een versnelling oplevert t.o.v. uitgeprogrammeerde berekeningen in loops (Access, Delphi). De versnelling die dit oplevert is afhankelijk van het probleem, met name de grootte van de brokken aan data die aangeboden worden. Bij kleine brokken worden de query-mogelijkheden van Access niet volledig benut. Te grote brokken levert een trage verwerking of een overflow van tijdelijke Access-bestanden op. Bij de gevoeligheidsanalyse van de Natuurplanner trad een versnelling op van een factor 30-40, afhankelijk van de volgorde waarin brokken aangeleverd werden (berekeningen zonder betrouwbaarheidsintervallen).

In het Move-initialisatiebestand wordt gespecificeerd welke set van responsiemodellen doorgerekend moet worden (resultaat van de bewerkingen die beschreven staan in Appendix 2), hoe de variabelen van de responsiemodellen gerelateerd zijn aan de variabelen in de invoer, welk biodiv-initialisatiebestand gebruikt moet worden, en waar de resultaten naar weg geschreven moeten worden (Access-database). Als optie kan aangegeven of er betrouwbaarheidsintervallen bij de kansen-op-voorkomen bereken moeten worden.

In de file die de vertaling van invoer-variabelen naar responsie-variabelen specificeert is opgenomen:

1. Id, (niet gebruikt);
2. Naam volgens invoer (zie biodiv-file sectie [Files]);
3. Data type (Integer, Long, Code, Double, FactorNum, FactorTxt);
4. Naam volgens de responsiemodellen, en
5. Eventueel benodigde transformatie van de invoervariabelen voordat deze gebruikt worden in de responsiemodellen (mogelijkheden momenteel: '-', 'Log(X)', 'Log(X+1)'.

Bij factor-variabelen is een extra vertaal-tabel nodig om de afzonderlijke waarden te relateren. Er kan zo rekening gehouden worden met verschillen in omnummering van codes (variabele 'veg'!; zie listing 3.3).

Gekozen is om de berekening van de responsiemodellen transparant te laten zijn aan de berekeningen zoals die nu in Biodiv plaatsvinden. Daarom is uitgegaan van de commandostructuur van Biodiv. Niet alle Biodiv-instellingen worden momenteel gebruikt. De specificatie van de set van responsiemodellen vindt b.v. in de Move-initialisatiefile plaats. Wel gebruikt worden:
TempDir, MultiRun, Description, ScenarioYear, SpecRefTable, UnitColumnName, UnitMapName, [Files], [runs] en [runx] waarbij x het run-nummer aangeeft. Specificaties in [runx]-sectie overschrijven de specificaties in de [Algemeen]-sectie. De sectie [Files] wordt geïnterpreteerd als de complete set van invoergrootheden. Wanneer opeenvolgende [runx]-secties dezelfde invoerbestanden gebruiken, worden deze niet opnieuw ingelezen.

De database bevat enkele macro's voor het opstarten van de berekeningen, waaronder een macro die gebruikt kan worden om de berekeningen automatisch te laten verlopen bij het opstarten van de database (b.v. bij een aanroep vanuit ArisFlow).

Listing 3.1. Initialisatiebestand voor Move-berekeningen

```
[Files]
ResponseModelsDatabaseName=D:\Projekten\Graadmeters\Move\Modellen\StepBIC2\Move   4
stepBIC2.mdb
VariableTranslationFileName=D:\Projekten\Graadmeters\Berekening
Responsies\VariableTranslation.txt
BiodivFileName=D:\Projekten\Graadmeters\Berekening Responsies\biodiv.ini
LogFileName=D:\Projekten\Graadmeters\Berekening Responsies\log multirun WOConf.txt
ResultsDatabaseName=D:\Projekten\Graadmeters\Berekening         Responsies\results
multirun WOConf.mdb
[Options]
CalculateConfidenceIntervals=False
```

Listing 3.2. Voorbeeld van een VariableTranslationFileName

```
RelationVariablesId,VariableName,VariableType,ResponseVariableName,Transformation
1,Zuurgraad,Double,r,
2,Nutrienten,Double,n,
3,Grondwater,Double,f,
4,Vegetatie,FactorNum,veg[],
5,Zoutgehalte,Double,s,
6,Fgr,FactorNum,fgr[],
```

Listing 3.3. Move-vertaaltabel voor factor-variabelen

```
RelationId     VariableName   VariableCode   ResponseModelCode      CodeInInput
1              Vegetatie      Dec            1                      1
2              Vegetatie      Grp            2                      5
3              Vegetatie      Hea            3                      4
4              Vegetatie      Pin            4                      2
5              Vegetatie      Spr            5                      3
6              Fgr            Hl             1                      1
7              Fgr            Hz             2                      2
8              Fgr            Rv             3                      3
9              Fgr            Lv             4                      4
10             Fgr            Zk             5                      5
11             Fgr            Du             6                      6
12             Fgr            Az             7                      7
13             Fgr            Gg             8                      8
14             Fgr            Nz             9                      9
```

Listing 3.4. Voorbeeld van een Biodiv-initialisatiefile

```
[Algemeen]
TempDir=F:\temp\
BioMethod=EKI
SpecRefTable=
SpecColumnName=Soort_code
RefColumnName=Referentie
UnitColumnName=idcode
UseModels=1
SpecDir=
AsciiGrids=0
UnitMapName=
RefUnitMapName=
Description=
ScenarioYear=
Move3Model=1
MoveRegrTable=move_32.dbf
ButterflyRegrTable=vl_mult2.dbf
LarchRegrTable=larch.dbf
PrAbsent=0
MoveInvoerDir=0
UseVlinderMove=1
ToestandID=1
UseReferences=0
MultiRun=1
Prefix=
```

```
[Larch_instellingen]
LarchProject=nvk2
UseBarriers=1

[Directories]
LarchVegDir=D:\AF-Natuurplanner\cases\np4gevoel_grove_den\bgt_larch\
LarchBarDir=D:\AF-Natuurplanner\cases\np4gevoel_grove_den\barrieres\

[Files]
Zuurgraad=
Nutrienten=
Grondwater=
Vegetatie=
Zoutgehalte=D:\Projekten\Graadmeters                 &                  Ecologische
Modellering\Gevoeligheidsanalyse\Berekening Responsies\run1_zout2000.asc
Fgr=D:\Projekten\Graadmeters                    &                       Ecologische
Modellering\Gevoeligheidsanalyse\Berekening Responsies\run1_fgr2000.asc
PAFzm=D:\Projekten\Graadmeters                    &                     Ecologische
Modellering\Gevoeligheidsanalyse\Berekening Responsies\run1_pafzw2000.asc

[Uitvoer]
CrtOutputAscii=1
OutputFileName=
OutputDirectory=
FieldSeparator=;
DelFiles=1

[natuurtypen]
PerNatuurtype=1
AantalNT=6
NT1=bos
NT2=heide
NT3=duin
NT4=moeras
NT5=agrarisch
NT6=water

[runs]
aantal=294

[run1]
Description=2000_1
ScenarioYear=2000
SpecRefTable=D:\Projekten\Graadmeters                 &                 Ecologische
Modellering\Gevoeligheidsanalyse\Berekening Responsies\soorten 2004.mdb:soort_1
Zuurgraad=D:\Projekten\Graadmeters                  &                   Ecologische
Modellering\Gevoeligheidsanalyse\Berekening Responsies\run1_ph2000ell.asc
Nutrienten=D:\Projekten\Graadmeters                  &                  Ecologische
Modellering\Gevoeligheidsanalyse\Berekening Responsies\run1_tnituptmx2000ell.asc
Grondwater=D:\Projekten\Graadmeters                  &                  Ecologische
Modellering\Gevoeligheidsanalyse\Berekening Responsies\run1_gvg2000ell.asc
Vegetatie=D:\Projekten\Graadmeters                  &                   Ecologische
Modellering\Gevoeligheidsanalyse\Berekening Responsies\run1_bgt2000.asc
UnitMapName=D:\Projekten\Graadmeters                 &                  Ecologische
Modellering\Gevoeligheidsanalyse\Berekening
Responsies\gebieden\run1_idcode2000.asc
RefUnitMapName=D:\Projekten\Graadmeters                &                Ecologische
Modellering\Gevoeligheidsanalyse\Berekening
Responsies\gebieden\run1_idcode2000.asc

[run2]
Description=2000_2
ScenarioYear=2000
SpecRefTable=D:\Projekten\Graadmeters                 &                 Ecologische
Modellering\Gevoeligheidsanalyse\Berekening Responsies\soorten 2004.mdb:soort_2
Zuurgraad=D:\Projekten\Graadmeters                  &                   Ecologische
Modellering\Gevoeligheidsanalyse\Berekening Responsies\run1_ph2000ell.asc
Nutrienten=D:\Projekten\Graadmeters                  &                  Ecologische
Modellering\Gevoeligheidsanalyse\Berekening Responsies\run1_tnituptmx2000ell.asc
Grondwater=D:\Projekten\Graadmeters                  &                  Ecologische
Modellering\Gevoeligheidsanalyse\Berekening Responsies\run1_gvg2000ell.asc
```

```
Vegetatie=D:\Projekten\Graadmeters & Ecologische
Modellering\Gevoeligheidsanalyse\Berekening Responsies\run1_bgt2000.asc
UnitMapName=D:\Projekten\Graadmeters & Ecologische
Modellering\Gevoeligheidsanalyse\Berekening
Responsies\gebieden\run1_idcode2000.asc
RefUnitMapName=D:\Projekten\Graadmeters & Ecologische
Modellering\Gevoeligheidsanalyse\Berekening
Responsies\gebieden\run1_idcode2000.asc
```

Listing 3.5. Procedures voor het berekenen van Move-responsies

```
Option Compare Database
Option Explicit

' Version 1.1 11-8-2004
' F.G. Wortlboer, NLB/MNP/RIVM
' Calculation of responsies optimized

Dim VariableRelationsFileName As String
Dim KOVResultsFileName As String
Dim ResponseModelsDatabaseName As String
Dim ResultsDatabase As Database

Dim DoCalculateConfidenceIntervals As Integer

' Contents of ini-file
Dim strApplication As String    ' optional
Dim strVersion As String        ' optional
Dim strRemarks As String        ' optional
Dim strProgrammePathAndName As String
Dim TemporaryDirectory As String
Dim LogFileName As String
Dim IdFileName As String
Dim CharacteristicsFileName As String
Dim strBeginYear As String
Dim strResultsYears As String
Dim VariablesFileName As String
Dim ModelResultsDatabaseName As String
Dim ModelResultsFileName As String
Dim SpeciesFileName As String

Dim ResponseSetsFilename As String
Dim VariableTranslationFileName As String
Dim BiodivFileName As String

Dim ResponseResultsFileName As String

Dim ResponseSetDatabaseName As String

Dim ArisFlowControlFileName As String

Const ModelResultsTableName = "ModelResults"
Const ModelResultsAveragesTableName = "ModelResults Averages"
Const ResponseModelSetsTableName = "ResponseModelSets"
Const ResponseModelResultsTableName = "ResponseModel Results"
Const VariableRelationsTableName = "Relation Variables Model - Responsemodel"
Const AsciiGridIntegerDataTemplateTableName = "Ascii Grid Data Integer Template"
Const AsciiGridLongDataTemplateTableName = "Ascii Grid Data Long Template"
Const AsciiGridCodeDataTemplateTableName = "Ascii Grid Data Code Template"
Const AsciiGridDoubleDataTemplateTableName = "Ascii Grid Data Double Template"
Const RelationSpeciesGridTableName = "Relation Taxon - Grid"
Const UnitTableName = "GridId"

' Constant Values
Const ugPerg = 1000000# ' ug.g-1; conversion from ug to g
Const m2perHa = 10000   ' m2.ha-1; conversion from m2 to hectare
Const SecPerDay = 86400 ' s.d-1; conversion second to day
Const DayPerYear = 365  ' d.y-1; Conversion day to year
Const MoleMassNH3 = 17  ' g.mol-1; Conversion g to Mole
Const MaxWord = 10000
Const MissingValue = -9999
Const tValue005 = 2.042  ' t-value for statistical testing at 5% level
```

```
'Types
Public Type YearValues
    Year As Double
    Value As Double
End Type

Public Type YearValuesArray
    Name As String
    YValue() As YearValues
End Type

Public Type ValuePair
    X As Double
    Value As Double
End Type

Public Type ValuePairArray
    Name As String
    Value() As ValuePair
End Type

Public Type DepositionData
    Year As Integer
    SOxWet As Double
    SOxDry As Double
    NOyWet As Double
    NOyDry As Double
    NHxWet As Double
    NHxDry As Double
    NH3Conc As Double
End Type

Public Type DepositionDataArray
    SOxWet As YearValuesArray
    SOxDry As YearValuesArray
    NOyWet As YearValuesArray
    NOyDry As YearValuesArray
    NHxWet As YearValuesArray
    NHxDry As YearValuesArray
    NH3Conc As YearValuesArray
End Type

Public Type VariableStruct
    Name As String
    Type As String
    ResponseVariableName As String
    Transformation As String
    InputFileName As String
End Type

Public Type Range
    X1 As Double
    X2 As Double
End Type


Function RunJobFromStartup()
    Dim CommandLine As String
    Dim InitializationFileName As String
    Dim Result As Integer
    Dim filenum As Integer

    Result = False
    CommandLine = Command
    If InStr(CommandLine, " -ac ") > 0 Then ' arisflow control file specified
      ArisFlowControlFileName = Mid(CommandLine, InStr(CommandLine, " -ac ") + 5)
      CommandLine = Left(CommandLine, InStr(CommandLine, " -ac ") - 1)
    Else

    End If
```

```
    ' Get Name of Steering File from Command line
    InitializationFileName = Trim(CommandLine)
    Forms("Move4")!SteeringFileName = InitializationFileName
    DoEvents

    If Len(InitializationFileName) > 0 Then
      Result = RunJob(InitializationFileName)
    End If

    If Len(ArisFlowControlFileName) > 0 Then
      filenum = FreeFile
      Open ArisFlowControlFileName For Output As #filenum
      If Result = True Then
        Print #filenum, "OK"
      Else
        Print #filenum, "Errors were encountered during execution. See file '" &
LogFileName & "' for errors."
      End If
      Close #filenum
    End If

    ' Close the application
    Application.Quit

End Function


Function RunJob(IniFileName As String)
On Error GoTo Error_RunJob
    Dim filenum As Integer, filenum2 As Integer
    Dim strPathWorkDir As String
    Dim VarSet As DAO.Recordset
    Dim tmpset As Recordset, tmpstr As String
    Dim filenumModelResults As Integer
    Dim iYear As Integer
    Dim strArray(MaxWord) As String, NrOfWords As Integer
    Dim starttime As Date, endtime As Date
    Dim sql As String

    Dim ivar As Integer, nvar As Integer
    Dim NrOfSpecies As Long
    Dim ResponseResultSet As Recordset
    Dim ResponseModelName As String, VariableRelationFilename As String
    Dim SoortSet As Recordset
    Dim ResponseFilenum As Integer
    Dim kappa As Double

    Dim InitializationSuccessful As Integer
    Dim ErrorEncountered As Integer

    Dim Variables() As VariableStruct
    Dim NrOfVariables As Integer
    Dim OldVariableInputFileName() As String
    Dim NewVariableInputFileName() As String

    Dim OldUnitFileName As String
    Dim OldSpecRefFileName As String
    Dim OldTaxonCodeColumnName As String
    Dim OldGridCodeColumnName As String

    Dim Description As String
    Dim Year As Integer

    ' Biodiv parameters
    Dim DoMultiRun As Integer
    Dim NrOfRuns As Long
    Dim DoSaveAsAsciiGrid As Integer
    Dim OutputDirectory As String
    Dim SpecRefFileName As String
    Dim TaxonCodeColumnName As String
    Dim GridCodeColumnName As String
    Dim UnitFileName As String
```

```
    Dim CurrentSectionName As String

    Dim VariableHasNewValues() As Integer

    Dim StartRun As Long
    Dim irun As Long


'    Dim myws As Workspace

    InitializationSuccessful = False

    starttime = Now


    If Not ProcessIniFile(IniFileName) Then
      ' Logging of messages not yet active; just quit
      GoTo Error_RunJob
    End If

    TemporaryDirectory = GetIniFileItem(BiodivFileName, "Algemeen", "TempDir")
    If Len(TemporaryDirectory) = 0 Then
      If Len(Dir("C:\temp")) > 0 Then
        TemporaryDirectory = "C:\temp"
      Else
      End If
    Else
      If Right(TemporaryDirectory, 1) = "\" Then
        TemporaryDirectory = Left(TemporaryDirectory, Len(TemporaryDirectory) - 1)
      End If
    End If
    strPathWorkDir = TemporaryDirectory & "\"

    OutputDirectory = GetIniFileItem(BiodivFileName, "Uitvoer", "OutputDirectory")
    tmpstr = GetIniFileItem(BiodivFileName, "Uitvoer", "CrtOutputAscii")
    If tmpstr = "1" Then
      DoSaveAsAsciiGrid = True
    End If

    InitializationSuccessful = True

    filenum = FreeFile

    InitLog
    WriteLog "Read Initialization File '" & IniFileName & "' successfully"
    If Len(strApplication) > 0 Then
      WriteLog "Application: " & strApplication
    End If
    If Len(strVersion) > 0 Then
      WriteLog "Version: " & strVersion
    End If
    If Len(strRemarks) > 0 Then
      WriteLog "Remarks: " & strRemarks
    End If

    ' Create Results Database
    If Len(Dir(ModelResultsDatabaseName)) > 0 Then Kill ModelResultsDatabaseName
    CreateDatabase ModelResultsDatabaseName, dbLangGeneral
    WriteLog "Made Results Database '" & ModelResultsDatabaseName & "'
successfully"
    ' Generate Tables in results database
    'DoCmd.TransferDatabase acExport, "Microsoft Access",
ModelResultsDatabaseName, acTable, "VariableInfo Template", "VariableInfo"
    DoCmd.TransferDatabase acExport, "Microsoft Access", ModelResultsDatabaseName,
acTable, "TaxonInfo Template", "TaxonInfo"
    DoCmd.TransferDatabase acExport, "Microsoft Access", ModelResultsDatabaseName,
acTable, "ModelResults Averages Template", ModelResultsAveragesTableName
    DoCmd.TransferDatabase acExport, "Microsoft Access", ModelResultsDatabaseName,
acTable, "ResponseModelSets Template", ResponseModelSetsTableName
```

```
    DoCmd.TransferDatabase acExport, "Microsoft Access", ModelResultsDatabaseName,
acTable, "ResponseModel Results Template", ResponseModelResultsTableName
    WriteLog "Tables constructed"


    ' Link newly formed tables to this database
    RefreshLink "TaxonInfo", ModelResultsDatabaseName, "TaxonInfo"
    RefreshLink ModelResultsAveragesTableName, ModelResultsDatabaseName,
ModelResultsAveragesTableName
    RefreshLink ResponseModelSetsTableName, ModelResultsDatabaseName,
ResponseModelSetsTableName
    RefreshLink ResponseModelResultsTableName, ModelResultsDatabaseName,
ResponseModelResultsTableName
    WriteLog "Links refreshed"

    CurrentDb.TableDefs.Refresh
    DBEngine.Idle dbRefreshCache

    ' Get inputs, store them in tables
'    If LoadTable(ResponseModelSetsTableName, ResponseSetsFilename) Then
'      WriteLog "Loading file '" & ResponseSetsFilename & "' successful."
'    Else
'      WriteLog "Loading file '" & ResponseSetsFilename & "' failed."
'      GoTo Error_RunJob
'    End If
    ' Check existence of relation files
'    If Len(Dir(VariableTranslationFileName)) = 0 Then
'      WriteLog "File '" & VariableTranslationFileName & "' does not exist"
'      GoTo Error_RunJob
'    End If

    ' Calculate Responsemodels for all lakes based upon yearly average data
    ' Link to response model database
    Set ResponseResultSet = CurrentDb.OpenRecordset(ResponseModelResultsTableName)


      RefreshLink "ResponseModel ModelInfo", ResponseModelsDatabaseName,
"ModelInfo"
      RefreshLink "ResponseModel Coefficients", ResponseModelsDatabaseName,
"Coefficienten"
      RefreshLink "ResponseModel Taxa", ResponseModelsDatabaseName, "Taxa"
      RefreshLink "ResponseModel Variables", ResponseModelsDatabaseName,
"Variabelen"
      RefreshLink "ResponseModel Statistics", ResponseModelsDatabaseName,
"Statistiek"
      RefreshLink "ResponseModel Correlations", ResponseModelsDatabaseName,
"Correlaties"
      RefreshLink "ResponseModel Covariances", ResponseModelsDatabaseName,
"Covarianties"


    Set tmpset = CurrentDb.OpenRecordset("ResponseModel ModelInfo")
    ResponseModelName = tmpset!ModelNaam
    tmpset.Close

    SysCmd acSysCmdInitMeter, "Calculating response models from set '" &
ResponseModelName & "': ", 1
    DoEvents
    WriteLog "Calculating response models from set '" & ResponseModelName & "'
from database '" & ResponseModelsDatabaseName & "'"

      ' Load file with relations between model variables and response model
variables
      DoCmd.TransferDatabase acExport, "Microsoft Access",
ModelResultsDatabaseName, acTable, VariableRelationsTableName & " Template",
VariableRelationsTableName
      RefreshLink VariableRelationsTableName, ModelResultsDatabaseName,
VariableRelationsTableName
      On Error GoTo Error_RunJob
      If Not LoadTable(VariableRelationsTableName, VariableTranslationFileName)
Then
        WriteLog "Loading file '" & VariableTranslationFileName & "' failed."
```

```
        GoTo Error_RunJob
      End If


      ' process Biodiv-file
      ErrorEncountered = False
      ' Multirun
      If GetIniFileItem(BiodivFileName, "Algemeen", "Multirun") = 1 Then
        DoMultiRun = True
        tmpstr = GetIniFileItem(BiodivFileName, "Runs", "Aantal")
        If tmpstr = "" Then 'no value given
          WriteLog "No value for number of runs specified while using Multirun
option"
          ErrorEncountered = True
        Else
          NrOfRuns = CInt(tmpstr)
          WriteLog "MultiRun = True; NrOfRuns=" & NrOfRuns
        End If
      Else  ' only one run
        DoMultiRun = False
        NrOfRuns = 0   ' do just one (1) run
        WriteLog "MultiRun = False"
      End If

      ' Input variables
      sql = "Select * FROM  [" & VariableRelationsTableName & "];"
      Set VarSet = CurrentDb.OpenRecordset(sql)
      NrOfVariables = AantalRecords(VarSet)
      ReDim Variables(NrOfVariables)
      ReDim OldVariableInputFileName(NrOfVariables)
      ReDim NewVariableInputFileName(NrOfVariables)
      ReDim VariableHasNewValues(NrOfVariables)
      ivar = 0
      While Not VarSet.EOF
        ivar = ivar + 1
        Variables(ivar).Name = VarSet!VariableName
        Variables(ivar).Type = VarSet!VariableType
        Variables(ivar).ResponseVariableName = VarSet!ResponseVariableName
        Variables(ivar).Transformation  =  IIf(Not  IsNull(VarSet!Transformation),
VarSet!Transformation, "")
        VarSet.MoveNext
      Wend
      VarSet.Close

      If ErrorEncountered Then
        GoTo Error_RunJob
      End If

      'Maak TaxonInfo
      sql = "INSERT INTO [TaxonInfo] (TaxonNr,TaxonCode,TaxonName) " & _
            "SELECT TaxonNr,TaxonCode,TaxonNaam " & _
            "FROM [ResponseModel Taxa];"
      CurrentDb.Execute sql
      DoEvents


      ' Get general inputs
      Year = -9999

      If DoMultiRun And DoCalculateConfidenceIntervals Then
        SysCmd  acSysCmdInitMeter,  "Performing  calculations  ("  &  NrOfRuns  &  "
runs): ", NrOfRuns
        DoEvents
      End If
      For irun = 0 To NrOfRuns
        WriteLog "Starting Run " & irun & ": " & Now()

        ' Specify section to look in
        If irun = 0 Then  ' Get general inputs
          CurrentSectionName = "Algemeen"
        Else
          CurrentSectionName = "run" & irun
```

```
        End If

        ' Get Description
        tmpstr = GetIniFileItem(BiodivFileName, CurrentSectionName, "Description")
        If Len(tmpstr) > 0 Then
          Description = tmpstr
        End If
        If Len(Description) = 0 Then
          If Not DoMultiRun Or (DoMultiRun And irun > 0) Then   ' Description MUST
be given
            WriteLog "No Description specified"
            ErrorEncountered = True
          End If
        End If

        ' Get Year
        tmpstr = GetIniFileItem(BiodivFileName, CurrentSectionName,
"ScenarioYear")
        If Len(tmpstr) > 0 Then
          Year = CInt(tmpstr)
        End If
        If Year < 0 Then
          If Not DoMultiRun Or (DoMultiRun And irun > 0) Then   ' ScenarioYear
MUST be given
            WriteLog "No ScenarioYear specified"
            ErrorEncountered = True
          End If
        End If

        ' Get gridid's
        tmpstr = GetIniFileItem(BiodivFileName, CurrentSectionName, "UnitMapName")
        If Len(tmpstr) > 0 Then
          UnitFileName = tmpstr
        End If
        If Len(UnitFileName) = 0 Then
          If Not DoMultiRun Or (DoMultiRun And irun > 0) Then
            WriteLog ("No UnitFileName specified in section [" &
CurrentSectionName & "]")
            ErrorEncountered = True
          End If
        Else
          If UnitFileName <> OldUnitFileName Then
            If InStr(UnitFileName, ".mdb:") > 0 Then 'table in Access-database
              DoCmd.TransferDatabase acLink, , Left(UnitFileName,
InStr(UnitFileName, ":") - 1), acTable, Mid(UnitFileName, InStr(UnitFileName, ":")
+ 1), UnitTableName, False
            Else
              LoadAsciiGridData UnitTableName, AsciiGridLongDataTemplateTableName,
UnitFileName
            End If
          End If
        End If

        ' Get relation between species and grids for which response need to be
calculated
        tmpstr = GetIniFileItem(BiodivFileName, CurrentSectionName,
"SpecRefTable")
        If Len(tmpstr) > 0 Then
          SpecRefFileName = tmpstr
        End If
        If Len(SpecRefFileName) = 0 Then
          If Not DoMultiRun Or (DoMultiRun And irun > 0) Then
            WriteLog ("No SpecRefFileName specified in section [" &
CurrentSectionName & "]")
            ErrorEncountered = True
          End If
        End If
        tmpstr = GetIniFileItem(BiodivFileName, CurrentSectionName,
"SpecColumnName")
        If Len(tmpstr) > 0 Then
          TaxonCodeColumnName = tmpstr
        End If
```

```
        If Len(TaxonCodeColumnName) = 0 Then
          If Not DoMultiRun Or (DoMultiRun And irun > 0) Then
            WriteLog ("No TaxonCodeColumnName specified in section [" &
CurrentSectionName & "]")
            ErrorEncountered = True
          End If
        End If
        tmpstr = GetIniFileItem(BiodivFileName, CurrentSectionName,
"UnitColumnName")
        If Len(tmpstr) > 0 Then
          GridCodeColumnName = tmpstr
        End If
        If Len(GridCodeColumnName) = 0 Then
          If Not DoMultiRun Or (DoMultiRun And irun > 0) Then
            WriteLog ("No GridCodeColumnName specified in section [" &
CurrentSectionName & "]")
            ErrorEncountered = True
          End If
        End If
        If Len(SpecRefFileName) > 0 And (SpecRefFileName <> OldSpecRefFileName Or
TaxonCodeColumnName <> OldTaxonCodeColumnName Or GridCodeColumnName <>
OldGridCodeColumnName) Then
          LoadRelationTaxonGridData RelationSpeciesGridTableName,
RelationSpeciesGridTableName & " Template", SpecRefFileName, TaxonCodeColumnName,
GridCodeColumnName
        End If

        ' Specify section to look for files
        If irun = 0 Then  ' Get general inputs
          CurrentSectionName = "Files"
        Else
          CurrentSectionName = "run" & irun
        End If
        ' Get new input filenames
        For ivar = 1 To NrOfVariables
          tmpstr = GetIniFileItem(BiodivFileName, CurrentSectionName,
Variables(ivar).Name)
          If Len(tmpstr) > 0 Then
            Variables(ivar).InputFileName = tmpstr
          End If
          If Variables(ivar).InputFileName = "" Then ' Files MUST be specified
            If Not DoMultiRun Or (DoMultiRun And irun > 0) Then
              WriteLog "Input file for variable '" & Variables(ivar).Name & "' not
specified."
              ErrorEncountered = True
            End If
          ElseIf Len(Dir(Variables(ivar).InputFileName)) = 0 Then  ' Files MUST be
present
            If Not DoMultiRun Or (DoMultiRun And irun > 0) Then
              WriteLog "Input file '" & Variables(ivar).InputFileName & "' for
variable '" & Variables(ivar).Name & "' not found."
              ErrorEncountered = True
            End If
          End If
        Next ivar

        ' Get input values
        For ivar = 1 To NrOfVariables
          If Variables(ivar).InputFileName <> OldVariableInputFileName(ivar) Then
            Select Case Variables(ivar).Type
              Case "Integer": LoadAsciiGridData Variables(ivar).Name,
AsciiGridIntegerDataTemplateTableName, Variables(ivar).InputFileName
              Case "Long": LoadAsciiGridData Variables(ivar).Name,
AsciiGridLongDataTemplateTableName, Variables(ivar).InputFileName
              Case "Code": LoadAsciiGridData Variables(ivar).Name,
AsciiGridCodeDataTemplateTableName, Variables(ivar).InputFileName
              Case "Double": LoadAsciiGridData Variables(ivar).Name,
AsciiGridDoubleDataTemplateTableName, Variables(ivar).InputFileName
              Case "FactorNum": LoadAsciiGridData Variables(ivar).Name,
AsciiGridIntegerDataTemplateTableName, Variables(ivar).InputFileName
              Case "FactorTxt": LoadAsciiGridData Variables(ivar).Name,
AsciiGridCodeDataTemplateTableName, Variables(ivar).InputFileName
```

```
              Case Else
                   WriteLog ("Unexpected type of Model Variable '" &
Variables(ivar).Name & "': " & Variables(ivar).Type)
                   ErrorEncountered = True
            End Select
            VariableHasNewValues(ivar) = True
          Else
            VariableHasNewValues(ivar) = False
          End If
          DoEvents
        Next ivar

        If ErrorEncountered Then
          GoTo Error_RunJob
        End If

        SysCmd acSysCmdInitMeter, "Performing calculations (" & NrOfRuns & "
runs): ", NrOfRuns
        SysCmd acSysCmdUpdateMeter, irun
        DoEvents


        ' if DoMultiRun then irun=0 is used for gathering common inputs
        If Not DoMultiRun Or (DoMultiRun And irun > 0) Then

          ' Add model results
          'EraseTable ModelResultsAveragesTableName
          For ivar = 1 To NrOfVariables
            If VariableHasNewValues(ivar) Then
              If InStr(Variables(ivar).ResponseVariableName, "[") = 0 Then
                sql = "DELETE [" & ModelResultsAveragesTableName & "].*,
[ResponseModel Variables].VariabeleNaam " & _
                      "FROM [" & ModelResultsAveragesTableName & "] INNER JOIN
[ResponseModel Variables] ON [" & ModelResultsAveragesTableName & "].VariableNr =
[ResponseModel Variables].VariabeleNr " & _
                      "WHERE ((([ResponseModel Variables].VariabeleNaam)='" &
Variables(ivar).Name & "'));"
                CurrentDb.Execute sql
                sql = "INSERT INTO [" & ModelResultsAveragesTableName & "] (
GridId, [Value], [Year], VariableNr ) " & _
                      "SELECT [" & UnitTableName & "].Value AS GridId, [" &
Variables(ivar).Name & "].Value, " & Year & " AS [Year], [ResponseModel
Variables].VariabeleNr " & _
                      "FROM GridId INNER JOIN [" & Variables(ivar).Name & "] ON
([" & UnitTableName & "].Y = [" & Variables(ivar).Name & "].Y) AND ([" &
UnitTableName & "].X = [" & Variables(ivar).Name & "].X), [" &
VariableRelationsTableName & "] INNER JOIN [ResponseModel Variables] ON [" &
VariableRelationsTableName & "].ResponseVariableName = [ResponseModel
Variables].VariabeleNaam " & _
                      "WHERE ((([" & VariableRelationsTableName & "].VariableName)
= '" & Variables(ivar).Name & "'));"
                CurrentDb.Execute sql
              Else  ' factor
                sql = "DELETE [ResponseModel Variables].VariabeleNaam, [" &
ModelResultsAveragesTableName & "].*, [" & ModelResultsAveragesTableName &
"].VariableNr " & _
                      "FROM [ResponseModel Variables] INNER JOIN [" &
ModelResultsAveragesTableName & "] ON [ResponseModel Variables].VariabeleNr = [" &
ModelResultsAveragesTableName & "].VariableNr " & _
                      "WHERE ((([ResponseModel Variables].VariabeleNaam) Like '" &
Left(Variables(ivar).ResponseVariableName,
InStr(Variables(ivar).ResponseVariableName, "[") - 1) & "*') AND (([" &
ModelResultsAveragesTableName & "].VariableNr)=[ResponseModel
Variables].[VariabeleNr]));"
                CurrentDb.Execute sql
                sql = "INSERT INTO [" & ModelResultsAveragesTableName & "] (
GridId, [Year], VariableNr, [Value] ) " & _
                      "SELECT [" & UnitTableName & "].Value AS GridId, " & Year &
" AS [Year], [ResponseModel Variables].VariabeleNr, 1 AS [Value] " & _
```

```
                      "FROM [ResponseModel Variables], (GridId INNER JOIN [" &
Variables(ivar).Name & "] ON ([" & UnitTableName & "].Y = [" &
Variables(ivar).Name & "].Y) AND ([" & UnitTableName & "].X = [" &
Variables(ivar).Name & "].X)) INNER JOIN [Relation Variables Coding Model -
Responsemodel] ON [" & Variables(ivar).Name & "].Value = [Relation Variables
Coding Model - Responsemodel].CodeInInput " & _
                      "WHERE ((([Relation Variables Coding Model -
Responsemodel].VariableName) = '" & Variables(ivar).Name & "') And
(([ResponseModel Variables].VariabeleNaam) = '" &
Left(Variables(ivar).ResponseVariableName,
InStr(Variables(ivar).ResponseVariableName, "[") - 1) & "'&[ResponseModelCode]));"
                CurrentDb.Execute sql
              End If
            End If
          Next ivar

          ' Calculate responses
          If Not DoCalculateConfidenceIntervals Then
            CalculateResponsesResponseSetWOConf ResponseModelName, Year, irun
            'CalculateResponsesResponseSetWOConfPerSoort ResponseModelName, Year,
irun
          Else
            CalculateResponsesResponseSetWConf ResponseModelName, Year, irun
          End If
        End If


        ' switch new to old file names
        For ivar = 1 To NrOfVariables
          OldVariableInputFileName(ivar) = Variables(ivar).InputFileName
        Next ivar
        OldUnitFileName = UnitFileName
        OldSpecRefFileName = SpecRefFileName
        OldTaxonCodeColumnName = TaxonCodeColumnName
        OldGridCodeColumnName = GridCodeColumnName

        DoEvents

        WriteLog "End Run " & irun & ": " & Now()

        If DoMultiRun And DoCalculateConfidenceIntervals Then
          SysCmd acSysCmdUpdateMeter, irun
          DoEvents
        End If

      Next irun

      RunJob = True
      If DoMultiRun And DoCalculateConfidenceIntervals Then
        SysCmd acSysCmdRemoveMeter
        DoEvents
      End If

'       SysCmd acSysCmdRemoveMeter
      DoEvents

    ResponseResultSet.Close


End_RunJob:
    ' Clean-up


    ' Remove links to specific inputs
    For ivar = 1 To NrOfVariables
      DeleteLink Variables(ivar).Name
    Next ivar
    DeleteLink RelationSpeciesGridTableName
    DeleteLink UnitTableName
```

```
    DeleteLink "ResponseModel ModelInfo"
    DeleteLink "ResponseModel Coefficients"
    DeleteLink "ResponseModel Taxa"
    DeleteLink "ResponseModel Variables"
    DeleteLink "ResponseModel Statistics"
    DeleteLink "ResponseModel Correlations"
    DeleteLink "ResponseModel Covariances"

    SysCmd acSysCmdRemoveMeter
    DoEvents

    ' Close the job
    ' Delete links to newly formed tables to this database
    DeleteLink "TaxonInfo"
    DeleteLink ModelResultsTableName
    DeleteLink ModelResultsAveragesTableName
    DeleteLink ResponseModelSetsTableName
    DeleteLink ResponseModelResultsTableName
    DoCmd.DeleteObject acTable, VariableRelationsTableName

    ' Delete Temporary Files
    On Error Resume Next

    endtime = Now

    WriteLog "Program finished successfully. StartTime=" & starttime & ";
EndTime=" & endtime & "; Elapsed: " & DateDiff("s", starttime, endtime) & "
seconds."

    EndLog
    Exit Function

Error_RunJob:
    If InitializationSuccessful = True Then
      WriteLog "Error in function RunJob: Err=" & Err & ": " & Err.Description
      WriteLog "Program ended with errors."
    Else
      MsgBox "Error in Move4 Control Module (before correct initialization): " &
Err.Description
    End If
    RunJob = False
    Resume End_RunJob
End Function

Function CalculateResponsesResponseSetWOConf(ResponsModelName As String, Year As
Integer, Run As Long)

    Dim sql As String, MyQD As QueryDef

    ' For all Grids and all species
    sql = "SELECT [" & ModelResultsAveragesTableName & "].GridId, [ResponseModel
Coefficients].TaxonNr, [ResponseModel Coefficients].Volgorde, [ResponseModel
Coefficients].TermType, [ResponseModel Coefficients].VarNrA, [ResponseModel
Variables].VariabeleCode, [ResponseModel Coefficients].VarNrB, [ResponseModel
Variables_1].VariabeleCode, [ResponseModel Coefficients].Coeff, [ResponseModel
Variables].VariabeleTransformatie AS TransA, [" & ModelResultsAveragesTableName &
"].Value, " & _
          "IIf([TransA]='Log(X)',Log([" & ModelResultsAveragesTableName &
"].[Value])/Log(10),IIf([TransA]='Log(X+1)',Log([" & ModelResultsAveragesTableName
& "].[Value]+1)/Log(10),[" & ModelResultsAveragesTableName & "].[Value])) AS ValA,
[ResponseModel Variables_1].VariabeleTransformatie AS TransB, " & _
          "IIf([TransB]='Log(X)',Log([ModelResults
Averages_1].[Value])/Log(10),IIf([TransB]='Log(X+1)',Log([ModelResults
Averages_1].[Value]+1)/Log(10),[ModelResults Averages_1].[Value])) AS ValB, " & _

"IIf([TermType]='cA',[ValA],IIf([TermType]='cA2',[ValA]*[ValA],IIf([TermType]='cAB
',[ValA]*[ValB],IIf([TermType]='cA2B',[ValA]*[ValA]*[ValB],IIf([TermType]='cAB2',[
ValA]*[ValB]*[ValB],Null))))) AS Val, " & _
          "IIf(IsNull([" & ModelResultsAveragesTableName &
"].[GridId]),[Coeff],[Coeff]*[Val]) AS W " & _
```

```
          "FROM ((((["  & ModelResultsAveragesTableName & "] INNER JOIN
[ResponseModel Coefficients] ON [" & ModelResultsAveragesTableName & "].VariableNr
= [ResponseModel Coefficients].VarNrA) LEFT JOIN [" &
ModelResultsAveragesTableName & "] AS [ModelResults Averages_1] ON [ResponseModel
Coefficients].VarNrB = [ModelResults Averages_1].VariableNr) INNER JOIN
[ResponseModel Variables] ON [ResponseModel Coefficients].VarNrA = [ResponseModel
Variables].VariabeleNr) LEFT JOIN [ResponseModel Variables] AS [ResponseModel
Variables_1] ON [ResponseModel Coefficients].VarNrB = [ResponseModel
Variables_1].VariabeleNr) INNER JOIN [" & RelationSpeciesGridTableName & "] ON (["
& RelationSpeciesGridTableName & "].TaxonNr = [ResponseModel
Coefficients].TaxonNr) AND ([" & ModelResultsAveragesTableName & "].GridId = [" &
RelationSpeciesGridTableName & "].GridId) " & _
          "WHERE ( (([ModelResults Averages_1].GridId)=[" &
ModelResultsAveragesTableName & "].[GridId] Or ([ModelResults Averages_1].GridId)
Is Null) );"
    On Error GoTo 0
    Set MyQD = CurrentDb.QueryDefs("tmpQuery")
    MyQD.sql = sql
    MyQD.Close

    sql = "SELECT tmpQuery.TaxonNr, tmpQuery.GridId, Sum(tmpQuery.W) AS SumOfW,
[ResponseModel Coefficients].Coeff AS Intercept, [Intercept]+[SumOfW] AS WTot,
Exp([WTot])/(1+Exp([WTot])) AS KOV, IIf([KOV]>=[KOVOpt],1,0) AS PresentKOV " & _
          "FROM (tmpQuery INNER JOIN [ResponseModel Coefficients] ON
tmpQuery.TaxonNr = [ResponseModel Coefficients].TaxonNr) INNER JOIN [ResponseModel
Statistics] ON [ResponseModel Coefficients].TaxonNr = [ResponseModel
Statistics].TaxonNr " & _
          "GROUP BY tmpQuery.TaxonNr, tmpQuery.GridId, [ResponseModel
Coefficients].Coeff, [ResponseModel Coefficients].TermType, [ResponseModel
Coefficients].Coeff, [ResponseModel Statistics].KOVOpt " & _
          "HAVING ((([ResponseModel Coefficients].TermType)='c'));"
    On Error GoTo 0
    Set MyQD = CurrentDb.QueryDefs("tmpSumQuery")
    MyQD.sql = sql
    MyQD.Close

    sql = "INSERT INTO [" & ResponseModelResultsTableName & "] ( Run, Year,
TaxonNr, GridId, KOV, PresentKOV ) " & _
          "SELECT " & Run & " AS Run, " & Year & " AS Year, TaxonNr, GridId, KOV,
PresentKOV " & _
          "FROM tmpSumQuery;"
    CurrentDb.Execute sql

End Function

Function CalculateResponsesResponseSetWOConfPerSoort(ResponsModelName As String,
Year As Integer, Run As Long)

      Dim sql As String
      Dim SoortSet As Recordset, NrOfSpecies As Long

      sql = "SELECT [ResponseModel Taxa].TaxonNr, [ResponseModel Taxa].TaxonCode,
[ResponseModel Taxa].TaxonNaam " & _
            "FROM ([ResponseModel Taxa] INNER JOIN [ResponseModel Statistics] ON
[ResponseModel Taxa].TaxonCode = [ResponseModel Statistics].Soort) INNER JOIN [" &
RelationSpeciesGridTableName & "] ON [ResponseModel Taxa].TaxonNr = [" &
RelationSpeciesGridTableName & "].TaxonNr " & _
            "GROUP BY [ResponseModel Taxa].TaxonNr, [ResponseModel
Taxa].TaxonCode, [ResponseModel Taxa].TaxonNaam " & _
            "ORDER BY [ResponseModel Taxa].TaxonNr;"

      Set SoortSet = CurrentDb.OpenRecordset(sql)
      NrOfSpecies = NrOfRecords(SoortSet)
      SysCmd acSysCmdInitMeter, "Calculating response models from set '" &
ResponsModelName & "': ", NrOfSpecies
      DoEvents
      While Not SoortSet.EOF
        CalculateResponsesResponseSetSoort SoortSet!TaxonNr, Year, False, Run
        SoortSet.MoveNext
        SysCmd acSysCmdUpdateMeter, SoortSet.AbsolutePosition + 1
```

```
        DoEvents
      Wend

      SoortSet.Close
      SysCmd acSysCmdRemoveMeter

End Function

Function CalculateResponsesResponseSetWConf(ResponsModelName As String, Year As
Integer, Run As Long)

      Dim sql As String
      Dim SoortSet As Recordset, NrOfSpecies As Long

      sql = "SELECT [ResponseModel Taxa].TaxonNr, [ResponseModel Taxa].TaxonCode,
[ResponseModel Taxa].TaxonNaam " & _
            "FROM ([ResponseModel Taxa] INNER JOIN [ResponseModel Statistics] ON
[ResponseModel Taxa].TaxonCode = [ResponseModel Statistics].Soort) INNER JOIN [" &
RelationSpeciesGridTableName & "] ON [ResponseModel Taxa].TaxonNr = [" &
RelationSpeciesGridTableName & "].TaxonNr " & _
            "GROUP BY [ResponseModel Taxa].TaxonNr, [ResponseModel
Taxa].TaxonCode, [ResponseModel Taxa].TaxonNaam " & _
            "ORDER BY [ResponseModel Taxa].TaxonNr;"

      Set SoortSet = CurrentDb.OpenRecordset(sql)
      NrOfSpecies = NrOfRecords(SoortSet)
      SysCmd acSysCmdInitMeter, "Calculating response models from set '" &
ResponsModelName & "': ", NrOfSpecies
      DoEvents
      While Not SoortSet.EOF

        'CalculateResponsesResponseSetSoortWConf SoortSet!TaxonNr, Year, Run
        CalculateResponsesResponseSetSoort SoortSet!TaxonNr, Year, True, Run
        SoortSet.MoveNext
        SysCmd acSysCmdUpdateMeter, SoortSet.AbsolutePosition + 1
        DoEvents
      Wend
      SoortSet.Close
      SysCmd acSysCmdRemoveMeter

End Function


Function CalculateResponsesResponseSetSoortWConf(TaxonNr As Long, Year As Integer,
Run As Long)
On Error GoTo Error_CalculateResponsesResponseSetSoortWConf
    Dim sql As String
    Dim MyQD As QueryDef

    sql = "SELECT [" & ModelResultsAveragesTableName & "].GridId, [ResponseModel
Coefficients].TaxonNr, [ResponseModel Coefficients].Volgorde, [ResponseModel
Coefficients].TermType, [ResponseModel Coefficients].VarNrA, [ResponseModel
Variables].VariabeleCode, [ResponseModel Coefficients].VarNrB, [ResponseModel
Variables_1].VariabeleCode, [ResponseModel Coefficients].Coeff, [ResponseModel
Variables].VariabeleTransformatie AS TransA, [" & ModelResultsAveragesTableName &
"].Value, " & _
          "IIf([TransA]='Log(X)',Log([" & ModelResultsAveragesTableName &
"].[Value])/Log(10),IIf([TransA]='Log(X+1)',Log([" & ModelResultsAveragesTableName
& "].[Value]+1)/Log(10),[" & ModelResultsAveragesTableName & "].[Value])) AS ValA,
" & _
          "[ResponseModel Variables_1].VariabeleTransformatie AS TransB,
IIf(IsNull([TransB]) And [VarNrB]>0,0,IIf([TransB]='Log(X)',Log([ModelResults
Averages_1].[Value])/Log(10),IIf([TransB]='Log(X+1)',Log([ModelResults
Averages_1].[Value]+1)/Log(10),[ModelResults Averages_1].[Value]))) AS ValB, " & _

"IIf([TermType]='cA',[ValA],IIf([TermType]='cA2',[ValA]*[ValA],IIf([TermType]='cAB
',[ValA]*[ValB],IIf([TermType]='cA2B
',[ValA]*[ValA]*[ValB],IIf([TermType]='cAB2',[ValA]*[ValB]*[ValB],Null))))) AS
Val, [Coeff]*[Val] AS W " & _
```

```
          "FROM ((((([" & ModelResultsAveragesTableName & "] INNER JOIN
[ResponseModel Coefficients] ON [" & ModelResultsAveragesTableName & "].VariableNr
= [ResponseModel Coefficients].VarNrA) LEFT JOIN [" &
ModelResultsAveragesTableName & "] AS [ModelResults Averages_1] ON [ResponseModel
Coefficients].VarNrB = [ModelResults Averages_1].VariableNr) INNER JOIN
[ResponseModel Variables] ON [ResponseModel Coefficients].VarNrA = [ResponseModel
Variables].VariabeleNr) LEFT JOIN [ResponseModel Variables] AS [ResponseModel
Variables_1] ON [ResponseModel Coefficients].VarNrB = [ResponseModel
Variables_1].VariabeleNr) INNER JOIN [" & RelationSpeciesGridTableName & "] ON (["
& ModelResultsAveragesTableName & "].GridId = [" & RelationSpeciesGridTableName &
"].GridId) AND ([ResponseModel Coefficients].TaxonNr = [" &
RelationSpeciesGridTableName & "].TaxonNr) " & _
          "WHERE ( (([ResponseModel Coefficients].TaxonNr)=" & TaxonNr & ") AND
(([ModelResults Averages_1].GridId)=[" & ModelResultsAveragesTableName &
"].[GridId] Or ([ModelResults Averages_1].GridId) Is Null));"
    Set MyQD = CurrentDb.QueryDefs("tmpQuery")
    MyQD.sql = sql
    MyQD.Close

    sql = "SELECT tmpQuery.TaxonNr, tmpQuery.GridId, Sum(tmpQuery.W) AS SumOfW,
[ResponseModel Coefficients].Coeff AS Intercept, [Intercept]+[SumOfW] AS WTot,
Exp([WTot])/(1+Exp([WTot])) AS KOV, [ResponseModel Statistics].KOVOpt,
IIf([KOV]>=[KOVOpt],1,0) AS PresentKOV " & _
          "FROM (tmpQuery INNER JOIN [ResponseModel Coefficients] ON
tmpQuery.TaxonNr = [ResponseModel Coefficients].TaxonNr) INNER JOIN [ResponseModel
Statistics] ON [ResponseModel Coefficients].TaxonNr = [ResponseModel
Statistics].TaxonNr " & _
          "GROUP BY tmpQuery.TaxonNr, tmpQuery.GridId, [ResponseModel
Coefficients].Coeff, [ResponseModel Statistics].KOVOpt, [ResponseModel
Coefficients].TermType, [ResponseModel Coefficients].Coeff " & _
          "HAVING ( ((tmpQuery.TaxonNr)=" & TaxonNr & ") AND (([ResponseModel
Coefficients].TermType)='c') );"
    Set MyQD = CurrentDb.QueryDefs("tmpSumQuery")
    MyQD.sql = sql
    MyQD.Close

    sql = "SELECT TaxonNr, [Select Values Intercept].GridId, [Select Values
Intercept].Volgorde, [Select Values Intercept].Val " & _
          "FROM [Select Values Intercept] " & _
          "WHERE [TaxonNr] = " & TaxonNr & " " & _
          "UNION SELECT TaxonNr,GridId, Volgorde, Val " & _
          "FROM [Select Values TmpQuery] " & _
          "WHERE [TaxonNr] = " & TaxonNr & ";"
    Set MyQD = CurrentDb.QueryDefs("Select Values of Terms")
    MyQD.sql = sql
    MyQD.Close

    sql = "SELECT [ResponseModel Covariances].TaxonNr, [ResponseModel
Covariances].CoeffNr1 AS Coef1, [ResponseModel Covariances].CoeffNr2 AS Coef2,
[ResponseModel Covariances].Waarde " & _
          "FROM [ResponseModel Covariances] " & _
          "WHERE (TaxonNr = " & TaxonNr & ") " & _
          "UNION " & _
          "SELECT [ResponseModel Covariances].TaxonNr, [ResponseModel
Covariances].CoeffNr2 AS Coef1, [ResponseModel Covariances].CoeffNr1 AS Coef2,
[ResponseModel Covariances].Waarde " & _
          "FROM [ResponseModel Covariances] " & _
          "WHERE (TaxonNr=" & TaxonNr & ");"
    Set MyQD = CurrentDb.QueryDefs("Select Covariances")
    MyQD.sql = sql
    MyQD.Close

    sql = "SELECT [Select Values of Terms].TaxonNr, [Select Values of
Terms].GridId, Sum([Select Values of Terms].[Val]*[Select Values of
Terms_1].[Val]*[Waarde]) AS SumOfProduct " & _
          "FROM ([Select Values of Terms] INNER JOIN [Select Covariances] ON
([Select Values of Terms].Volgorde = [Select Covariances].Coef1) AND ([Select
Values of Terms].TaxonNr = [Select Covariances].TaxonNr)) INNER JOIN [Select
Values of Terms] AS [Select Values of Terms_1] ON ([Select Values of Terms].GridId
= [Select Values of Terms_1].GridId) AND ([Select Covariances].Coef2 = [Select
Values of Terms_1].Volgorde) AND ([Select Covariances].TaxonNr = [Select Values of
Terms_1].TaxonNr) " & _
```

```
          "GROUP BY [Select Values of Terms].TaxonNr, [Select Values of
Terms].GridId " & _
          "HAVING (((([Select Values of Terms].TaxonNr)=" & TaxonNr & "));"
    Set MyQD = CurrentDb.QueryDefs("Select Sum Product Covariances")
    MyQD.sql = sql
    MyQD.Close

    sql = "SELECT [Select Sum Product Covariances].TaxonNr, [Select Sum Product
Covariances].GridId, tmpSumQuery.KOV, tmpSumQuery.KOVOpt, [Select Sum Product
Covariances].SumOfProduct AS SumTotaal, [SumTotaal]^0.5 AS SELogit, " & tValue005
& "*[SELogit]^0.5 AS BetrouwbaarheidsintervalLogit, [WTot]-
[BetrouwbaarheidsintervalLogit] AS WTot_L1, [WTot]+[BetrouwbaarheidsintervalLogit]
AS WTot_L2, Exp([Wtot_l1])/(1+Exp([wtot_l1])) AS KOV_L1,
Exp([WTot_L2])/(1+Exp([WTot_L2])) AS KOV_L2, tmpSumQuery.PresentKOV,
([KOV_L1]>=[KOVOpt])*-1 AS PresentL1, ([KOV_L2]>=[KOVOpt])*-1 AS PresentL2 " & _
          "FROM [Select Sum Product Covariances] INNER JOIN tmpSumQuery ON
([Select Sum Product Covariances].TaxonNr = tmpSumQuery.TaxonNr) AND ([Select Sum
Product Covariances].GridId = tmpSumQuery.GridId) " & _
          "WHERE ((([Select Sum Product Covariances].TaxonNr)=" & TaxonNr & "));"
    Set MyQD = CurrentDb.QueryDefs("Calculate Confidence Statistics")
    MyQD.sql = sql
    MyQD.Close

    sql = "INSERT INTO [" & ResponseModelResultsTableName & "] ( Run, [Year],
TaxonNr, GridId, KOV, L1, L2, PresentKOV, PresentL1, PresentL2 ) " & _
          "SELECT " & Run & " AS Run, " & Year & " AS [Year], TaxonNr, GridId,
KOV, KOV_L1, KOV_L2, PresentKOV, PresentL1, PresentL2 " & _
          "FROM [Calculate Confidence Statistics];"
    DoEvents
    CurrentDb.Execute sql
    DoEvents

End_CalculateResponsesResponseSetSoortWConf:
    Exit Function

Error_CalculateResponsesResponseSetSoortWConf:
    WriteLog "Error in function CalculateResponsesResponseSetSoortWConf(" &
TaxonNr & "): Err=" & Err & ": " & Err.Description
    Resume End_CalculateResponsesResponseSetSoortWConf

End Function

Function CalculateResponsesResponseSetSoort(TaxonNr As Long, Year As Integer,
DoCalculateConfidenceIntervals As Integer, Run As Long)
On Error GoTo Error_CalculateResponsesResponseSetSoort
    Dim sql As String
    Dim tmpset As DAO.Recordset
    Dim CovMatrix() As Double
    Dim CoefSet As DAO.Recordset
    Dim CoefAantal As Long
    Dim NoOfGrids As Long
    Dim iYear As Integer, iCoef As Integer
    Dim Intercept As Double
    Dim MyQD As QueryDef
    Dim WTot As Double
    Dim SumTotaal As Double, Sum As Double
    Dim SELogit As Double
    Dim BetrouwbaarheidsintervalLogit As Double
    Dim ResponseResultSet As Recordset

    ' For all Grids and all years
    sql = "SELECT [" & ModelResultsAveragesTableName & "].GridId, [ResponseModel
Coefficients].TaxonNr, [ResponseModel Coefficients].Volgorde, [ResponseModel
Coefficients].TermType, [ResponseModel Coefficients].VarNrA, [ResponseModel
Variables].VariabeleCode, [ResponseModel Coefficients].VarNrB, [ResponseModel
Variables_1].VariabeleCode, [ResponseModel Coefficients].Coeff, [ResponseModel
Variables].VariabeleTransformatie AS TransA, [" & ModelResultsAveragesTableName &
"].Value, " & _
```

```
          "IIf([TransA]='Log(X)',Log([" & ModelResultsAveragesTableName &
"].[Value])/Log(10),IIf([TransA]='Log(X+1)',Log([" & ModelResultsAveragesTableName
& "].[Value]+1)/Log(10),[" & ModelResultsAveragesTableName & "].[Value])) AS ValA,
[ResponseModel Variables_1].VariabeleTransformatie AS TransB, " & _
          "IIf([TransB]='Log(X)',Log([ModelResults
Averages_1].[Value])/Log(10),IIf([TransB]='Log(X+1)',Log([ModelResults
Averages_1].[Value]+1)/Log(10),[ModelResults Averages_1].[Value])) AS ValB, " & _

"IIf([TermType]='cA',[ValA],IIf([TermType]='cA2',[ValA]*[ValA],IIf([TermType]='cAB
',[ValA]*[ValB],IIf([TermType]='cA2B',[ValA]*[ValA]*[ValB],IIf([TermType]='cAB2',[
ValA]*[ValB]*[ValB],Null))))) AS Val, " & _
          "IIf(IsNull([" & ModelResultsAveragesTableName &
"].[GridId]),[Coeff],[Coeff]*[Val]) AS W " & _
          "FROM (((([" & ModelResultsAveragesTableName & "] INNER JOIN
[ResponseModel Coefficients] ON [" & ModelResultsAveragesTableName & "].VariableNr
= [ResponseModel Coefficients].VarNrA) LEFT JOIN [" &
ModelResultsAveragesTableName & "] AS [ModelResults Averages_1] ON [ResponseModel
Coefficients].VarNrB = [ModelResults Averages_1].VariableNr) INNER JOIN
[ResponseModel Variables] ON [ResponseModel Coefficients].VarNrA = [ResponseModel
Variables].VariabeleNr) LEFT JOIN [ResponseModel Variables] AS [ResponseModel
Variables_1] ON [ResponseModel Coefficients].VarNrB = [ResponseModel
Variables_1].VariabeleNr) INNER JOIN [" & RelationSpeciesGridTableName & "] ON (["
& RelationSpeciesGridTableName & "].TaxonNr = [ResponseModel
Coefficients].TaxonNr) AND ([" & ModelResultsAveragesTableName & "].GridId = [" &
RelationSpeciesGridTableName & "].GridId) " & _
          "WHERE ((([ResponseModel Coefficients].TaxonNr)=" & TaxonNr & ") AND
(([ModelResults Averages_1].GridId)=[" & ModelResultsAveragesTableName &
"].[GridId] Or ([ModelResults Averages_1].GridId) Is Null) );"
    On Error GoTo 0
    Set MyQD = CurrentDb.QueryDefs("tmpQuery")
    MyQD.sql = sql
    MyQD.Close

    sql = "SELECT " & Run & " AS Run, tmpQuery.TaxonNr, " & Year & " AS [Year],
tmpQuery.GridId, Sum(tmpQuery.W) AS SumOfW, [ResponseModel Coefficients].Coeff AS
Intercept, [Intercept]+[SumOfW] AS WTot, Exp([WTot])/(1+Exp([WTot])) AS KOV,
KOVOpt, IIf([KOV]>=[KOVOpt],1,0) AS PresentKOV " & _
          "FROM (tmpQuery INNER JOIN [ResponseModel Coefficients] ON
tmpQuery.TaxonNr = [ResponseModel Coefficients].TaxonNr) INNER JOIN [ResponseModel
Statistics] ON [ResponseModel Coefficients].TaxonNr = [ResponseModel
Statistics].TaxonNr " & _
          "GROUP BY " & Run & ", tmpQuery.TaxonNr, " & Year & " , tmpQuery.GridId,
[ResponseModel Coefficients].Coeff, [ResponseModel Coefficients].TermType,
[ResponseModel Coefficients].Coeff, [ResponseModel Statistics].KOVOpt " & _
          "HAVING ((([ResponseModel Coefficients].TermType)='c'));"
    On Error GoTo 0
    Set MyQD = CurrentDb.QueryDefs("tmpSumQuery")
    MyQD.sql = sql
    MyQD.Close


    If Not DoCalculateConfidenceIntervals Then

      sql = "INSERT INTO [" & ResponseModelResultsTableName & "] ( Run, Year,
TaxonNr, GridId, KOV, PresentKOV ) " & _
            "SELECT Run, Year, TaxonNr, GridId, KOV, PresentKOV " & _
            "FROM tmpSumQuery;"
      CurrentDb.Execute sql

    Else   ' calculate confidence intervals

      Set ResponseResultSet =
CurrentDb.OpenRecordset(ResponseModelResultsTableName)

      sql = "SELECT TaxonNr, Volgorde, TermType " & _
            "FROM [ResponseModel Coefficients] " & _
            "WHERE (TaxonNr=" & TaxonNr & ");"
      Set CoefSet = CurrentDb.OpenRecordset(sql)
      CoefAantal = AantalRecords(CoefSet)
      CoefSet.Close

      ReDim AbiotiekWaarde(CoefAantal)
```

```
      ' Vul CovMatrix
      ReDim CovMatrix(CoefAantal, CoefAantal)
      sql = "SELECT [ResponseModel Covariances].CoeffNr1, [ResponseModel
Covariances].CoeffNr2, [ResponseModel Covariances].Waarde AS Val " & _
            "FROM [ResponseModel Covariances] " & _
            "WHERE ( TaxonNr = " & TaxonNr & " ) " & _
            "ORDER BY [ResponseModel Covariances].CoeffNr1, [ResponseModel
Covariances].CoeffNr2;"
      Set tmpset = CurrentDb.OpenRecordset(sql)
      If Not tmpset.EOF Then
        While Not tmpset.EOF
          CovMatrix(tmpset!CoeffNr1, tmpset!CoeffNr2) = tmpset!Val
          tmpset.MoveNext
        Wend
      End If
      tmpset.Close

      sql = "TRANSFORM Avg(Val) AS AvgOfVal " & _
            "SELECT GridId, TaxonNr " & _
            "FROM " & "tmpQuery" & " " & _
            "WHERE ((GridId Is Not Null)) " & _
            "GROUP BY GridId, TaxonNr " & _
           "PIVOT Volgorde;"
      On Error GoTo 0
      Set MyQD = CurrentDb.QueryDefs("tmpCoefQuery")
      MyQD.sql = sql
      MyQD.Close


      sql = "SELECT tmpSumQuery.Intercept, tmpSumQuery.SumOfW, tmpSumQuery.WTot,
tmpSumQuery.KOV, tmpSumQuery.KOVOpt, tmpSumQuery.PresentKOV, tmpCoefQuery.* " & _
            "FROM tmpSumQuery INNER JOIN tmpCoefQuery ON (tmpSumQuery.TaxonNr =
tmpCoefQuery.TaxonNr) AND (tmpSumQuery.GridId = tmpCoefQuery.GridId);"
      Set tmpset = CurrentDb.OpenRecordset(sql)

      While Not tmpset.EOF
        AbiotiekWaarde(1) = 1
        For iCoef = 2 To CoefAantal
          Err = 0
          On Error Resume Next
          AbiotiekWaarde(iCoef) = tmpset("" & iCoef & "")
          If Err > 0 Then  ' value not present
            AbiotiekWaarde(iCoef) = 0
          End If
          If IsNull(AbiotiekWaarde(iCoef)) Then
            AbiotiekWaarde(iCoef) = 0
          End If
        Next iCoef

        ' Calculate confidence interval
        Dim iCoef1 As Integer, iCoef2 As Integer
        SumTotaal = 0
        For iCoef1 = 1 To CoefAantal
          Sum = 0
          For iCoef2 = 1 To iCoef1
            Sum = Sum + AbiotiekWaarde(iCoef2) * CovMatrix(iCoef1, iCoef2)
          Next iCoef2
          For iCoef2 = iCoef1 + 1 To CoefAantal
            Sum = Sum + AbiotiekWaarde(iCoef2) * CovMatrix(iCoef2, iCoef1)
          Next iCoef2
          Sum = Sum * AbiotiekWaarde(iCoef1)
          SumTotaal = SumTotaal + Sum
        Next iCoef1
        SELogit = SumTotaal ^ 0.5

        ' Calculate confidence interval
        BetrouwbaarheidsintervalLogit = tValue005 * (SELogit ^ 0.5)


        ' Bereken KOV-resultaat en betrouwbaarheidsgrenzen en sla deze op
        Dim L1 As Double, L2 As Double
```

```
        Dim Kov_L1 As Double, Kov_L2 As Double


        ' Bereken betrouwbaarheidsinterval
        L1 = tmpset!WTot - BetrouwbaarheidsintervalLogit
        L2 = tmpset!WTot + BetrouwbaarheidsintervalLogit

        Kov_L1 = Exp(L1) / (1 + Exp(L1))
        Kov_L2 = Exp(L2) / (1 + Exp(L2))

        ' Sla resultaten op
        ResponseResultSet.AddNew
        ResponseResultSet!Run = Run
        ResponseResultSet!GridId = tmpset!GridId
        ResponseResultSet!TaxonNr = TaxonNr
        ResponseResultSet!Year = Year
        ResponseResultSet!KOV = tmpset!KOV
        ResponseResultSet!PresentKOV = tmpset!PresentKOV
        ResponseResultSet!L1 = Kov_L1
        ResponseResultSet!L2 = Kov_L2
        ResponseResultSet!PresentL1 = IIf(Kov_L1 >= tmpset!KOVOpt, 1, 0)
        ResponseResultSet!PresentL2 = IIf(Kov_L2 >= tmpset!KOVOpt, 1, 0)
        ResponseResultSet.Update

        tmpset.MoveNext
      Wend
      tmpset.Close

      ResponseResultSet.Close

    End If

End_CalculateResponsesResponseSetSoort:
    Exit Function

Error_CalculateResponsesResponseSetSoort:
    WriteLog "Error in function CalculateResponsesResponseSetSoort(" & TaxonNr &
"): Err=" & Err & ": " & Err.Description
    Resume End_CalculateResponsesResponseSetSoort

End Function

Function ProcessIniFile(IniFileName As String) As Integer
On Error GoTo Error_ProcessIniFile

    Dim filenum As Long, arg1 As String, arg2 As String, StrLine As String

    ResponseModelsDatabaseName = GetIniFileItem(IniFileName, "Files",
"ResponseModelsDatabaseName")
    VariableTranslationFileName = GetIniFileItem(IniFileName, "Files",
"VariableTranslationFileName")
    BiodivFileName = GetIniFileItem(IniFileName, "Files", "BiodivFileName")
    ModelResultsDatabaseName = GetIniFileItem(IniFileName, "Files",
"ResultsDatabaseName")
    LogFileName = GetIniFileItem(IniFileName, "Files", "LogFileName")
    DoCalculateConfidenceIntervals = False
    DoCalculateConfidenceIntervals = IIf(GetIniFileItem(IniFileName, "Options",
"CalculateConfidenceIntervals") = "True", True, False)

    ' Check input
    If Len(ResponseModelsDatabaseName) = 0 Or _
       Len(VariableTranslationFileName) = 0 Or _
       Len(BiodivFileName) = 0 Or _
       Len(ModelResultsDatabaseName) = 0 Or _
       Len(LogFileName) = 0 Then
      WriteLog "Error in initialization file: No or incorrect filename specified"
      ProcessIniFile = False
    Else
      ProcessIniFile = True
    End If

Einde_ProcessIniFile:
```

```
    Close #filenum
    Exit Function

Error_ProcessIniFile:
    WriteLog "Error in function ProcessIniFile; err=" & Err & ": " &
Err.Description
    ProcessIniFile = False
    Resume Einde_ProcessIniFile

End Function

Function InitLog()
    Dim filenum As Integer
    filenum = FreeFile
    Open LogFileName For Output As filenum
    Print #filenum, "Log started at " & Now
    Close #filenum
End Function

Function WriteLog(Msg As String)
    Dim filenum As Integer
    filenum = FreeFile
    Open LogFileName For Append As filenum
    Print #filenum, Msg
    Close #filenum
End Function

Function EndLog()
    Dim filenum As Integer
    filenum = FreeFile
    If Len(LogFileName) > 0 Then
      Open LogFileName For Append As filenum
      Print #filenum, "Log ended at " & Now
      Close #filenum
    End If
End Function

Function RefreshLink(TargetTableName As String, DatabaseName As String,
SourceTableName As String)
    DeleteLink TargetTableName
    DoCmd.TransferDatabase acLink, "Microsoft Access", DatabaseName, acTable,
SourceTableName, TargetTableName
End Function

Function DeleteLink(TableName As String)
    On Error Resume Next
    DoCmd.Close acTable, TableName, acSaveNo
    DoCmd.DeleteObject acTable, TableName
    CurrentDb.TableDefs.Refresh
    On Error GoTo 0
End Function

Function LoadTable(TableName As String, InputFileName As String, Optional DoErase
As Integer = True) As Integer
On Error GoTo Error_LoadTable
    Dim ifield As Integer, sql As String, dumval
    Dim strFieldNames As String
    Dim FieldArray(100) As String, FieldIndex(100) As Integer, nFieldNames As
Integer
    Dim tmpset As DAO.Recordset
    Const tmpInputTableName = "tmp Input"

    On Error Resume Next
    DoCmd.Close acTable, tmpInputTableName
    DoCmd.DeleteObject acTable, tmpInputTableName
    On Error GoTo Error_LoadTable

    If Right(InputFileName, 3) = "dbf" Then
      DoCmd.TransferDatabase acImport, "dBase IV", Left(InputFileName,
rinstr(InputFileName, "\") - 1), acTable, Mid(InputFileName, rinstr(InputFileName,
"\") + 1), tmpInputTableName
    Else
```

```
   DoCmd.TransferText acLinkDelim, , tmpInputTableName, InputFileName, True
  End If
  CurrentDb.TableDefs.Refresh
  'DBEngine.Idle dbRefreshCache
  DoEvents
  ' Test for one or more fields
  ' if more fields: delimination worked apparently well: test existence of all
fields needed and load them
  If CurrentDb.TableDefs(tmpInputTableName).Fields.Count > 1 Then
   ' Test existence ot TableName
   On Error Resume Next
   dumval = CurrentDb.TableDefs(TableName).Name
   If Err > 0 Then  ' table does not exists
    DoCmd.CopyObject , TableName, acTable, tmpInputTableName
   Else  'table exists: check fields
    For ifield = 0 To CurrentDb.TableDefs(TableName).Fields.Count - 1
     On Error Resume Next
     Err = 0
     dumval =
CurrentDb.TableDefs(tmpInputTableName).Fields(CurrentDb.TableDefs(TableName).Field
s(ifield).Name).Type
     If Err > 0 Then
      WriteLog "Error: Field '" &
CurrentDb.TableDefs(TableName).Fields(ifield).Name & "' missing from table '" &
tmpInputTableName & "' while generating Table '" & TableName & "'"
      GoTo Error_LoadTable
     End If
     On Error GoTo 0
    Next ifield
    ' Fields checked and present: fill table
    If DoErase Then
     EraseTable TableName
    End If
    For ifield = 0 To CurrentDb.TableDefs(TableName).Fields.Count - 1
     strFieldNames = strFieldNames & ",[" &
CurrentDb.TableDefs(TableName).Fields(ifield).Name & "]"
    Next ifield
    strFieldNames = Mid(strFieldNames, 2)
    sql = "INSERT INTO [" & TableName & "] ( " & strFieldNames & " ) " & _
       "SELECT " & strFieldNames & " " & _
       "FROM [" & tmpInputTableName & "];"
    CurrentDb.Execute sql
   End If
  Else   ' fields not deliminated
   ' Reload file without headings
   DoCmd.DeleteObject acTable, tmpInputTableName
   CurrentDb.TableDefs.Refresh
   DBEngine.Idle dbRefreshCache
   DoEvents
   DoCmd.TransferText acLinkDelim, , tmpInputTableName, InputFileName, False
   CurrentDb.TableDefs.Refresh
   DBEngine.Idle dbRefreshCache
   DoEvents

   Dim SourceSet As Recordset, TargetSet As Recordset
   Set SourceSet = CurrentDb.OpenRecordset(tmpInputTableName)
   strFieldNames = SourceSet.Fields(0).Value   ' get fieldnames form first
record
   For ifield = 0 To CurrentDb.TableDefs(TableName).Fields.Count - 1
    FieldIndex(ifield) = GetWordIndex(strFieldNames,
CurrentDb.TableDefs(TableName).Fields(ifield).Name, ",")
    If FieldIndex(ifield) = 0 Then
     WriteLog "Error: Field '" &
CurrentDb.TableDefs(TableName).Fields(ifield).Name & "' missing from file '" &
InputFileName & "'"
     GoTo Error_LoadTable
    End If
   Next ifield

   If DoErase Then
    EraseTable TableName
   End If
```

```
      Set TargetSet = CurrentDb.OpenRecordset(TableName)
      SourceSet.MoveNext   ' skip heading
      While Not SourceSet.EOF
        If Not IsNull(SourceSet(0)) Then
        nFieldNames = SplitStringArg(FieldArray, SourceSet(0), ",")
        TargetSet.AddNew
        For ifield = 0 To CurrentDb.TableDefs(TableName).Fields.Count - 1
          If TargetSet(ifield).Type = dbText Then
            TargetSet(ifield).Value = IIf(Len(FieldArray(ifield + 1)) > 0,
FieldArray(ifield + 1), Null)
          Else
            TargetSet(ifield).Value = IIf(Len(FieldArray(ifield + 1)) > 0,
CDbl(FieldArray(ifield + 1)), Null)
          End If
        Next ifield
        TargetSet.Update
        End If
        SourceSet.MoveNext
      Wend
      SourceSet.Close
      TargetSet.Close

    End If
    LoadTable = True

End_LoadTable:
    DoCmd.DeleteObject acTable, tmpInputTableName
    Exit Function

Error_LoadTable:
    LoadTable = False
    Resume Next
    GoTo End_LoadTable

End Function

Function LoadList(TableName As String, InputFileName As String, Optional DoErase
As Integer = True) As Integer
On Error GoTo error_loadlist
    Dim narg As Long, strArray(100) As String
    Dim tSet As Recordset
    Dim filenum As Integer
    Dim StrLine As String
    Dim strTopic As String, strValue As String

    If DoErase Then
      EraseTable TableName
    End If
    filenum = FreeFile
    Open InputFileName For Input As #filenum
    Set tSet = CurrentDb.OpenRecordset(TableName)
    Err = 0
    tSet.AddNew
    While Not EOF(filenum) And Err = 0
      Input #filenum, StrLine
      If Len(StrLine) > 0 Then
        If InStr(StrLine, " ") > 0 Then
          strTopic = Trim(Left(StrLine, InStr(StrLine, " ")))
          strValue = Trim(Mid(StrLine, InStr(StrLine, " ")))
          On Error Resume Next
          tSet(strTopic) = strValue
          If Err > 0 Then
            WriteLog ("Error updating field '" & strArray(1) & "' in table '" &
TableName & "'")
            GoTo error_loadlist
          End If
          On Error GoTo error_loadlist
        Else
          WriteLog ("Error in List in file '" & InputFileName & "': no space found
in line '" & StrLine & "'")
          GoTo error_loadlist
```

```
        End If
      End If
    Wend
    Close #filenum
    tSet.Update
    tSet.Close
    LoadList = True

end_loadlist:
    Exit Function

error_loadlist:
    WriteLog ("Error in function LoadList: Err=" & Err & ": " & Err.Description)
    LoadList = False
    Resume end_loadlist

End Function

Function CalculateConfidenceInterval(LinPred As Double, TaxonNr As Long, LakeId As
Long, Year As Integer) As Range
' Functie voor het berekenen van de betrouwbaarheidsintervallen voor een model,
een soort en een abiotiekset
' Resultaat: afstand van schatting lineaire predictor tot betrouwbaarheidsgrens
(voor gebruik in query; logit-schaal)
' Resultaten worden ook opgeslagen in de tabel AbiotiekResultaat als KOV en
betrouwbaarheidsinterval KOV
' Eenvoudige(r) sequentiele procedure die het principe weergeeft maar veel langer
rekent
    Dim CoefSet As Recordset, CorSet As Recordset, tmpset As Recordset,
AbiotiekSet As Recordset
    Dim CorMatrix() As Double, CovMatrix() As Double
    Dim CoefWaarde() As Double, CoefSE() As Double
    Dim AbiotiekWaarde() As Double
    Dim VariabeleId() As Long   ' vertaaltabel voor VariabeleId's
    Dim CoefAantal As Long, Sum As Double, SumTotaal As Double
    Dim SELogit As Double
    Dim sql As String
    Dim ivar1 As Long, ivar2 As Long
    Dim W As Double, BetrouwbaarheidsintervalLogit As Double
    Dim Resultaat As Range
    Dim iCoef As Integer

    sql = "SELECT TaxonNr, Volgorde, TermType " & _
          "FROM [ResponseModel Coefficients] " & _
          "WHERE (TaxonNr=" & TaxonNr & ");"
    Set CoefSet = CurrentDb.OpenRecordset(sql)
    CoefAantal = AantalRecords(CoefSet)
    CoefSet.Close

    ReDim CovMatrix(CoefAantal, CoefAantal)
    ReDim AbiotiekWaarde(CoefAantal)

    ' Vul CovMatrix
    sql = "SELECT SpeciesInfo.SpeciesId, [ResponseModel Covariances].CoeffNr1,
[ResponseModel Covariances].CoeffNr2, [ResponseModel Covariances].Waarde AS Val "
& _
          "FROM (SpeciesInfo INNER JOIN [ResponseModel Taxa] ON
SpeciesInfo.SpeciesName = [ResponseModel Taxa].TaxonNaam) INNER JOIN
[ResponseModel Covariances] ON [ResponseModel Taxa].TaxonCode = [ResponseModel
Covariances].SoortNaam " & _
          "WHERE ((([ResponseModel Taxa].TaxonNr) = " & TaxonNr & ")) " & _
          "ORDER BY [ResponseModel Covariances].CoeffNr1, [ResponseModel
Covariances].CoeffNr2;"
    Set tmpset = CurrentDb.OpenRecordset(sql)
    If Not tmpset.EOF Then
      While Not tmpset.EOF
        CovMatrix(tmpset!CoeffNr1, tmpset!CoeffNr2) = tmpset!Val
        tmpset.MoveNext
      Wend
      tmpset.Close

      ' Calculate variable values for all terms
```

```
      For iCoef = 1 To CoefAantal
         sql = "SELECT [" & ModelResultsAveragesTableName & "].LakeId, [" &
ModelResultsAveragesTableName & "].Year, [ResponseModel Taxa].TaxonNr,
[ResponseModel Taxa].TaxonCode, [ResponseModel Coefficients].Volgorde,
[ResponseModel Coefficients].TermType, [ResponseModel Taxa].TaxonNaam,
[ResponseModel Coefficients].VarNrA, [ResponseModel Coefficients].VarNrB,
[ResponseModel Coefficients].Coeff, [" & VariableRelationsTableName &
"].Transformation AS TransA, VariableInfo.VariableName AS VarA, [" &
ModelResultsAveragesTableName & "].Value AS ValA, VariableInfo_1.VariableName AS
VarB, [[" & VariableRelationsTableName & "]._1].Transformation AS TransB,
[ModelResults Averages_1].Value AS ValB " & _
               "FROM (((VariableInfo RIGHT JOIN (([ResponseModel Taxa] INNER JOIN
(([ResponseModel Coefficients] LEFT JOIN [ResponseModel Variables] ON
[ResponseModel Coefficients].VarNrA = [ResponseModel Variables].VariabeleNr) LEFT
JOIN [ResponseModel Variables] AS [ResponseModel Variables_1] ON [ResponseModel
Coefficients].VarNrB = [ResponseModel Variables_1].VariabeleNr) ON [ResponseModel
Taxa].TaxonNr = [ResponseModel Coefficients].TaxonNr) LEFT JOIN [" &
VariableRelationsTableName & "] ON [ResponseModel Variables].VariabeleCode = [" &
VariableRelationsTableName & "].ResponseVariableName) ON VariableInfo.VariableName
= [" & VariableRelationsTableName & "].VariableName) LEFT JOIN [" &
ModelResultsAveragesTableName & "] ON VariableInfo.VariableId = [" &
ModelResultsAveragesTableName & "].VariableId) " & _
               "LEFT JOIN [" & VariableRelationsTableName & "] AS [[" &
VariableRelationsTableName & "]._1] ON [ResponseModel Variables_1].VariabeleCode =
[[" & VariableRelationsTableName & "]._1].ResponseVariableName) LEFT " & _
               "JOIN VariableInfo AS VariableInfo_1 ON [[" &
VariableRelationsTableName & "]._1].VariableName = VariableInfo_1.VariableName)
LEFT JOIN [" & ModelResultsAveragesTableName & "] AS [ModelResults Averages_1] ON
VariableInfo_1.VariableId = [ModelResults Averages_1].VariableId " & _
               "WHERE ((([" & ModelResultsAveragesTableName & "].LakeId) = " &
LakeId & " Or ([" & ModelResultsAveragesTableName & "].LakeId) Is Null) And (([" &
ModelResultsAveragesTableName & "].Year) = " & Year & " Or ([" &
ModelResultsAveragesTableName & "].Year) Is Null) And (([ResponseModel
Taxa].TaxonNr) = " & TaxonNr & ") And (([ResponseModel Coefficients].Volgorde) = "
& iCoef & ") And (([ModelResults Averages_1].LakeId) = [" &
ModelResultsAveragesTableName & "].[LakeId] Or ([ModelResults Averages_1].LakeId)
Is Null) And (([ModelResults Averages_1].Year) = " & Year & " Or ([ModelResults
Averages_1].Year) Is Null)) " & _
               "ORDER BY [ResponseModel Taxa].TaxonNaam;"
          Set tmpset = CurrentDb.OpenRecordset(sql)
          If Not tmpset.EOF Then
                ' Perform transformation of variables
                Dim ValTrA As Double, ValTrB As Double
                If Not IsNull(tmpset!ValA) Then
                  If IsNull(tmpset!TransA) Then
                    ValTrA = tmpset!ValA
                  Else
                    If tmpset!TransA = "Log(x)" Then
                      ValTrA = Log10(tmpset!ValA)
                    ElseIf tmpset!TransA = "Log(x+1)" Then
                      ValTrA = Log10(tmpset!ValA + 1#)
                    Else
                      MsgBox "Unknown transformation '" & tmpset!TransA & "'"
                    End If
                  End If
                End If
                If Not IsNull(tmpset!ValB) Then
                  If IsNull(tmpset!TransB) Then
                    ValTrB = tmpset!ValB
                  Else
                    If tmpset!TransB = "Log(x)" Then
                      ValTrB = Log10(tmpset!ValB)
                    ElseIf tmpset!TransB = "Log(x+1)" Then
                      ValTrB = Log10(tmpset!ValB + 1#)
                    Else
                      MsgBox "Unknown transformation '" & tmpset!TransB & "'"
                    End If
                  End If


                End If
```

```
                Select Case tmpset!TermType
                  Case "c":  W = 1
                  Case "cA": W = ValTrA
                  Case "cA2": W = (ValTrA ^ 2)
                  Case "cAB": W = ValTrA * ValTrB
                  Case "cA2B": W = (ValTrA ^ 2) * ValTrB
                  Case "cAB2": W = ValTrA * (ValTrB ^ 2)
                  Case "cA2B2": W = (ValTrA ^ 2) * (ValTrB ^ 2)
                End Select

              tmpset.Close
              AbiotiekWaarde(iCoef) = W
        Else
          tmpset.Close
          MsgBox "Coefficient " & iCoef & " niet gevonden"
        End If
      Next iCoef

      ' Calculate SE of estimate

      Dim iCoef1 As Integer, iCoef2 As Integer
      SumTotaal = 0
      For iCoef1 = 1 To CoefAantal
        Sum = 0
        For iCoef2 = 1 To iCoef1
          Sum = Sum + AbiotiekWaarde(iCoef2) * CovMatrix(iCoef1, iCoef2)
        Next iCoef2
        For iCoef2 = iCoef1 + 1 To CoefAantal
          Sum = Sum + AbiotiekWaarde(iCoef2) * CovMatrix(iCoef2, iCoef1)
        Next iCoef2
        Sum = Sum * AbiotiekWaarde(iCoef1)
        SumTotaal = SumTotaal + Sum
      Next iCoef1
      SELogit = SumTotaal ^ 0.5

      BetrouwbaarheidsintervalLogit = tValue005 * (SELogit ^ 0.5)

      ' Bereken KOV-resultaat en betrouwbaarheidsgrenzen en sla deze op
      Dim L1 As Double, L2 As Double
      Dim Kov_L1 As Double, Kov_L2 As Double

      ' Bereken betrouwbaarheidsinterval
      L1 = LinPred - BetrouwbaarheidsintervalLogit
      L2 = LinPred + BetrouwbaarheidsintervalLogit

      Kov_L1 = Exp(L1) / (1 + Exp(L1))
      Kov_L2 = Exp(L2) / (1 + Exp(L2))

      Resultaat.X1 = Kov_L1
      Resultaat.X2 = Kov_L2

    Else
      tmpset.Close
      Resultaat.X1 = 0
      Resultaat.X2 = 0

    End If

    CalculateConfidenceInterval = Resultaat

End Function


Function GetIniFileItem(IniFileName As String, Section As String, Item As String)
As String
    Dim filenum As Long
    Dim CurrentSection As String, CurrentItem As String
    Dim Found As Integer, tmpstr As String
    Dim SectionFound As Integer, SectionProcessed As Integer

    filenum = FreeFile
    Open IniFileName For Input As filenum
```

```
    CurrentSection = ""
    SectionFound = False
    SectionProcessed = False
    Found = False
    While Not EOF(filenum) And Not SectionProcessed And Not Found
      Input #filenum, tmpstr
      tmpstr = Trim(tmpstr)
      If Left(tmpstr, 1) = "[" And Right(tmpstr, 1) = "]" Then
        CurrentSection = Mid(tmpstr, 2, Len(tmpstr) - 2)
        If CurrentSection = Section Then
          SectionFound = True
        Else
          If SectionFound Then
            SectionProcessed = True
          End If
          SectionFound = False
        End If
      ElseIf SectionFound And Left(tmpstr, 2) <> "//" Then
        If InStr(tmpstr, "=") > 0 Then
          CurrentItem = Trim(Left(tmpstr, InStr(tmpstr, "=") - 1))
          If CurrentItem = Item Then
            Found = True
            GetIniFileItem = Trim(Mid(tmpstr, InStr(tmpstr, "=") + 1))
          End If
        End If
      End If
    Wend
    Close filenum
    If Not Found Then
      GetIniFileItem = ""
    End If
End Function


Function  LoadAsciiGridData(TableName  As  String,  TemplateTableName  As  String,
FileName As String, Optional DoErase As Integer = True)
    Dim Name As String

    If DoErase Then   ' Load data in empty table
      On Error Resume Next
      ' Check if Table exists
      Name = CurrentDb.TableDefs(TableName).Name
      If Err = 0 Then  ' Table exists
        EraseTable TableName
      Else
        DoCmd.Close acTable, TableName, acSaveNo
        DoCmd.DeleteObject acTable, TableName
        DoCmd.TransferDatabase        acExport,        "Microsoft        Access",
ModelResultsDatabaseName, acTable, TemplateTableName, TableName
        RefreshLink TableName, ModelResultsDatabaseName, TableName
        CurrentDb.TableDefs.Refresh
        DoEvents
      End If
      On Error GoTo 0
      End If

    ReadAsciiGrid FileName, TableName, "X", "Y", "Value", False, 0, 0, 0
    DoEvents

End Function

Function    LoadRelationTaxonGridData(RelationSpeciesGridTableName    As    String,
RelationSpeciesGridTemplateTableName   As   String,   SpecRefFileName   As   String,
SpeciesCodeColumnName As String, GridCodeColumnName As String, Optional DoErase As
Integer = True)
On Error GoTo Error_LoadRelationTaxonGridData
    Dim sql As String
    Dim Name As String

    If DoErase Then   ' Load data in empty table
      On Error Resume Next
      ' Check if Table exists
```

```
      Name = CurrentDb.TableDefs(RelationSpeciesGridTableName).Name
      If Err = 0 Then  ' Table exists
        EraseTable RelationSpeciesGridTableName
      Else
        On Error GoTo Error_LoadRelationTaxonGridData
        CurrentDb.TableDefs.Refresh
        DoCmd.TransferDatabase acExport, "Microsoft Access",
ModelResultsDatabaseName, acTable, RelationSpeciesGridTemplateTableName,
RelationSpeciesGridTableName
        RefreshLink RelationSpeciesGridTableName, ModelResultsDatabaseName,
RelationSpeciesGridTableName
        CurrentDb.TableDefs.Refresh
        DoEvents
      End If
    End If

    If InStr(SpecRefFileName, ".mdb:") > 0 Then 'table in Access-database
      DoCmd.TransferDatabase acLink, "Microsoft Access", Left(SpecRefFileName,
InStr(SpecRefFileName, ".mdb:") + 3), acTable, Mid(SpecRefFileName,
InStr(SpecRefFileName, ".mdb:") + 5), "tmpData", False
    Else
      LoadTable "tmpData", SpecRefFileName
    End If
    sql = "INSERT INTO [" & RelationSpeciesGridTableName & "] ( GridId, TaxonNr )
" & _
          "SELECT tmpData.[" & GridCodeColumnName & "], [ResponseModel
Taxa].TaxonNr " & _
          "FROM tmpData INNER JOIN [ResponseModel Taxa] ON tmpData.[" &
SpeciesCodeColumnName & "] = [ResponseModel Taxa].TaxonCode;"
    CurrentDb.Execute sql
    DoCmd.DeleteObject acTable, "tmpData"

    DoEvents

End_LoadRelationTaxonGridData:
    Exit Function

Error_LoadRelationTaxonGridData:
    WriteLog "Error in function LoadRelationTaxonGridData(): " & Err.Description
    Resume End_LoadRelationTaxonGridData

End Function
```

Appendix 4 Algemene Access VBA-functies

Deze functies zijn door mij verzameld en geprogrammeerd om het leven van een Access-programmeur wat aangenamen te maken. Ze kunnen in een aparte module in een database geplaatst worden, of in een centrale bibliotheek die vanuit een database aangesproken wordt. Dit laatste kan door vanuit een module een referentie op te nemen naar de database met de algemene modules. Zie verder de Access-documentatie.

Listing 4.1. Algemene procedures AccessLib.

```
Option Compare Database
Option Explicit
Const MaxLinesReadInDisplay = 100

Public Function AantalRecords(dataset As Recordset) As Long
    On Error Resume Next
    dataset.MoveLast
    If Err > 0 Then
      AantalRecords = 0
    Else
      AantalRecords = dataset.RecordCount
      dataset.MoveFirst
    End If
    On Error GoTo 0
End Function

Public Function NrOfRecords(dataset As Recordset) As Long
    NrOfRecords = AantalRecords(dataset)
End Function

Public Function Log10(X As Double) As Double
    Log10 = Log(X) / Log(CDbl(10))
End Function

Public Function BerekenAfstand(X1 As Double, Y1 As Double, X2 As Double, Y2 As
Double) As Double
  BerekenAfstand = (Abs(X2 - X1) ^ 2 + Abs(Y2 - Y1) ^ 2) ^ 0.5
End Function

Public Function max(a As Variant, b As Variant) As Variant
  max = IIf(a >= b, a, b)
End Function
Public Function min(a As Variant, b As Variant) As Variant
  min = IIf(a <= b, a, b)
End Function

Public Function maximum(a As Variant, b As Variant) As Variant
  maximum = IIf(a >= b, a, b)
End Function
Public Function minimum(a As Variant, b As Variant) As Variant
  minimum = IIf(a <= b, a, b)
End Function

Public Function GetFileNameFromCompletePath(Path As String) As String
    Dim i As Integer, ipos As Integer
    If InStr(Path, "\") > 0 Then
      GetFileNameFromCompletePath = Mid(Path, rinstr(Path,"\") + 1)
    Else
      GetFileNameFromCompletePath = Path
    End If
End Function

Public Function GetPathFromCompleteFileName(CompleteFileName As String) As String
```

```
    GetPathFromCompleteFileName = Left(CompleteFileName, Len(CompleteFileName) -
Len(GetFileNameFromCompletePath(CompleteFileName)))
End Function

Public Function GetWord(s As String, nWord As Integer) As String
' Gets the nWord word out of string s
On Error GoTo Error_GetWord
  Dim i As Integer, j As Integer, iWord As Integer

  iWord = 1
  i = 1
  ' jump to beginning of first word
  While Mid(s, i, 1) = " "
    i = i + 1
  Wend
  ' go to beginning of word n-1
  For iWord = 1 To nWord - 1
    ' skip word
    While Mid(s, i, 1) <> " "
      i = i + 1
    Wend
    ' skip spaces
    While Mid(s, i, 1) = " "
      i = i + 1
    Wend
  Next iWord
  ' walk to end of word
  j = i
  While Mid(s, j, 1) <> " " And j < Len(s)
    j = j + 1
  Wend

End_GetWord:
  ' retrieve the word
  GetWord = Mid(s, i, j - i + IIf(j = Len(s), 1, 0))
  Exit Function

Error_GetWord:
  If i > Len(s) Or j > Len(s) Then
    MsgBox "Not so many words (" & nWord & ") in string (" & s & ")"
    i = 1
    j = 1
  End If
  Resume End_GetWord

End Function


Public Function GetWordx(s As String, nWord As Integer, delim As String) As String
' Gets the nWord word out of string s
On Error GoTo Error_GetWordx
  Dim i As Integer, j As Integer, iWord As Integer

  iWord = 1
  i = 1
  ' jump to beginning of first word
  While Mid(s, i, 1) = " "
    i = i + 1
  Wend
  ' go to beginning of word n-1
  For iWord = 1 To nWord - 1
    ' skip word
    While InStr(delim, Mid(s, i, 1)) = 0 And i < Len(s)
      i = i + 1
    Wend
    ' skip spaces
    While InStr(delim, Mid(s, i, 1)) = 1 And i < Len(s)
      i = i + 1
    Wend
  Next iWord
```

```
  ' walk to end of word
  j = i
  While InStr(delim, Mid(s, j, 1)) = 0 And j < Len(s)
    j = j + 1
  Wend

End_GetWordx:
  ' retrieve the word
  GetWordx = Mid(s, i, j - i + IIf(j = Len(s), 1, 0))
  Exit Function

Error_GetWordx:
  If i > Len(s) Or j > Len(s) Then
    MsgBox "GetWordx: Not so many words (" & nWord & ") in string (" & s & ")"
    i = 1
    j = 1
  End If
  Resume End_GetWordx

End Function

Function GetWordIndex(s As String, Tekst As String, delim As String) As Integer
    Dim WordArray(1000) As String, nWords As Integer, i As Integer
    nWords = SplitStringArg(WordArray, s, delim)
    For i = 1 To nWords
      If WordArray(i) = Tekst Then
        Exit For
      End If
    Next i
    GetWordIndex = IIf(i <= nWords, i, 0)
End Function

Public Sub Testje()
    MsgBox "Een test vanuit een andere database"
    Dim MyDb As Database, MySet As Recordset
    Set MyDb = DBEngine.Workspaces(0).Databases(0)
    MsgBox "twee"
    MsgBox "Eerste tabel: " & MyDb.TableDefs(0).Name
End Sub

Public Function VulTabelMetNullen(TabelNaam As String)
    MsgBox "Begin VulTabelMetNullen"
    Dim MyDb As Database, MySet As Recordset, ifield As Integer, nField As Integer
    Set MyDb = DBEngine.Workspaces(0).Databases(0)
    Set MySet = MyDb.OpenRecordset(TabelNaam)
    nField = MySet.Fields.Count
    While Not MySet.EOF
      For ifield = 0 To nField - 1
        If IsNull(MySet.Fields(ifield).Value) Then
          MySet.Edit
          MySet.Fields(ifield).Value = 0
          MySet.Update
          DoEvents
        End If
      Next ifield
      MySet.MoveNext
    Wend
    MySet.Close
    VulTabelMetNullen = 1
End Function


Public Function VulTabelMetNA(TabelNaam As String)
    MsgBox "Begin VulTabelMetNA"
    Dim MyDb As Database, MySet As Recordset, ifield As Integer, nField As Integer
    Set MyDb = DBEngine.Workspaces(0).Databases(0)
    Set MySet = MyDb.OpenRecordset(TabelNaam)
    nField = MySet.Fields.Count
    While Not MySet.EOF
      For ifield = 0 To nField - 1
```

```
        If IsNull(MySet.Fields(ifield).Value) Then
          MySet.Edit
          MySet.Fields(ifield).Value = "NA"
          MySet.Update
          DoEvents
        End If
      Next ifield
      MySet.MoveNext
    Wend
    MySet.Close
    VulTabelMetNA = 1
End Function

Public Function SchrijfTabelMetNA(FileNaam As String)

End Function

Public Function BerekenCorrelatie(VarSet As Recordset, Var1Naam As String,
Var2Naam As String)
On Error GoTo Fout_BerekenCorrelatie
  Dim Count As Long
  Dim SumVar1 As Double, SumVar2 As Double, SumVar1Squared As Double,
SumVar2Squared As Double
  Dim SumVar1Var2Product As Double
  Dim SumOfSquaresVar1 As Double, SumOfSquaresVar2 As Double
  Dim SumOfProducts As Double
  Dim CorrCoef As Double

  Count = 0
  With VarSet
    While Not .EOF
      SumVar1 = SumVar1 + CDbl(.Fields(Var1Naam))
      SumVar2 = SumVar2 + CDbl(.Fields(Var2Naam))
      SumVar1Squared = SumVar1Squared + CDbl(.Fields(Var1Naam)) ^ 2
      SumVar2Squared = SumVar2Squared + CDbl(.Fields(Var2Naam)) ^ 2
      SumVar1Var2Product = SumVar1Var2Product + CDbl(.Fields(Var1Naam)) *
CDbl(.Fields(Var2Naam))
      Count = Count + 1
      .MoveNext
    Wend
  End With
  SumOfSquaresVar1 = SumVar1Squared - (SumVar1 ^ 2) / Count
  SumOfSquaresVar2 = SumVar2Squared - (SumVar2 ^ 2) / Count
  SumOfProducts = SumVar1Var2Product - (SumVar1 * SumVar2) / Count
  CorrCoef = SumOfProducts / (SumOfSquaresVar1 * SumOfSquaresVar2) ^ 0.5
  BerekenCorrelatie = CorrCoef

Einde_BerekenCorrelatie:
  Exit Function

Fout_BerekenCorrelatie:
  BerekenCorrelatie = "NA"
  Resume Einde_BerekenCorrelatie

End Function

Public Function Hourglass(Optie As String)
    If Optie = "On" Then
      DoCmd.Hourglass True
    Else
      DoCmd.Hourglass False
    End If
End Function

Public Function IsSpace(chrText As String) As Integer
    IsSpace = False
    Select Case Left(chrText, 1)
      Case " ", Chr(9): IsSpace = True
    End Select
End Function
```

```
Public Function SplitString(arrString() As String, strInput As String) As Long
    Dim i As Long, i1 As Long, i2 As Long, iWord As Long
    Dim StrLen As Long

    For i = LBound(arrString) To UBound(arrString)
      arrString(i) = ""
    Next i

    i1 = 1
    i2 = 1
    iWord = 0
    StrLen = Len(strInput)

    ' Go to first non-space character
    While i1 <= StrLen And IsSpace(Mid(strInput, i1, 1))
      i1 = i1 + 1
    Wend
    If i1 > StrLen Then   ' string is empty
      SplitString = 0
      Exit Function
    End If

    ' i1 points to first non-space character
    While i1 <= StrLen
      While i1 <= StrLen And IsSpace(Mid(strInput, i1, 1))
        i1 = i1 + 1
      Wend
      i2 = i1
      ' i2 points to first non-space character
      While i2 <= StrLen And Not IsSpace(Mid(strInput, i2, 1))
        i2 = i2 + 1
      Wend
      'i2 points now to first non-space character of next word or end of string
      iWord = iWord + 1
      If iWord > UBound(arrString) Then
        MsgBox "Overflow in number of arguments: number of words > " & iWord + 1
        i1 = StrLen + 1
      Else
        arrString(iWord) = Mid(strInput, i1, i2 - i1)
        i1 = i2
      End If
    Wend
    SplitString = iWord

End Function

Public Function SplitStringArg(arrString() As String, strInput As String,
SplitChar As String) As Long
    Dim i As Long, i1 As Long, i2 As Long, iWord As Long
    Dim StrLen As Long

    For i = LBound(arrString) To UBound(arrString)
      arrString(i) = ""
    Next i

    i1 = 1
    i2 = 1
    iWord = 0
    StrLen = Len(strInput)

    ' Go to first non-space character
    While i1 <= StrLen And IsSpace(Mid(strInput, i1, 1))
      i1 = i1 + 1
    Wend
    If i1 > StrLen Then   ' string is empty
      SplitStringArg = 0
      Exit Function
    End If
```

```
    While i1 <= StrLen
      While i1 <= StrLen And IsSpace(Mid(strInput, i1, 1))
        i1 = i1 + 1
      Wend
      ' i1 points to first non-space character
      i2 = i1
      ' i2 points to first non-space character
      While i2 <= StrLen And Not Mid(strInput, i2, 1) = SplitChar
        i2 = i2 + 1
      Wend
      'i2 now points to SplitChar or end of string
      iWord = iWord + 1
      If iWord > UBound(arrString) Then
        MsgBox "Overflow in number of arguments: number of words > " & iWord + 1
        i1 = StrLen + 1
      Else
        arrString(iWord) = Trim(Mid(strInput, i1, i2 - i1))
        i1 = i2 + 1 ' skip SplitChar
      End If
    Wend
    SplitStringArg = iWord

End Function

Function TestSplitString()
    Dim ArgArray(100) As String, NumArgs As Long, Tekst As String, i As Long, Msg
As String
    Tekst = "Dit is een zin     met   woorden"
    NumArgs = SplitString(ArgArray, Tekst)
    Msg = "Tekst='" & Tekst & "'" & Chr(13) & Chr(10)
    For i = 1 To NumArgs
      Msg = Msg & "Arg " & i & ": '" & ArgArray(i) & "'" & Chr(13) & Chr(10)
    Next i
    MsgBox Msg, , "Test SplitString"

End Function

Function TestSplitStringArg()
    Dim ArgArray(100) As String, NumArgs As Long, Tekst As String, i As Long, Msg
As String
    Tekst = "Dit, is,een,zin ,met , woorden"
    NumArgs = SplitStringArg(ArgArray, Tekst, ",")
    Msg = "Tekst='" & Tekst & "'" & Chr(13) & Chr(10)
    For i = 1 To NumArgs
      Msg = Msg & "Arg " & i & ": '" & ArgArray(i) & "'" & Chr(13) & Chr(10)
    Next i
    MsgBox Msg, , "Test SplitStringArg"

End Function

Public Function WriteAsciiGrid(FileName As String, QueryName As String, XMin As
Double, XMax As Double, YMin As Double, YMax As Double, CellSize As Double,
MissingValue As Double) As Integer
On Error GoTo Error_WriteAsciiGrid
  Dim filenum As Integer
  filenum = FreeFile
  Open FileName For Output As filenum
  Dim ncols As Long, nrows As Long
  Dim Result As Variant

  ncols = (XMax - XMin) / CellSize
  nrows = (YMax - YMin) / CellSize
  Print #filenum, "ncols"; Tab; ncols
  Print #filenum, "nrows"; Tab; nrows
  Print #filenum, "xllcorner"; Tab; XMin
  Print #filenum, "yllcorner"; Tab; YMin
  Print #filenum, "cellsize"; Tab; CellSize
  Print #filenum, "NODATA_value"; Tab; MissingValue

  Dim Db As Database, RS As Recordset, i As Long, j As Long, irow As Long
```

```
  Set Db = DBEngine.Workspaces(0).Databases(0)
  Result = SysCmd(acSysCmdInitMeter, "Writing Grid: ", nrows)
  Set RS = Db.OpenRecordset(QueryName)
  Result = SysCmd(acSysCmdInitMeter, "Writing Grid: ", nrows)
  irow = 0
  For j = YMax - CellSize To YMin Step -CellSize
    For i = XMin To XMax - CellSize Step CellSize
      If Not RS.EOF Then
        On Error GoTo ResumeAtEndOfSet  ' at end of dataset
        While Not RS.EOF And (RS!Y > j Or (RS!Y = j And RS!X < i))
          RS.MoveNext
        Wend
ResumeAtEndOfSet:
        On Error GoTo Error_WriteAsciiGrid
        If Not RS.EOF Then
          If RS!Y = j And RS!X = i Then
            Print #filenum, RS!Value;
          Else
            Print #filenum, MissingValue;
          End If
        Else
          Print #filenum, MissingValue;
        End If
      Else
        Print #filenum, MissingValue;
      End If
    Next i
    Print #filenum,
    irow = irow + 1
    Result = SysCmd(acSysCmdUpdateMeter, irow)
    DoEvents
  Next j
  RS.Close
  Close #filenum
  Result = SysCmd(acSysCmdRemoveMeter)
  WriteAsciiGrid = 1

End_WriteAsciiGrid:
  Exit Function

Error_WriteAsciiGrid:
  MsgBox "Error in function WriteAsciiGrid: Err=" & Err & ", " & Error
  WriteAsciiGrid = 0
  Resume End_WriteAsciiGrid

End Function

Public Function WriteAsciiGridGeneric(FileName As String, QueryName As String,
XMin As Double, XMax As Double, YMin As Double, YMax As Double, CellSize As
Double, MissingValue As Double, XColumnName As String, YColumnName As String,
ValueColumnName As String) As Integer
On Error GoTo Error_WriteAsciiGridGeneric
  Dim filenum As Integer
  filenum = FreeFile
  Open FileName For Output As filenum
  Dim ncols As Long, nrows As Long
  Dim Result As Variant

  ncols = (XMax - XMin) / CellSize
  nrows = (YMax - YMin) / CellSize
  Print #filenum, "ncols"; Tab; ncols
  Print #filenum, "nrows"; Tab; nrows
  Print #filenum, "xllcorner"; Tab; XMin
  Print #filenum, "yllcorner"; Tab; YMin
  Print #filenum, "cellsize"; Tab; CellSize
  Print #filenum, "NODATA_value"; Tab; MissingValue

  Dim Db As Database, RS As Recordset, i As Long, j As Long, irow As Long
  Set Db = DBEngine.Workspaces(0).Databases(0)
  SysCmd acSysCmdInitMeter, "Writing Grid: ", nrows
```

```
  Set RS = Db.OpenRecordset(QueryName)
  SysCmd acSysCmdInitMeter, "Writing Grid: ", nrows
  irow = 0
  For j = YMax - CellSize To YMin Step -CellSize
    For i = XMin To XMax - CellSize Step CellSize
      If Not RS.EOF Then
        On Error GoTo ResumeAtEndOfSet  ' at end of dataset
        While Not RS.EOF And (RS(YColumnName) > j Or (RS(YColumnName) = j And
RS(XColumnName) < i))
          RS.MoveNext
        Wend
ResumeAtEndOfSet:
        On Error GoTo Error_WriteAsciiGridGeneric
        If Not RS.EOF Then
          If RS(YColumnName) = j And RS(XColumnName) = i Then
            Print #filenum, RS(ValueColumnName);
          Else
            Print #filenum, MissingValue;
          End If
        Else
          Print #filenum, MissingValue;
        End If
      Else
        Print #filenum, MissingValue;
      End If
    Next i
    Print #filenum,
    irow = irow + 1
    SysCmd acSysCmdUpdateMeter, irow
    DoEvents
  Next j
  RS.Close
  Close #filenum
  SysCmd acSysCmdRemoveMeter
  WriteAsciiGridGeneric = 1

End_WriteAsciiGridGeneric:
  Exit Function

Error_WriteAsciiGridGeneric:
  MsgBox "Error in function WriteAsciiGridGeneric: Err=" & Err & ", " & Error
  WriteAsciiGridGeneric = 0
  Resume End_WriteAsciiGridGeneric

End Function


Public Function WriteAsciiGridLine(FileName As String, QueryName As String, XMin
As Double, XMax As Double, YMin As Double, YMax As Double, CellSize As Double,
MissingValue As Double) As Integer

  Dim ncols As Long, nrows As Long
  Dim Result As Variant
  Dim line As String
  Dim filenum As Integer

  filenum = FreeFile

  Open FileName For Output As #filenum

  ncols = (XMax - XMin) / CellSize
  nrows = (YMax - YMin) / CellSize
  Print #filenum, "ncols"; Tab; ncols
  Print #filenum, "nrows"; Tab; nrows
  Print #filenum, "xllcorner"; Tab; XMin
  Print #filenum, "yllcorner"; Tab; YMin
  Print #filenum, "cellsize"; Tab; CellSize
  Print #filenum, "NODATA_value"; Tab; MissingValue

  Dim Db As Database, RS As Recordset, i As Long, j As Long, irow As Long
```

```
  Set Db = DBEngine.Workspaces(0).Databases(0)
  Set RS = Db.OpenRecordset(QueryName)
  Result = SysCmd(acSysCmdInitMeter, "Writing Grid: ", nrows)
  irow = 0
  For j = YMax - CellSize To YMin Step -CellSize
    line = ""
    For i = XMin To XMax - CellSize Step CellSize
      If Not RS.EOF Then
        On Error GoTo ResumeAtEndOfSet  ' at end of dataset
        While Not RS.EOF And (RS!Y > j Or (RS!Y = j And RS!X < i))
          RS.MoveNext
        Wend
ResumeAtEndOfSet:
        If Not RS.EOF Then
          If RS!Y = j And RS!X = i Then
            line = line & RS!Value & " "
          Else
            line = line & MissingValue & " "
          End If
        Else
          line = line & MissingValue & " "
        End If
      Else
        line = line & MissingValue & " "
      End If
    Next i
    Print #filenum, line
    irow = irow + 1
    Result = SysCmd(acSysCmdUpdateMeter, irow)
    DoEvents
  Next j
  RS.Close
  Close #filenum
  Result = SysCmd(acSysCmdRemoveMeter)

End Function

Function ReadAsciiGrid(FileName As String, TableName As String, XColumnName As
String, YColumnName As String, DataFieldName As String, CheckExtraMissingValue As
Integer, ExtraMissingValue As Double, NumberOfFirstLineToRead As Long,
NumberOfLinesToRead As Long)

  If Len(FileName) = 0 Then
    FileName = InputBox("File om uit te lezen:", "Inlezen Ascii-grid")
    If Len(FileName) = 0 Then
      Exit Function
    Else
      MsgBox "Gegevens worden gelezen uit file '" & FileName & "'"
    End If
  End If

  Dim WS As Workspace, MyDb As Database, RS As Recordset
  Dim WriteWS As Workspace
  Dim filenum As Integer

  Set WriteWS = DBEngine.Workspaces(0)
  Set MyDb = WriteWS.Databases(0)
  Set RS = MyDb.OpenRecordset(TableName)

  filenum = FreeFile
  Open FileName For Input Access Read As #filenum
  Dim dumstr As String, dumvar As Variant, dumint As Integer, dumlong As Long

  Dim ncols As Long, nrows As Long, xllcorner As Double, yllcorner As Double,
CellSize As Double, MissingValue As Double
  Dim i As Long, j As Long, X As Double, Y As Double, Value As Double
  Dim Result As Variant

  Input #filenum, dumint, ncols
  Input #filenum, dumint, nrows
```

```
  Input #filenum, dumint, xllcorner
  Input #filenum, dumint, yllcorner
  Input #filenum, dumint, CellSize
  Input #filenum, dumint, MissingValue
  'MsgBox "Asci-grid file header:" & Chr(13) & Chr(10) & _
  '       "ncols: " & ncols & Chr(13) & Chr(10) & _
  '       "nrows: " & nrows & Chr(13) & Chr(10) & _
  '       "xllcorner: " & xllcorner & Chr(13) & Chr(10) & _
  '       "yllcorner: " & yllcorner & Chr(13) & Chr(10) & _
  '       "cellsize: " & CellSize & Chr(13) & Chr(10) & _
  '       "MissingValue: " & MissingValue & Chr(13) & Chr(10)
  Result = SysCmd(acSysCmdInitMeter, "Skipping lines: ", NumberOfFirstLineToRead)
  For i = 1 To NumberOfFirstLineToRead - 1
    For j = 1 To ncols
      Input #filenum, Value
    Next j
    Result = SysCmd(acSysCmdUpdateMeter, i)
  Next i
  Result = SysCmd(acSysCmdRemoveMeter)
  'For i = 1 To 200
  'For i = 1 To nrows
  Dim NumRows As Long
  NumRows   =   IIf(NumberOfLinesToRead   =   0,   nrows   -   NumberOfFirstLineToRead,
min(NumberOfLinesToRead, nrows - NumberOfFirstLineToRead))
  For i = 1 To NumRows
    If i Mod MaxLinesReadInDisplay = 1 Then
      SysCmd acSysCmdInitMeter, "Reading grid (" & FileName & "; rows " & i & "-"
& (i - 1) + MaxLinesReadInDisplay & "): ", MaxLinesReadInDisplay
    End If
    ' Calculate lower left point of current grid
    Y = yllcorner + (nrows - i) * CellSize
    WriteWS.BeginTrans
    For j = 1 To ncols
      X = xllcorner + (j - 1) * CellSize
      Input #filenum, Value
      If   (Value   <>   MissingValue)   And   (CheckExtraMissingValue   =   0   Or
(CheckExtraMissingValue = 1 And Value <> ExtraMissingValue)) Then
        RS.AddNew
        RS(XColumnName) = CLng(X)
        RS(YColumnName) = CLng(Y)
        RS(DataFieldName) = CDbl(Value)
        RS.Update
      End If
      'DoEvents
    Next j
    WriteWS.CommitTrans
    On  Error  Resume  Next    '  door  tegelijkertijd  ander  gebruik  kan  progressbar
verdwenen zijn
    SysCmd acSysCmdUpdateMeter, i Mod MaxLinesReadInDisplay
    On Error GoTo 0
    'DoEvents
  Next i

  Result = SysCmd(acSysCmdRemoveMeter)

  RS.Close
  Close #filenum
End Function

Function  ReadAsciiGridInMemory(ValArray()  As  Double,  XValArray()  As  Double,
YValArray() As Double, FileName As String) As Integer
On Error GoTo error_ReadAsciiGridInMemory
  Dim filenum As Integer
  Dim dumint As Integer
  Dim  ncols  As  Long,  nrows  As  Long,  xllcorner  As  Double,  yllcorner  As  Double,
CellSize As Double, MissingValue As Double
  Dim irow As Long, icol As Long, Value As Double

  If Len(FileName) = 0 Then
    ReadAsciiGridInMemory = False
```

```
    Exit Function
  End If

  filenum = FreeFile
  Open FileName For Input Access Read As #filenum

  Input #filenum, dumint, ncols
  Input #filenum, dumint, nrows
  Input #filenum, dumint, xllcorner
  Input #filenum, dumint, yllcorner
  Input #filenum, dumint, CellSize
  Input #filenum, dumint, MissingValue

  ReDim ValArray(ncols, nrows)
  ReDim XValArray(ncols)
  ReDim YValArray(nrows)

  Dim NumRows As Long
  For icol = 1 To ncols
    XValArray(icol) = xllcorner + (icol - 1) * CellSize
  Next icol
  For irow = 1 To nrows
    YValArray(irow) = yllcorner + (irow - 1) * CellSize
  Next irow

  For irow = nrows To 1 Step -1
    For icol = 1 To ncols
      Input #filenum, Value
      ValArray(icol, irow) = Value
    Next icol
  Next irow
  Close #filenum

  ReadAsciiGridInMemory = True

end_ReadAsciiGridInMemory:
  Exit Function

error_ReadAsciiGridInMemory:
  MsgBox "Error in function ReadAsciiGridInMemory(FileName:'" & FileName & "):
Err:" & Err & "; " & Err.Description
  Resume end_ReadAsciiGridInMemory
End Function


Function Pi() As Double
  Pi = 3.141592654
End Function


Function EraseTable(TableName As String)
    WisTabel (TableName)
End Function

Function WisTabel(TabelNaam As String)
    Dim MyQD As QueryDef, sql As String
    sql = "DELETE [" & TabelNaam & "].* FROM [" & TabelNaam & "];"
    Set MyQD = CurrentDb.CreateQueryDef("", sql)
    On Error Resume Next
    MyQD.Execute dbFailOnError
    If Err > 0 Then
      MsgBox "Wissen van inhoud van tabel '" & TabelNaam & "' niet sucesvol"
    End If
    Err = 0
    MyQD.Close
    Set MyQD = Nothing
End Function

Function WisTabelWhere(TabelNaam As String, WhereClause As String)
    Dim MyQD As QueryDef, sql As String
```

```
    sql = "DELETE [" & TabelNaam & "].* FROM [" & TabelNaam & "] " & _
          "WHERE(" & WhereClause & ");"
    Set MyQD = CurrentDb.CreateQueryDef("", sql)
    On Error Resume Next
    MyQD.Execute dbFailOnError
    If Err > 0 Then
      MsgBox "Wissen van inhoud van tabel '" & TabelNaam & "' niet sucesvol"
    End If
    Err = 0
    MyQD.Close
    Set MyQD = Nothing
End Function

Public Function CalculatePercentile(ByRef MySet As Recordset, Percentile As
Integer) As Double
    CalculatePercentile = BerekenPercentiel(MySet, Percentile)
End Function

Public Function BerekenPercentielMetQueryNaam(QueryNaam As String, Percentiel As
Integer) As Double
    Dim MySet As Recordset
    Set MySet = CurrentDb.OpenRecordset(QueryNaam)
    BerekenPercentielMetQueryNaam = BerekenPercentiel(MySet, Percentiel)
    MySet.Close
End Function

Public Function BerekenPercentielMetSQL(sql As String, Percentiel As Integer) As
Double
    Dim MySet As Recordset
    Set MySet = CurrentDb.OpenRecordset(sql)
    BerekenPercentielMetSQL = BerekenPercentiel(MySet, Percentiel)
    MySet.Close
End Function

Public Function BerekenPercentiel(ByRef dataset As Recordset, Percentiel As
Integer) As Double
    ' DataSet: gesorteerd (oplopend); Veld: Waarde
    ' Percentiel: 0-100
    ' Nieuwe versie 19-5-2003
    Dim Aantal As Long, PercentielWaarde As Double
    dataset.MoveLast
    Aantal = dataset.RecordCount
    PercentielWaarde = Percentiel
    If PercentielWaarde >= 100 Then
      BerekenPercentiel = dataset!Waarde   ' currently last record
      Exit Function
    End If
    If PercentielWaarde <= 0 Then
      dataset.MoveFirst
      BerekenPercentiel = dataset!Waarde   ' currently first record
      Exit Function
    End If
    PercentielWaarde = Aantal * (Percentiel / 100#)
    If PercentielWaarde = Int(PercentielWaarde) Then    ' whole number: get Waarde
directly from recordset
      dataset.AbsolutePosition = Int(PercentielWaarde) - 1 ' absolute position is
zero-based
      BerekenPercentiel = dataset!Waarde
    Else  ' interpolate
      Dim X1 As Double, X2 As Double
      dataset.AbsolutePosition = min(max(Int(PercentielWaarde), 0), Aantal - 1)  '
absolute position is zero-based
      X1 = dataset!Waarde
      dataset.MoveNext
      If Not dataset.EOF Then
        X2 = dataset!Waarde
        BerekenPercentiel = (1 - (PercentielWaarde - Int(PercentielWaarde))) * X1
+ _
                               (PercentielWaarde - Int(PercentielWaarde)) * X2
      Else
```

```
        BerekenPercentiel = X1
      End If
    End If
End Function

Function JaarDMJ(dag As Integer, maand As Integer, jaar As Integer) As Double
    JaarDMJ = jaar + DatePart("y", DateSerial(jaar, maand, dag)) / DatePart("y",
DateSerial(jaar, 12, 31))
End Function

Function JaarMJ(maand As Integer, jaar As Integer) As Double
    JaarMJ = jaar + DatePart("y", DateSerial(jaar, maand, 15)) / DatePart("y",
DateSerial(jaar, 12, 31))
End Function

Function JaarKJ(Kwartaal As Integer, jaar As Integer) As Double
    JaarKJ = jaar + (Kwartaal - 1) * 0.25 + 0.125
End Function

Function JaarDecJ(Decade As Integer, jaar As Integer) As Double
    Dim maand As Integer, dec As Integer, dag As Integer
    maand = (Decade - 1) \ 3 + 1
    dec = Decade Mod 3
    Select Case dec
      Case 0: dag = 25
      Case 1: dag = 5
      Case 2: dag = 15
    End Select
    JaarDecJ = jaar + DatePart("y", DateSerial(jaar, maand, dag)) / DatePart("y",
DateSerial(jaar, 12, 31))
End Function

Function JaarDatum(datum As Date) As Double
    JaarDatum = Year(datum) + DatePart("y", datum) / DatePart("y",
DateSerial(Year(datum), 12, 31))
End Function

Function Kwartaal(datum As Date) As Integer
    Kwartaal = Int((month(datum) - 1) / 3) + 1
End Function

Function Halfjaar(datum As Date) As String
    Halfjaar = IIf(month(Date) >= 4 And month(Date) <= 9, "Z", "W")
End Function

Function MaandDec(Decade As Integer) As Integer
    MaandDec = Int((Decade - 1) / 3) + 1
End Function

Function KwartaalDec(Decade As Integer) As Integer
    KwartaalDec = Int((MaandDec(Decade) - 1) / 3) + 1
End Function

Function HalfjaarDec(Decade As Integer) As String
    HalfjaarDec = IIf(KwartaalDec(Decade) = 2 Or KwartaalDec(Decade) = 3, "Z",
"W")
End Function

Function Decade(datum As Date) As Integer
    Dim m As Integer, d As Integer
    m = month(datum)
    d = DatePart("d", datum)
    If d <= 10 Then
      Decade = (m - 1) * 3 + 1
    ElseIf d <= 20 Then
      Decade = (m - 1) * 3 + 2
    Else
      Decade = (m - 1) * 3 + 3
    End If
End Function
```

```
Function RunAccessFunctionExternal(DatabaseName As String, FunctionName As String,
Arguments As String, Types As String)
    Dim msa As Application
    Set msa = CreateObject("access.application")
    msa.OpenCurrentDatabase (DatabaseName)
    msa.Visible = False
    RunAccessFunctionExternalActual msa, FunctionName, Arguments, Types
    Set msa = Nothing
End Function

Function  RunAccessFunctionExternalVisible(DatabaseName  As  String,  FunctionName  As
String, Arguments As String, Types As String)
    Dim msa As Application
    Set msa = CreateObject("access.application")
    msa.OpenCurrentDatabase (DatabaseName)
    msa.Visible = True
    RunAccessFunctionExternalActual msa, FunctionName, Arguments, Types
    Set msa = Nothing
End Function

Function  RunAccessFunctionExternalNotVisible(DatabaseName  As  String,  FunctionName
As String, Arguments As String, Types As String)
    Dim msa As Application
    Set msa = CreateObject("access.application")
    msa.OpenCurrentDatabase (DatabaseName)
    msa.Visible = False
    RunAccessFunctionExternalActual msa, FunctionName, Arguments, Types
    Set msa = Nothing
End Function

Function  RunAccessFunctionExternalActual(msa  As  Application,  FunctionName  As
String, Arguments As String, Types As String)
    Dim  Arg(100)  As  String,  NumArgs  As  Integer,  ArgType(100)  As  String,
NumArgTypes As Integer
    NumArgs = SplitStringArg(Arg, Arguments, ",")
    NumArgTypes = SplitStringArg(ArgType, Types, ",")

    Select Case NumArgs
      Case 0:   msa.Run FunctionName
      Case 1:   msa.Run FunctionName, _
                                      IIf(ArgType(1)   =   "Num",   CDbl(Arg(1)),
Arg(1))
      Case 2:   msa.Run FunctionName, _
                                      IIf(ArgType(1)   =   "Num",   CDbl(Arg(1)),
Arg(1)), _
                                      IIf(ArgType(2)   =   "Num",   CDbl(Arg(2)),
Arg(2))
      Case 3:   msa.Run FunctionName, _
                                      IIf(ArgType(1)   =   "Num",   CDbl(Arg(1)),
Arg(1)), _
                                      IIf(ArgType(2)   =   "Num",   CDbl(Arg(2)),
Arg(2)), _
                                      IIf(ArgType(3)   =   "Num",   CDbl(Arg(3)),
Arg(3))
      Case 4:   msa.Run FunctionName, _
                                      IIf(ArgType(1)   =   "Num",   CDbl(Arg(1)),
Arg(1)), _
                                      IIf(ArgType(2)   =   "Num",   CDbl(Arg(2)),
Arg(2)), _
                                      IIf(ArgType(3)   =   "Num",   CDbl(Arg(3)),
Arg(3)), _
                                      IIf(ArgType(4)   =   "Num",   CDbl(Arg(4)),
Arg(4))
      Case 5:   msa.Run FunctionName, _
                                      IIf(ArgType(1)   =   "Num",   CDbl(Arg(1)),
Arg(1)), _
                                      IIf(ArgType(2)   =   "Num",   CDbl(Arg(2)),
Arg(2)), _
```

```
                                      IIf(ArgType(3)    =    "Num",    CDbl(Arg(3)),
Arg(3)), _
                                      IIf(ArgType(4)    =    "Num",    CDbl(Arg(4)),
Arg(4)), _
                                      IIf(ArgType(5)    =    "Num",    CDbl(Arg(5)),
Arg(5))
      Case 6:   msa.Run FunctionName, _
                                      IIf(ArgType(1)    =    "Num",    CDbl(Arg(1)),
Arg(1)), _
                                      IIf(ArgType(2)    =    "Num",    CDbl(Arg(2)),
Arg(2)), _
                                      IIf(ArgType(3)    =    "Num",    CDbl(Arg(3)),
Arg(3)), _
                                      IIf(ArgType(4)    =    "Num",    CDbl(Arg(4)),
Arg(4)), _
                                      IIf(ArgType(5)    =    "Num",    CDbl(Arg(5)),
Arg(5)), _
                                      IIf(ArgType(6)    =    "Num",    CDbl(Arg(6)),
Arg(6))
      Case 7:   msa.Run FunctionName, _
                                      IIf(ArgType(1)    =    "Num",    CDbl(Arg(1)),
Arg(1)), _
                                      IIf(ArgType(2)    =    "Num",    CDbl(Arg(2)),
Arg(2)), _
                                      IIf(ArgType(3)    =    "Num",    CDbl(Arg(3)),
Arg(3)), _
                                      IIf(ArgType(4)    =    "Num",    CDbl(Arg(4)),
Arg(4)), _
                                      IIf(ArgType(5)    =    "Num",    CDbl(Arg(5)),
Arg(5)), _
                                      IIf(ArgType(6)    =    "Num",    CDbl(Arg(6)),
Arg(6)), _
                                      IIf(ArgType(7)    =    "Num",    CDbl(Arg(7)),
Arg(7))
      Case 8:   msa.Run FunctionName, _
                                      IIf(ArgType(1)    =    "Num",    CDbl(Arg(1)),
Arg(1)), _
                                      IIf(ArgType(2)    =    "Num",    CDbl(Arg(2)),
Arg(2)), _
                                      IIf(ArgType(3)    =    "Num",    CDbl(Arg(3)),
Arg(3)), _
                                      IIf(ArgType(4)    =    "Num",    CDbl(Arg(4)),
Arg(4)), _
                                      IIf(ArgType(5)    =    "Num",    CDbl(Arg(5)),
Arg(5)), _
                                      IIf(ArgType(6)    =    "Num",    CDbl(Arg(6)),
Arg(6)), _
                                      IIf(ArgType(7)    =    "Num",    CDbl(Arg(7)),
Arg(7)), _
                                      IIf(ArgType(8)    =    "Num",    CDbl(Arg(8)),
Arg(8))
      Case 9:   msa.Run FunctionName, _
                                      IIf(ArgType(1)    =    "Num",    CDbl(Arg(1)),
Arg(1)), _
                                      IIf(ArgType(2)    =    "Num",    CDbl(Arg(2)),
Arg(2)), _
                                      IIf(ArgType(3)    =    "Num",    CDbl(Arg(3)),
Arg(3)), _
                                      IIf(ArgType(4)    =    "Num",    CDbl(Arg(4)),
Arg(4)), _
                                      IIf(ArgType(5)    =    "Num",    CDbl(Arg(5)),
Arg(5)), _
                                      IIf(ArgType(6)    =    "Num",    CDbl(Arg(6)),
Arg(6)), _
                                      IIf(ArgType(7)    =    "Num",    CDbl(Arg(7)),
Arg(7)), _
                                      IIf(ArgType(8)    =    "Num",    CDbl(Arg(8)),
Arg(8)), _
                                      IIf(ArgType(9)    =    "Num",    CDbl(Arg(9)),
Arg(9))
```

```
      Case 10:  msa.Run FunctionName, _
                                      IIf(ArgType(1) = "Num", CDbl(Arg(1)),
Arg(1)), _
                                      IIf(ArgType(2) = "Num", CDbl(Arg(2)),
Arg(2)), _
                                      IIf(ArgType(3) = "Num", CDbl(Arg(3)),
Arg(3)), _
                                      IIf(ArgType(4) = "Num", CDbl(Arg(4)),
Arg(4)), _
                                      IIf(ArgType(5) = "Num", CDbl(Arg(5)),
Arg(5)), _
                                      IIf(ArgType(6) = "Num", CDbl(Arg(6)),
Arg(6)), _
                                      IIf(ArgType(7) = "Num", CDbl(Arg(7)),
Arg(7)), _
                                      IIf(ArgType(8) = "Num", CDbl(Arg(8)),
Arg(8)), _
                                      IIf(ArgType(9) = "Num", CDbl(Arg(9)),
Arg(9)), _
                                      IIf(ArgType(10) = "Num", CDbl(Arg(10)),
Arg(10))
      Case Else
        MsgBox "Too many arguments in function RunAccessFunctionExternalActual();
expand select statement"
    End Select
End Function

Function TestRunAccessFunctionExternal()
    RunAccessFunctionExternalVisible "G:\AccessLib\AccessLib.mdb", "JaarKJ",
"1,1990", ""
End Function


Function rinstr(Tekst As String, ZoekTekst As String) As Integer
    Dim i As Integer, Found As Integer
    i = Len(Tekst)
    While i > 0 And Not Found
      If InStr(i, Tekst, ZoekTekst) = i Then
        Found = True
      Else
        i = i - 1
      End If
    Wend
    rinstr = i
End Function
```

Appendix 5 Validation scores predicted POO and Presence higher than 4

Taxon ID	Total validation POO	Total validation Presence	Name	Dutch name
16	4	4	Aira praecox	Vroege haver
23	6	4	Allium ursinum	Daslook
32	4	6	Althaea officinalis	Echte heemst
34	4	4	Ammophila arenaria	Helm
38	6	5	Andromeda polifolia	Lavendelhei
48	4	4	Anthyllis vulneraria	Wondklaver
58	6	6	Armeria maritima	Engels gras
61	4	4	Artemisia campestris subsp. maritima	Duinaveruit
62	6	6	Artemisia maritima	Zeealsem
64	6	4	Arum maculatum	Gevlekte aronskelk
67	4	4	Galium odoratum	Lievevrouwebedstro
68	6	6	Asplenium ruta-muraria	Muurvaren
69	4	4	Aster tripolium	Zulte
74	6	4	Azolla filiculoides	Grote kroosvaren
88	8	4	Brachypodium pinnatum	Gevinde kortsteel
96	4	4	Bupleurum tenuissimum	Fijn goudscherm
98	8	4	Cakile maritima	Zeeraket
104	6	4	Callitriche obtusangula	Stomphoekig sterrekroos
105	6	4	Callitriche platycarpa	Gewoon sterrekroos
107	8	6	Calluna vulgaris	Struikhei
112	4	4	Campanula trachelium	Ruig klokje
128	6	7	Carex diandra	Ronde zegge
129	4	4	Carex distans	Zilte zegge
131	4	4	Carex echinata	Sterzegge
133	6	4	Carex extensa	Kwelderzegge
136	6	4	Carex hostiana	Blonde zegge
140	4	4	Carex cuprina	Valse voszegge
144	6	4	Carex paniculata	Pluimzegge
145	4	4	Carex pilulifera	Pilzegge
146	4	4	Carex pseudocyperus	Hoge cyperzegge
147	6	4	Carex pulicaris	Vlozegge
154	6	4	Carex trinervis	Drienervige zegge
160	4	4	Catabrosa aquatica	Watergras
162	4	4	Centaurea scabiosa	Grote centaurie
172	4	4	Ceratophyllum demersum	Grof hoornblad
173	6	4	Ceratophyllum submersum	Fijn hoornblad
185	4	4	Cichorium intybus	Wilde cichorei
190	4	4	Cirsium dissectum	Spaanse ruiter
204	4	4	Corydalis solida	Vingerhelmbloem
206	8	4	Corynephorus canescens	Buntgras
212	5	5	Crepis vesicaria	Paardebloemstreepzaad
216	4	4	Cynoglossum officinale	Veldhondstong
223	6	4	Dianthus deltoides	Steenanjer
229	6	4	Drosera rotundifolia	Ronde zonnedauw
234	4	4	Thelypteris palustris	Moerasvaren
239	6	6	Eleocharis multicaulis	Veelstengelige waterbies
242	4	4	Eleocharis palustris subsp. uniglumis	Slanke waterbies
243	4	4	Elodea canadensis	Brede waterpest

Taxon ID	Total validation POO	Total validation Presence	Name	Dutch name
246	6	4	Elymus farctus	Biestarwegras
264	6	5	Equisetum variegatum	Bonte paardestaart
265	4	5	Erica tetralix	Gewone dophei
266	4	5	Erigeron acris	Scherpe fijnstraal
269	6	4	Eriophorum vaginatum	Eenarig wollegras
272	4	4	Erodium cicutarium subsp. dunense	Duinreigersbek
275	4	4	Eryngium maritimum	Blauwe zeedistel
297	4	4	Galanthus nivalis	Gewoon sneeuwklokje
307	6	5	Galium pumilum	Kalkwalstro
310	4	4	Genista anglica	Stekelbrem
314	6	4	Gentianella germanica	Duitse gentiaan
315	8	4	Gentiana pneumonanthe	Klokjesgentiaan
321	6	4	Glaux maritima	Melkkruid
332	6	4	Atriplex portulacoides	Gewone zoutmelde
348	6	4	Honckenya peploides	Zeepostelein
352	4	4	Hottonia palustris	Waterviolier
354	6	4	Hydrocharis morsus-ranae	Kikkerbeet
382	6	4	Juncus gerardi	Zilte rus
384	6	4	Juncus maritimus	Zeerus
396	4	4	Lamium maculatum	Gevlekte dovenetel
404	4	4	Lemna trisulca	Puntkroos
410	6	4	Limonium vulgare	Lamsoor
415	4	4	Linum catharticum	Geelhartje
420	8	6	Lobelia dortmanna	Waterlobelia
460	4	4	Mercurialis perennis	Bosbingelkruid
462	4	6	Milium effusum	Bosgierstgras
471	4	4	Myosotis sylvatica	Bosvergeet-mij-nietje
474	6	4	Myrica gale	Wilde gagel
478	6	4	Najas marina	Groot nimfkruid
480	6	4	Narthecium ossifragum	Beenbreek
481	4	4	Rorippa microphylla	Slanke waterkers
483	4	4	Nuphar lutea	Gele plomp
484	4	4	Nymphaea alba	Witte waterlelie
485	4	4	Nymphoides peltata	Watergentiaan
488	4	4	Oenanthe lachenalii	Zilt torkruid
495	6	4	Orchis militaris	Soldaatje
498	6	5	Origanum vulgare	Wilde marjolein
505	6	6	Oxycoccus macrocarpos	Grote veenbes
506	4	4	Oxycoccus palustris	Kleine veenbes
510	6	4	Parapholis strigosa	Dunstaart
511	4	4	Paris quadrifolia	Eenbes
512	6	6	Parnassia palustris	Parnassia
534	6	4	Plantago maritima	Zeeweegbree
538	5	4	Poa compressa	Plat beemdgras
542	4	4	Polygala comosa	Kuifvleugeltjesbloem
561	4	4	Potamogeton acutifolius	Spits fonteinkruid
565	6	4	Potamogeton crispus	Gekroesd fonteinkruid
566	4	4	Groenlandia densa	Paarbladig fonteinkruid
570	6	4	Potamogeton natans	Drijvend fonteinkruid
576	4	4	Potamogeton trichoides	Haarfonteinkruid
584	6	4	Potentilla verna	Voorjaarsganzerik
594	6	4	Puccinellia fasciculata	Blauw kweldergras
595	4	4	Puccinellia maritima	Gewoon kweldergras
598	8	7	Quercus petraea	Wintereik

Taxon ID	Total validation POO	Total validation Presence	Name	Dutch name
606	6	4	Ranunculus circinatus	Stijve waterranonkel
620	6	4	Rhynchospora alba	Witte snavelbies
621	8	6	Rhynchospora fusca	Bruine snavelbies
633	6	4	Rumex acetosella	Schapezuring
641	4	4	Rumex sanguineus	Bloedzuring
644	6	5	Sagina maritima	Zeevetmuur
647	6	4	Sagittaria sagittifolia	Pijlkruid
648	6	4	Salicornia europaea + Salicornia procumbens	Kortarige + Langarige zeekraal
661	6	4	Salvia pratensis	Veldsalie
665	4	4	Sanguisorba minor	Kleine pimpernel
667	4	4	Sanicula europaea	Heelkruid
673	6	4	Scabiosa columbaria	Duifkruid
676	6	4	Scirpus fluitans	Vlottende bies
678	6	4	Scirpus maritimus	Heen
680	6	4	Scirpus rufus	Rode bies
685	6	4	Scleranthus perennis	Overblijvende hardbloem
691	6	6	Sedum reflexum	Tripmadam
699	4	4	Senecio sylvaticus	Boskruiskruid
706	4	4	Silene otites	Oorsilene
707	4	4	Silene vulgaris	Blaassilene
709	4	4	Sisymbrium altissimum	Hongaarse raket
723	6	6	Spartina townsendii	Engels slijkgras
726	6	4	Spergularia maritima	Gerande schijnspurrie
728	6	4	Spergularia salina	Zilte schijnspurrie
729	6	4	Spirodela polyrhiza	Veelwortelig kroos
739	6	4	Stratiotes aloides	Krabbescheer
740	6	5	Suaeda maritima	Schorrekruid
750	4	4	Teesdalia nudicaulis	Klein tasjeskruid
755	6	4	Thymus serpyllum	Wilde tijm
756	4	4	Tilia cordata	Winterlinde
770	4	4	Typha angustifolia	Kleine lisdodde
777	4	4	Utricularia vulgaris	Groot blaasjeskruid
779	4	4	Vaccinium vitis-idaea	Rode bosbes
784	4	4	Verbascum thapsus	Koningskaars
811	6	4	Viola curtisii	Duinviooltje
813	4	4	Viola odorata	Maarts viooltje
817	6	4	Viola rupestris	Zandviooltje
821	4	4	Wolffia arrhiza	Wortelloos kroos
822	6	4	Zannichellia palustris subsp. palustris	Zittende zannichellia
823	6	4	Zannichellia palustris subsp. pedicellata	Gesteelde zannichellia
824	4	4	Phleum pratense subsp. bertolonii	Klein timoteegras
837	6	4	Salicornia europaea	Kortarige zeekraal
838	6	5	Salicornia procumbens	Langarige zeekraal
864	4	4	Scirpus lacustris	Mattenbies s.l.
867	6	6	Zannichellia palustris	Zannichellia
882	5	4	Nitella mucronata	Puntdragend glanswier
883	4	4	Nitellopsis obtusa	Sterkranswier
904	4	4	Arenaria serpyllifolia	Zandmuur
908	4	4	Scirpus cespitosus	Veenbies s.l.

Appendix 6 Number of findings, model terms, interactions and quadratic terms

TaxonNr	# findings	# terms	# interactions	# quadratic terms
685	51	8	1	1
905	52	10	2	4
583	56	6	1	2
42	57	7	1	2
442	67	7	1	1
818	67	10	1	3
68	68	8	1	2
793	72	8	1	1
22	77	9	1	2
75	78	9	2	2
408	80	8	1	2
50	85	12	1	3
475	88	9	2	3
745	88	10	3	3
679	89	13	2	4
43	89	8	1	2
182	91	9	1	3
494	96	8	1	3
757	97	6	1	1
849	101	6	1	1
832	102	10	1	2
446	105	13	1	4
188	111	9	1	2
903	111	10	1	3
36	114	10	1	3
166	115	12	2	2
313	115	12	2	3
694	117	6	1	2
178	119	8	1	2
604	120	13	2	4
368	122	15	3	4
856	129	7	1	2
106	139	12	2	4
543	142	6	1	2
261	142	11	3	2
465	144	11	2	3
714	147	14	5	3
299	147	9	2	2
279	149	10	2	4
333	151	8	1	2
721	152	8	1	2

TaxonNr	# findings	# terms	# interactions	# quadratic terms
863	156	7	1	3
275	157	15	1	5
304	159	11	1	3
302	164	10	1	2
501	164	10	1	3
660	165	9	1	2
653	169	11	1	2
295	173	10	1	2
330	173	10	2	3
709	174	8	1	3
338	178	12	1	4
90	181	11	3	4
401	186	11	2	4
197	187	14	1	4
123	187	13	3	4
839	188	17	4	5
211	189	14	4	5
343	192	8	1	1
787	195	11	1	2
640	197	6	1	2
488	197	14	2	5
673	198	8	1	1
264	200	17	6	7
30	200	13	2	2
350	206	9	1	2
394	211	9	1	1
283	212	9	1	2
38	214	6	1	1
241	215	12	1	3
807	216	11	1	2
577	216	5	1	2
160	217	11	1	3
480	217	8	2	2
236	218	10	2	3
164	220	12	3	4
455	226	10	1	4
389	226	8	1	2
502	228	9	2	2
416	228	15	3	4
536	229	10	2	2
489	232	10	1	2
72	235	7	1	3
185	235	10	2	3
15	236	9	2	2
492	236	12	2	3
676	246	9	2	3
147	247	8	1	3

TaxonNr	# findings	# terms	# interactions	# quadratic terms
102	248	11	3	2
266	248	10	1	3
298	249	14	3	3
715	251	13	2	2
119	251	9	2	2
753	251	10	3	2
758	254	11	2	3
18	255	11	3	5
175	256	10	1	2
603	262	14	3	4
598	266	14	3	3
153	269	13	1	3
515	270	9	1	2
421	271	12	3	2
280	277	11	2	1
226	279	10	1	2
909	281	14	3	5
161	284	12	2	4
718	286	12	4	2
861	287	15	4	4
869	288	14	3	5
451	288	7	1	1
425	293	18	5	6
194	299	16	4	3
177	299	11	2	1
574	300	8	1	2
136	304	16	2	4
269	306	9	1	2
571	307	14	4	5
834	317	14	3	4
270	317	14	4	2
345	317	11	3	6
35	324	9	1	1
508	325	11	1	3
235	325	12	2	2
224	327	11	1	2
621	331	11	1	3
40	338	10	2	3
472	344	7	1	2
419	345	15	5	5
433	350	9	2	1
245	363	12	1	2
84	368	14	5	5
429	375	11	1	3
163	378	20	5	7
57	379	12	3	4
708	379	15	6	5

TaxonNr	# findings	# terms	# interactions	# quadratic terms
128	384	15	4	5
253	386	11	2	2
125	386	17	4	5
527	387	14	2	4
506	388	11	2	3
388	390	14	4	3
417	393	12	3	5
152	397	15	2	4
246	399	14	5	4
88	406	6	1	1
822	407	10	2	4
674	413	12	1	4
196	416	13	1	4
829	417	19	7	5
73	421	8	1	1
638	425	17	5	5
195	425	12	4	6
760	426	16	5	4
180	431	14	3	4
509	434	15	2	2
776	435	15	3	4
558	436	10	3	1
684	442	16	5	6
867	443	16	4	6
613	443	17	4	5
165	447	9	1	3
821	448	8	1	2
566	452	13	2	4
666	453	20	5	6
593	455	11	1	4
19	457	11	2	2
645	462	18	4	4
759	463	12	2	4
12	469	17	5	5
58	470	14	3	5
664	471	25	9	6
774	473	12	2	3
156	475	13	1	3
620	476	12	3	4
46	478	11	1	3
772	479	14	3	3
33	479	12	1	4
806	484	16	3	4
585	492	12	1	3
89	494	13	1	3
131	495	11	2	3
239	500	12	2	5

TaxonNr	# findings	# terms	# interactions	# quadratic terms
596	502	15	4	4
824	504	14	4	5
129	506	13	3	5
310	506	15	4	4
567	507	9	1	3
682	508	14	3	3
912	510	14	4	3
238	520	14	3	4
672	522	8	1	1
499	524	16	6	4
514	527	16	4	5
797	528	12	1	3
49	535	16	3	6
127	542	12	2	4
611	544	13	1	4
908	549	8	1	2
551	549	14	2	2
555	551	12	1	3
65	551	11	2	3
199	551	17	6	4
55	557	11	1	3
366	558	14	4	4
497	560	16	4	4
842	561	16	3	5
93	564	11	1	2
397	568	9	1	1
237	572	9	1	2
462	573	10	1	2
64	575	13	2	4
813	576	21	8	6
358	584	16	5	5
7	584	13	1	3
91	587	17	7	5
789	597	12	1	3
665	604	11	2	4
658	606	15	5	6
115	610	13	3	3
399	610	20	6	5
258	611	14	1	5
361	611	15	3	3
512	620	15	3	6
228	626	10	1	3
624	628	12	1	2
750	639	13	5	4
390	644	14	3	4
535	644	10	1	2
619	660	15	3	3

TaxonNr	# findings	# terms	# interactions	# quadratic terms
683	662	20	7	8
365	663	16	4	5
257	670	20	8	5
769	673	15	4	6
693	675	16	4	5
546	680	14	2	3
723	683	11	2	4
190	686	11	1	3
140	687	16	3	5
866	688	20	6	4
767	690	19	6	6
498	701	17	5	4
315	705	15	4	5
914	706	13	2	3
530	707	16	3	4
174	708	15	4	2
779	713	19	7	7
798	717	11	1	2
450	722	11	1	2
617	725	20	7	7
792	725	13	4	3
540	733	17	5	6
840	738	13	3	4
385	747	16	3	5
138	749	17	5	5
208	755	15	3	5
155	758	13	1	4
698	760	18	6	5
1	762	17	7	4
733	774	16	4	5
553	777	15	6	6
651	779	15	4	5
669	785	19	7	4
479	785	20	7	6
292	793	14	2	4
114	795	16	5	4
447	796	12	1	2
491	799	20	9	7
457	806	15	4	3
610	816	14	5	6
677	817	12	3	4
724	826	15	3	3
641	827	11	2	1
111	840	18	7	6
816	842	19	7	6
889	845	10	2	4
513	866	11	1	3

TaxonNr	# findings	# terms	# interactions	# quadratic terms
835	870	21	7	6
897	871	27	11	7
249	872	16	4	4
157	878	17	4	4
151	880	15	3	7
395	883	20	6	6
710	898	13	4	1
648	909	7	1	2
893	913	13	2	4
780	929	13	1	3
762	932	24	10	8
82	935	20	7	4
812	937	24	11	9
287	941	15	3	5
739	959	12	2	4
717	972	16	4	3
118	973	11	2	3
659	980	15	4	4
316	984	15	3	4
751	991	19	7	4
857	1010	16	5	5
888	1017	11	2	4
833	1018	21	8	8
370	1031	13	2	3
141	1036	16	4	3
286	1043	13	3	3
415	1046	20	7	8
490	1052	11	2	5
187	1057	13	2	4
484	1060	12	1	3
105	1060	11	1	4
695	1061	26	12	9
544	1061	20	5	5
126	1071	13	1	4
803	1081	16	3	4
737	1088	11	1	3
874	1105	13	3	3
329	1128	14	3	4
777	1137	10	1	4
10	1139	22	9	7
790	1141	15	3	3
85	1149	17	5	7
576	1151	11	1	3
398	1161	10	1	1
17	1162	22	8	8
256	1169	14	4	4
565	1178	15	2	5

TaxonNr	# findings	# terms	# interactions	# quadratic terms
332	1181	8	1	2
201	1184	17	4	6
351	1186	20	6	5
232	1188	10	2	2
752	1188	15	4	5
273	1198	14	4	3
626	1202	22	9	8
5	1208	15	4	4
901	1216	13	5	6
503	1227	15	3	4
810	1230	16	3	4
186	1237	14	4	4
716	1241	16	4	6
487	1257	10	1	3
531	1261	20	8	8
467	1265	18	6	5
459	1269	14	4	5
534	1278	16	4	4
466	1282	18	8	6
255	1284	20	7	5
766	1287	14	2	3
809	1303	20	5	4
95	1303	18	6	6
635	1305	15	3	5
627	1312	13	5	5
481	1327	14	3	2
804	1330	15	3	3
274	1348	19	6	6
229	1354	15	5	5
712	1357	12	1	2
267	1358	17	5	2
4	1362	20	7	7
393	1367	13	2	4
722	1378	14	3	4
591	1399	25	10	3
650	1417	12	4	5
137	1423	14	2	3
595	1436	9	1	3
445	1440	16	3	3
444	1443	20	8	9
588	1448	17	4	4
339	1449	21	8	8
474	1467	19	7	5
305	1468	12	2	4
831	1479	22	10	10
134	1486	18	3	5
735	1493	12	2	4

TaxonNr	# findings	# terms	# interactions	# quadratic terms
741	1494	25	10	10
880	1498	10	2	2
618	1504	17	4	4
132	1521	12	3	3
63	1551	22	8	6
573	1584	19	9	9
476	1588	11	2	6
801	1606	27	11	9
732	1618	17	2	4
606	1620	15	6	6
904	1622	15	3	6
687	1648	18	6	6
646	1659	21	9	6
910	1661	13	2	3
386	1666	25	12	8
308	1667	17	5	4
528	1673	18	5	5
352	1691	13	4	5
623	1694	17	4	5
525	1695	20	6	6
592	1696	19	6	7
443	1698	23	10	3
409	1711	21	7	5
560	1712	12	3	6
414	1715	18	5	5
21	1771	21	6	4
622	1773	18	6	5
321	1787	22	8	7
24	1797	23	9	8
402	1797	19	9	7
181	1798	26	10	9
539	1801	19	4	6
902	1804	24	9	7
734	1808	15	4	5
678	1848	17	5	5
319	1860	16	3	3
116	1863	16	5	3
382	1864	16	3	4
243	1865	23	10	8
814	1873	15	4	7
743	1873	26	11	8
363	1874	14	1	3
176	1917	8	2	1
907	1924	13	1	3
605	1925	21	8	6
247	1933	18	4	5
486	1945	18	6	7

TaxonNr	# findings	# terms	# interactions	# quadratic terms
210	1952	25	11	10
272	1961	18	4	4
97	1967	15	3	4
754	1999	17	5	5
376	2020	25	11	7
703	2027	33	16	10
771	2053	19	6	5
150	2060	16	4	6
234	2077	15	4	4
39	2080	29	16	10
71	2101	25	11	9
550	2112	27	13	12
342	2125	21	7	5
407	2133	24	10	8
148	2143	15	3	2
711	2164	19	7	6
570	2170	10	2	1
146	2173	13	3	3
144	2181	22	9	8
277	2194	22	7	3
841	2204	32	17	13
448	2207	24	10	5
359	2226	26	12	8
200	2244	22	8	8
795	2250	18	7	5
892	2263	16	5	4
206	2290	11	1	3
701	2295	16	5	3
440	2297	16	4	3
768	2307	15	3	4
335	2309	13	2	4
371	2317	20	7	5
400	2347	29	13	8
317	2381	21	8	6
744	2381	20	8	7
699	2384	21	7	6
899	2421	26	12	9
791	2429	20	6	7
556	2480	17	6	5
688	2483	28	14	10
647	2492	19	7	8
614	2494	19	7	7
689	2521	18	6	6
8	2522	23	11	9
452	2530	16	5	2
16	2575	20	7	6
340	2584	20	7	6

TaxonNr	# findings	# terms	# interactions	# quadratic terms
800	2614	15	4	3
852	2628	14	2	2
320	2631	12	2	3
649	2657	30	15	7
143	2704	23	11	7
437	2707	22	9	8
770	2708	19	8	6
469	2717	11	2	2
853	2728	15	3	2
607	2731	19	7	5
219	2750	26	11	6
572	2767	23	10	6
172	2799	22	9	6
862	2869	19	6	5
216	2873	14	2	2
268	2904	23	10	9
192	2907	25	10	7
731	2913	20	8	6
70	2932	22	8	5
391	3028	15	3	6
34	3047	31	13	9
778	3097	19	5	5
121	3129	24	11	7
285	3145	36	22	15
587	3205	26	12	10
251	3250	28	13	9
240	3286	24	10	7
145	3294	28	13	8
149	3304	22	10	6
2	3313	23	7	9
198	3318	14	4	3
575	3350	21	7	5
378	3359	21	10	8
344	3440	26	11	9
354	3453	12	2	3
631	3459	22	10	6
252	3469	30	15	11
436	3480	20	8	7
167	3514	26	10	7
217	3566	27	12	8
637	3578	19	7	6
844	3592	30	14	12
130	3643	26	14	9
545	3729	30	14	10
761	3781	34	19	11
135	3789	29	15	10
788	3794	24	9	8

TaxonNr	# findings	# terms	# interactions	# quadratic terms
625	3822	18	6	6
552	3861	26	13	9
589	3872	25	11	8
306	3873	29	14	9
580	4029	20	6	7
108	4047	30	16	11
171	4162	21	9	3
873	4209	27	15	11
205	4262	20	5	6
581	4333	27	12	8
334	4344	25	11	9
533	4377	29	14	11
278	4377	32	18	10
519	4395	28	14	8
265	4431	17	5	7
353	4520	27	11	7
244	4536	24	10	8
657	4542	31	15	6
729	4578	22	8	6
404	4608	18	5	5
608	4636	38	22	18
20	4648	20	7	6
802	4665	29	16	11
431	4672	30	17	10
720	4675	20	7	7
284	4694	17	4	6
781	4700	38	23	14
139	4750	20	8	7
578	4790	33	19	10
426	4877	30	15	10
529	4940	20	9	7
120	4965	26	13	7
374	5005	21	8	5
220	5009	30	15	10
294	5138	26	13	7
830	5178	22	7	6
463	5185	23	10	4
355	5216	24	8	8
424	5254	34	18	10
107	5275	18	8	7
41	5357	15	4	5
203	5382	17	4	5
109	5418	41	26	16
636	5443	27	10	9
742	5450	20	5	4
336	5555	32	15	10
895	5564	22	9	6

TaxonNr	# findings	# terms	# interactions	# quadratic terms
652	5616	29	12	11
113	5688	27	14	10
47	5771	35	20	16
259	5874	37	21	15
362	5882	23	9	7
548	5970	27	12	8
309	5975	24	10	7
470	6012	29	16	10
260	6101	27	14	9
263	6171	37	22	12
291	6257	21	9	5
288	6503	22	9	8
639	6585	33	19	11
405	6694	32	16	10
590	6781	24	8	5
434	6788	33	19	8
828	6813	37	21	11
230	7001	29	14	7
428	7094	31	17	9
60	7332	35	20	12
99	7544	25	10	7
764	7559	33	17	9
713	7635	34	21	10
221	7664	18	5	4
233	7742	18	7	3
403	7744	28	15	10
300	7787	21	7	7
522	7818	18	6	5
633	8096	14	4	5
906	8099	25	10	9
77	8196	22	10	7
100	8244	36	19	7
423	8281	36	19	10
207	8282	29	15	7
191	8292	29	15	7
81	8347	32	16	12
439	8367	42	27	16
369	8525	23	8	5
456	8540	34	18	7
347	8552	23	10	6
662	8567	33	17	12
303	8911	22	9	6
80	9028	25	12	7
124	9088	40	23	12
324	9339	27	13	10
547	9432	34	18	12
29	9444	47	29	17

TaxonNr	# findings	# terms	# interactions	# quadratic terms
31	9648	23	10	5
325	9693	26	11	8
438	9705	34	20	10
293	9871	26	10	9
3	10431	24	8	7
911	10758	22	10	8
25	10872	16	5	4
630	10943	36	20	9
537	10982	29	15	12
520	11390	40	23	12
379	11799	31	16	8
322	12071	38	22	14
464	12295	41	26	13
117	12362	19	6	6
189	12541	42	25	13
736	12945	37	21	15
44	13565	23	9	8
532	13595	34	20	13
719	13982	28	14	10
601	14176	43	25	17
218	15125	39	23	13
170	15281	53	36	20
836	15611	38	21	14
747	15988	36	20	13
14	16610	44	26	17
632	16703	51	31	17
900	17267	41	25	16
523	17818	49	32	16
765	17903	39	23	15
248	18268	44	26	20
599	19653	37	20	14
422	19797	48	30	19
773	21154	36	20	12
859	21385	43	25	13
13	24242	37	22	17
346	28794	16	6	5
541	29773	41	25	15

Appendix 7 Review[1]

Betreft: Review MOVE rapport (titel MOdel for terrestrial VEgetation version 4.0 door M. H. C. van Adrichem, F.G. Wortelboer en G.W.W. Wamelink)
Van: Paul W. Goedhart (Biometris)
Voor: Harm Houweling (WOt Natuur & Milieu)
Datum: 18 januari 2010

Inleiding

Het doel van MOVE is het voorspellen van de kans op aanwezigheid van plantensoorten in afhankelijkheid van Ellenberg getallen N, R, F en S, het vegetatietype (VEG) en de fysisch geografische regio (FGR). Daartoe wordt voor elke individuele soort een logistisch regressiemodel aangepast op presentie/absentie gegevens voor een groot aantal vegetatie opnames. Het regressiemodel komt tot stand door stapsgewijze selectie met lineaire en kwadratische termen in N, F, R en S, de hoofdeffecten VEG en FGR, alsmede een groot aantal tweede orde interacties tussen deze termen. Het model met het laagste Bayesian Information Criterion (BIC) wordt geselecteerd. Elke plantensoort heeft zijn eigen regressiemodel.

De voorspelde kans op aanwezigheid van een soort wordt getransformeerd naar een voorspelde harde aan- of afwezigheid door kansen groter dan een grenswaarde te interpreten als hard aanwezig. Deze grenswaarde wordt gevonden door Kappa, een statistische maat voor overeenstemming tussen waargenomen en voorspelde aanwezigheid, te optimaliseren. Deze stap is blijkbaar noodzakelijk omdat in het vervolgmodel DIMO, dat dispersie van planten modelleert, gewerkt wordt met harde aanwezigheid van soorten.

Het werkdocument is als volgt opgebouwd:

1. Introduction. Hierin wordt het doel van MOVE kort beschreven, alsmede de positie van MOVE in de Natuurplanner modelketen.
2. Material and methods. Dit hoofdstuk start met een beschrijving van de verschillen tussen de huidige versie 4.0 van MOVE en de vorige versie 3.2. Dan wordt de vegetatie dataset beschreven die gebruikt is om de logistische regressiemodellen aan te passen. Er zijn aparte paragrafen voor de vegetatie opnames (§ 2.2) en voor de beschrijving van de predictoren N, R, F, S, VEG en FGR (§ 2.3). In § 2.4 wordt de stapsgewijze logistische regressie kort beschreven en in § 2.5 worden verschillende goodness-of-fit statistics beschreven. In deze paragraaf wordt ook de methodiek om de grenswaarde te bepalen gedocumenteerd. In de laatste paragraaf (§ 2.6) wordt beschreven hoe de validatie van de regressiemodellen is uitgevoerd.
3. Results. In dit hoofdstuk worden allereerst de aangepaste regressiemodellen beschreven. In § 3.2 volgt de validatie met betrekking tot de voorspelde kans op aanwezigheid en in § 3.3 de validatie voor de voorspelde harde aan- of afwezigheid.
4. Discussion and conclusions.
5. References
6. Diverse technische bijlagen met S-plus, Access en VBA scripts.

---

[1] De verwijzingen in appendix 7 zijn verwijzingen naar een concept-versie van dit werkdocument

Dit werkdocument is onderdeel voor het verkrijgen van de status A. De opdrachtgever voor deze review heeft de volgende centrale vragen:

1. Zijn de methode en resultaten helder en duidelijk beschreven en is de beschreven methode juist gehanteerd. Worden de juiste conclusies getrokken?
2. Zijn de juiste methoden gehanteerd. Zou je aanbevelingen kunnen doen voor eventuele verbeteringen m.b.t. dit aspect?

Deze vragen worden eerst globaal en daarna in meer detail beantwoordt. Tevens zijn eerdere opmerkingen van Arjen van Hinsberg en van Cajo ter Braak aan deze review toegevoegd.

De reactie van de auteurs is steeds cursief onder aan elk punt toegevoegd

Globaal commentaar

1. Een groot deel van de rapportage betreft de gebruikte statistische methoden. Hoewel het meestal duidelijk is welke methode gebruikt wordt, is de beschrijving regelmatig onnauwkeurig en soms onduidelijk, incompleet of verwarrend. De gebruikte terminologie wijkt soms af van datgene wat gebruikelijk is. Het Engels is voor verbetering vatbaar.
   De terminologie is aangepast en de beschrijvingen zijn uitgebreid daar waar nodig, grotendeels uitgevoerd door Paul Goedhart.

2. De stapsgewijze selectie (§ 2.4) houdt waarschijnlijk wel rekening met marginaliteit van hoofdeffecten en interacties (Venables & Ripley, 1997, pp 218-219), maar waarschijnlijk niet met marginaliteit van lineaire en kwadratische termen (Venables, 2008). Dat impliceert dat een model met een kwadratische term zonder de bijbehorende lineaire term geselecteerd kan worden. Dit wordt in het algemeen als ongewenst beschouwd. Dit aspect is onvoldoende gedocumenteerd in de rapportage.
   Paul Goedhart heeft hierover een stuk discussie toegevoegd en een figuur.

   In de stapsgewijze selectie worden een groot aantal mogelijke termen gebruikt; in totaal 22 hoofdeffecten en 105 interacties. Dit brengt het risico met zich mee dat termen worden geselecteerd die toevallig samenhangen met de response. Dit zal naar verwachting met name optreden voor soorten met weinig presenties. Voor een aantal soorten is wellicht vooraf bekend welke indicator waarden van invloed zijn op de response. Die voorinformatie zou gebruikt kunnen worden bij het formuleren van het startmodel. Daarmee wordt de kans verkleind dat zulke toevallige termen worden geselecteerd.
   Opgenomen in de discussie. In een volgende versie van MOVE zal hier verder over nagedacht worden

3. De voorspelde kans op aanwezigheid wordt via een grenswaarde, de critical COO, afgerond op 0 (afwezig) of 1 (aanwezig) ten behoeve van vervolgmodellen als DIMO. De grenswaarde wordt bepaald door maximalisatie van de mate van overeenstemming tussen de waarnemingen en de voorspelde 0/1 via de kappa statistic. De eigenschappen van deze methode zijn onduidelijk en de referentie (Nelson & Mead) is niet opgenomen in de lijst van referenties.
   De referentie is opgenomen. Met de rest is nu niets gedaan, er wordt later op terug gekomen.

Een alternatief is gebruik te maken van de door Cajo ter Braak voorgestelde ROC curve en het oppervlakte onder die curve; zie hiervoor de review hieronder. Ook in deze methode wordt echter het vinden van een grenswaarde uitbesteed aan een statistische grootheid, zonder dat duidelijk is wat de grenswaarde keuze betekent voor het vervolgmodel. Aangezien DIMO een stochastisch model is voor de ruimtelijke verspreiding van plantensoorten kan de stochasticiteit van het wel of niet aanwezig zijn van soorten aan DIMO worden toegevoegd. Indien dit onwenselijk is, dan behoort ergens maar niet noodzakelijkerwijs in dit rapport, de gevoeligheid van DIMO en eventuele andere vervolgmodellen voor verschillende grenswaarden gerapporteerd te worden.
Dank voor de suggestie. De net uitgevoerde onzekerheidsanalyse van SMART2-SUMO2-MOVE4 heeft als omissie dat de onzekerheid in MOVE4 zelf niet mee kon worden genomen. De volgende logische stap zou dus zijn om MOVE4 op zijn opvolger in samenhang met DIMO en de nieuwe doelbereikingsgraadmeter aan een onzekerheidsanalyse te onderwerpen. Dit is echter niet zozeer bedoeld voor dit onderzoek en rapport, er is dus hier verder niets mee gedaan.

4. De titel van § 2.6 luidt “Cross-validation of results”. Er is echter geen kruisvalidatie toegepast maar resubstitutie waarbij precies dezelfde data zijn gebruikt voor het aanpassen en het valideren van het model. Bij de eenvoudigste vorm van kruisvalidatie wordt het model aangepast op een deel van de data en vervolgens gevalideerd op het resterende deel van de data.
   Aangepast.

   Het aantal termen in de stapsgewijze selectie is groot; het betreft 22 hoofdeffecten en 115 interactie termen. Selectie procedures zijn kwetsbaar wanneer het aantal mogelijke termen van dezelfde orde grootte is als het aantal waarnemingen. Dan kunnen termen geselecteerd worden die lijken samen te hangen met de respons maar die in werkelijkheid geen voorspellend vermogen hebben. Voor de MOVE data kan dit vooral optreden wanneer het aantal presenties gering is. Daarom is kruisvalidatie met name van belang voor de meer zeldzame soorten.
   Er zijn nu meer data beschikbaar en die zouden kunnen worden gebruikt voor validatie, nu niet uitgevoerd.

5. In § 2.6 worden 4 maten voorgesteld waarmee het model gevalideerd wordt; 3 van deze maten zijn gebaseerd op een regressie van de (geclusterde) waarnemingen op de voorspellingen. Op basis van deze maten worden, via grenswaarden, scores van 0, 1 of 2 toegekend die vervolgens gesommeerd worden tot een “total validation score”. De gebruikte maten zijn problematisch en de grenswaarden en bijbehorende scores worden slecht beargumenteerd. De toegevoegde waarde van de “Total validation score” is onduidelijk. Op basis van de voorgestelde maten kunnen daarom moeilijk conclusies getrokken worden ten aanzien van de voorspelkracht van MOVE.

   Overigens is validatie inherent lastig omdat voorspelde kansen nu eenmaal slecht vergeleken kunnen worden met 0/1 waarnemingen. De in het rapport voorgestelde clustering, samen met een geschikte maat van overeenstemming, bieden wellicht goede mogelijkheden. Een goed startpunt is ook Hosmer *et al* (1997).
   We hebben besloten de testen in het hoofdrapport te laten staan. Wel wordt er op verschillende plekken ingegaan op de voors en tegens en hoe moeilijk het is om dit te

valideren, zeker ten tijde van de ontwikkeling van het model. De wetenschap is inmiddels weer gevorderd en er zijn nu testen beschikbaar.

6. In de discussie wordt opgemerkt dat MOVE 4 niet (veel) beter is dan MOVE 3.2. Volgens de Hosmer Lemeshow test is MOVE 4 zelfs slechter. Het grootste verschil tussen beide versies lijkt dat in MOVE 3.2 de factoren VEG en FGR in de modelselectie worden opgenomen, terwijl in MOVE 4 de aparte dummy vectoren worden gebruikt. Tevens worden in MOVE 3.2 geen interacties meegenomen. Dit impliceert dat de MOVE 3.2. modellen in de regel simpeler zullen zijn. Waarom toch voor de ingewikkeldere modellen van MOVE 4 wordt gekozen wordt niet beargumenteerd. Een vuistregel in de statistiek is dat simpele modellen in het algemeen beter voorspellen dan ingewikkelde modellen.
   In de discussie is nu aangegeven dat MOVE4 ook simpeler is door FGR en Veg te splitsen wat tot vereenvoudiging leidt van de interpretatie.

   Er wordt voorgesteld om na te gaan of een verdere verfijning van de vegetatie klassen en de toevoeging van een management variabele tot betere voorspellingen zou kunnen leiden. Ik stel voor om in een dergelijke studie ook vereenvoudigingen van het model en verschillende selectie methoden op te nemen. De ROC curve kan dan bijvoorbeeld gebruikt worden om de voorspelkwaliteit van de verschillende modellen te vergelijken. *Doen we nu niets mee, maar er zal in een volgende versie rekening mee worden gehouden.*

7. In een eerdere audit van MOVE 3.2 wordt een validatie aanbevolen waarbij data verzameld tot zeg 1990 gebruikt worden om het model te kalibreren en data vanaf 1990 om het model te valideren. Deze validatie ontbreekt ook nu. Deze specifieke validatie is met name interessant om te zien of historische data gebruikt kunnen worden om toekomstige data te voorspellen. Daarmee sluit deze validatie goed aan bij het gebruik van MOVE in de Natuurplanner.
   De hier beschreven versie van MOVE4 is van voor de audit. Op de audit is geen actie gevolgd. De aanbeveling blijft staan. We gaan geen validatie uitvoeren binnen dit project, het was ook niet de opdracht van het project waarbinnen MOVE4 is beschreven.

8. Indien MOVE ook gebruikt wordt om veranderingen in aanwezigheid te voorspellen als gevolg van gewijzigde abiotiek, dan zou een validatie van dit aspect ook welkom zijn. De huidige validatie richt zich alleen op de grootte van de voorspelde kans.
   In de discussie bij het stuk over validatie is een paar zinnen hierover toegevoegd, ook in samenhang met punt 7.

Gedetailleerd commentaar

1. Er wordt steeds gesproken over “chance of occurence”. Ik zou dit vervangen door “probability of occurence” of door “presence probability”.
   OK, veranderd.

2. Er wordt steeds gesproken over calculated chance of occurence en “calculated presence”. Hierin zou ik “calculated” vervangen door “predicted”.
   OK

3. Ik betwijfel of de korte beschrijving van MOVE in paragraaf § 1.1 onder Figure 1 voldoende duidelijk is voor diegenen die onbekend zijn met het onderwerp. Er wordt bijvoorbeeld niet expliciet vermeld dat het om 0/1 gegevens gaat.
   Een beetje extra info gegeven, paragraaf 1.1 is echter een korte inleiding, meer info over MOVE4 is verkrijgbaar in de rest van het rapport.

4. In paragraaf § 1.2 worden SMART en SUMO kort beschreven. Een beschrijving van LGM ontbreekt. Tevens ontbreekt hier een beschrijving van de conversie van fysieke eenheden, die blijkbaar volgen uit SMART en SUMO, naar Ellenberg getallen die invoer zijn voor MOVE. Dit P2E model ontbreekt ook in Figure 1. Het P2E model wordt voor het eerst genoemd in de discussie.
   P2E is toegevoegd aan de figuur en in de beschrijving. Een kort beschrijving van LGM is ook toegevoegd.

5. Het hoofdstuk “Material and Methods” start met een paragraaf met een beschrijving van de verschillen tussen de huidige versie MOVE 4 en de vorige versie 3.2. Deze paragraaf is tegelijkertijd een beschrijving van MOVE 4. Ik stel voor om de verschillen met MOVE 3.2 in een aparte paragraaf aan het eind van dit hoofdstuk op te nemen. De meeste stukken tekst in § 2.1 die dan nog overblijven kunnen beter ondergebracht worden in § 2.4. Bij de beschrijving van MOVE zou het helpen om een grafiek op te nemen met voorbeelden van 0/1 gegevens en hun gefitte responsiecurven in één of twee dimensies.
   Uiteindelijk besloten om dit niet te doen, veel werk en er zitten ook wel voordelen in om op deze wijze aan het begin MOVE4 neer te zetten.

6. In § 2.1, pagina 9 wordt gesproken over
   a. “All continuous variables were allowed to have optimum curves” terwijl bedoeld wordt dat “fitted curves were allowed to have an optimum”.
      OK aangepast
   b. “all variables were allowed to have interactions which the other variables”. Uit textbox 1 blijkt echter dat interacties tussen enerzijds R,N en F en anderzijds S niet zijn opgenomen. Ook interacties tussen de VEG en FGR dummies zijn niet opgenomen.
      Niet aangepast
   c. “explains (a little) more of the total deviation” er wordt bedoeld deviance
      OK aangepast
   d. “variables were added to the model when adding a significant ($p<0.05$) amount to the deviance explained only”. Er wordt bedoeld “when the deviance based test statistic is significant at the 5% level”
      OK aangepast
   e. “The multiple regression procedure chooses between the model terms presented in the full model”. Er wordt echter geen full model aangepast. Deze zin leidt tot verwarring en kan weg.
      OK aangepast
   f. “The categorical variables (different vegetation types and geophysical regions) only have a linear component”. Er wordt bedoeld dat de dummy variabelen slechts lineair zijn opgenomen; deze zin kan weg omdat het kwadraat van een dummy variabele identiek is aan de dummy variabele zelf.
      OK aangepast

7. In paragraaf § 2.2 wordt de volgende motivatie gegeven om soorten met minder dan 50 presenties te verwijderen “This avoids the time consuming calculation of many species that will not result in a significant model.” Met de huidige computerkracht lijkt dit een gelegenheidsargument. Waarom voor deze zeldzame soorten de COO lager zou zijn dan de werkelijke kans, zoals iets verder wordt gesteld, wordt niet duidelijk gemaakt, en waarom is dat geen probleem voor soorten met tussen de zeg 50 en de 100 presenties?
   OK, stuk tekst weggelaten over COO en verder aangegeven dat 50 arbitrair is en dat we weinig vertrouwen hebben in de data bij zo lage aantallen.

8. Het laatste stuk van § 2.2 beschrijft terecht dat de opnames geen aselecte steekproef vormen waardoor bijvoorbeeld zeldzame soorten oververtegenwoordigd kunnen zijn. Dat impliceert juist dat de COO voor deze soorten te hoog kan zijn en dat is in tegenspraak met punt 7. Het is zeker zo dat het opnemen van meerdere variabelen, zoals bijvoorbeeld FGR, dit probleem deels ondervangt. Dit is slechts een kwalitatieve uitspraak; de mate waarin dit probleem ondervangen wordt is onbekend. De op één na laatste zin in deze paragraaf “Displacements that arose before, through mixing of information about these environments can no longer occur.” behoeft daarom nuancering.
   Aangepast

9. Op pagina 11 wordt gemeld dat “For MOVE3 (de Heer *et al.*, 2000) the vegetation type was determined”. In aanvulling op de referentie zou een korte beschrijving van de methodiek welkom zijn.
   In de Heer et al staat niet meer dan in dit rapport, namelijk dat de indeling is gemaakt op basis van de soorten in de opnamen. Niets veranderd

10. Op pagina 11 “all relevés with missing values for any of the variables were removed from the database”. Graag aangeven per indicator variabele hoeveel missings er zijn.
    Het is nu te veel werk om dit nog toe te voegen.

11. In aanvulling op Table 3 zou ik ook graag statistische maten zien per niveau van VEG en FGR met bijbehorende discussie. Nog beter zou zijn zogenaamde kernel density estimates die in één oogopslag de verdeling van de Ellenberg variabelen weergeven. Dat kan dan ook in plaats van de (ongenummerde) figuur bovenaan pagina 13 die bovendien suggereert dat de gebruikte indicator variabelen discreet zijn.
    Terechte vraag, maar dit wordt nu niet aangepast

12. Op pagina 13 “By incorporating this variable (FGR) into some regression models the reliability of the simulation of these plant species occurrence increases tremendously in many cases”. Graag een kwantitatieve maat. Geldt dit overigens ook voor VEG?
    De tekst is hergeformuleerd, het vergt nu te veel tijd om dit uit te voeren.

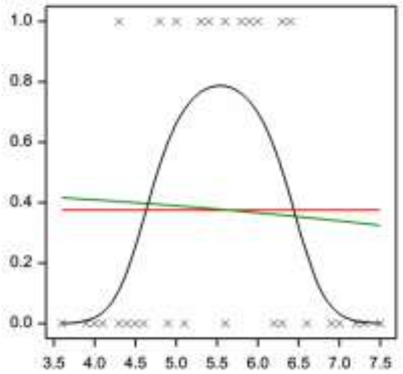
13. In aanvulling op Figure 2 en Table 4 graag een kruistabel FGR x VEG met aantallen opnames.
    Verzoek niet uitgevoerd, want kost te veel tijd

14. De inhoud van textbox 1 kan geschreven worden als (VEG[1…5] + FGR[1…9]) * (R + N + F + S + R^2 + N^2 + F^2 + S^2) + R.N + R.F + N.F waarbij VEG[1…5] bijvoorbeeld de 5

dummy variabelen zijn die de vegetatie factor representeren. Hierbij wordt de gebruikelijke notatie voor statistische modellen gebruikt.
Aangepast

15. De gebruikte S-plus routines voor stapsgewijze selectie, zie § 2.4, houden waarschijnlijk rekening met marginaliteit. Dat wil zeggen dat interacties slechts dan geselecteerd kunnen worden als beide hoofdeffecten reeds in het model zitten, en een hoofdeffect kan niet uit het model verwijderd worden indien een interactie met dit hoofdeffect nog in het model zit. Er wordt echter, weer waarschijnlijk, geen rekening gehouden met de "marginaliteit" van lineaire en kwadratische termen. Dat impliceert dat modellen met een kwadratische term zonder de bijbehorende lineaire term mogelijk zijn. Omdat de correlatie tussen de lineaire en kwadratische term veelal hoog is zal dit regelmatig voorkomen. Zo is de correlatie tussen (1,2…9) en (1,4…81) gelijk aan 0.975. Met zulke hoge correlaties hangt het van de toevallige configuratie van de data af of de lineaire dan wel de kwadratische term wordt geselecteerd. Meestal wordt een regressiemodel met alleen een kwadratische term als onwenselijk beschouwd omdat het optimum van dat model bij de waarde 0 optreedt. Ook geldt dat een model met alleen een kwadratische term gevoeliger is voor extrapolatie dan een model met een lineaire term. Vanwege het potentieel grote aantal interacties met VEG en FGR zal in het gebruik van MOVE extrapolatie regelmatig optreden.
Er wordt geen rekening gehouden met marginaliteit. Dit in de tekst gezet.

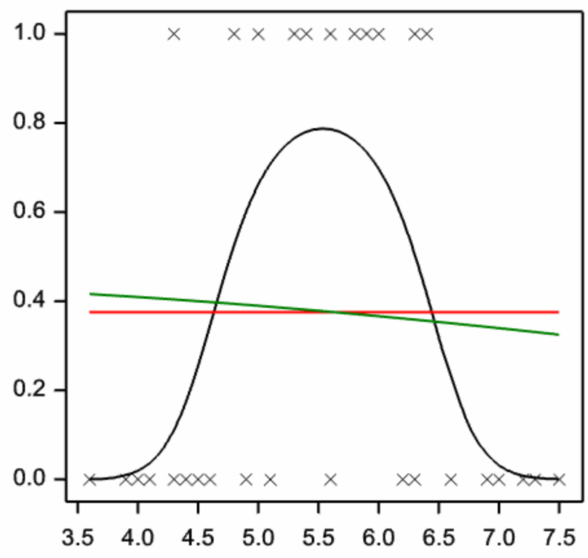
De stapsgewijze selectie wordt gestart vanuit het model met geen enkele term. Indien de responsiecurve perfect klokvormig is, dan worden zowel de lineaire als de kwadratische term niet geselecteerd. Een voorbeeld hiervan, voor een kleine dataset, wordt hiernaast gegeven. Hierin is de zwarte lijn de aangepaste curve met een lineaire en een kwadratische term, de rode lijn de gefitte lineaire curve en de groene lijn de curve met alleen een kwadratische term. Zowel de lineaire als de kwadratische term zijn niet significant, maar indien beiden in het model worden opgenomen dan zijn ze zeer significant. Vanwege de grootte van de dataset en de waarschijnlijk niet symmetrische verdeling van de Ellenberg variabelen zal deze situatie zich mogelijk weinig voordoen.



Bovenstaande stuk is door Paul verder uitgewerkt tot beneden staand stuk

For each species, a stepwise selection procedure was used to obtain the "best" model with respect to the Bayesian Information Criterion. The selection procedure starts from the null model with only a constant, and adds and drops terms according to their significance. In case the true response curve is perfectly quadratic in an environmental variable, the linear and quadratic term might not be selected. An hypothetical example of this is given in the figure. The black line is the true response curve, the red line the fitted linear model and the green line the curve with only the quadratic term. Both the linear and the quadratic term are not significant when added separately to the model, but very significant when added both. Because of the large size of the dataset, it is unlikely that this will ocuur in practice. However, because the stepwise selection procedure treats the linear and quadratic term similarly, it can occur that the quadratic term is selected before the linear term because it is more significant.

This implies that models with a quadratic term without the accompanying linear term can be selected. This is undesirable because a quadratic term alone implies that the optimum of the curve is at zero, and because a quadratic term is more sensitive to extrapolation than a linear term. Moreover the interaction between the quadratic term and e.g. *fgr* can be selected before the interaction between the linear term and *fgr*. Note that extrapolation might occur frequently because of the large number of possible interactions with *veg* and *fgr*.



Hypothetical example of a case the where the true response curve is perfectly quadratic in an environmental variable and thus the the linear and quadratic term might not be selected. The black line is the true response curve, the red line the fitted linear model and the green line the curve with only the quadratic term.

The automatic model selection procedure employs 137 possible terms of which 22 are main effects. This might pose selection problems for species which are rather rare, say with 50-300 presences. For such species variables can be selected by pure chance, not because there is an underlying relationship with the response. Such variables will depreciate the predictive power of the regression model. This problem is partly overcome by using the conservative BIC criterion. On the other hand the BIC criterion only selects a variable when the deviance decreases with minimally Log(108826) = 11.60. This is equivalent with a p value of 0.00066 when testing for the variable. This seems unnessary stringent especially with respect to the main effect.

Alternative model selection procedures could be investigated. For example the following 4-step procedure respects marginality of linear and quadratic terms and their interaction with *fgr* and *veg* terms:

1. The three possible models for each environmental variable *f*, *r*, *n* and *s* are a) no inclusion in the model, b) a linear relationship and c) a quadratic relationship. This give $3^4 = 81$ different models. These models are all fitted and the best model, according to some criterion, is retained.
2. Using a stepwise procedure the *fgr* and *veg* terms are added to the model obtained in step 1.
3. Linear interactions between selected environmental variabels in step 1 and *fgr* and *veg* terms selected in step 2 are added to the model again using a stepwise procedure.
4. Finally for the linear interactions selected in step 3) the corresponding interactions with quadratic environmental variabels are considered, but only in case the quadratic term itself is selected in step 1).

In each of the 4 steps a different selection criterion might be used, for example with increasing strictness such that interactions in step 3) and 4) are less swiftly selected. Also in step 1) prior information about the relationship with the environmental variables might be used.

The fitted models are tested in several ways. One way was by testing for the percentage deviance explained. However, the percentage deviance explained is not a real goodness of fit measure but is often used as such (Swartzman en Huang, 1989; Heikkinen, 1996; Van de Rijt *et al.*, 1996, Yee and Mitchell, 1991). When the number of zeroes increases, the percentage deviance explained decreases. Also when the presences are located on either end of the range of an environmental variable, the percentage deviance explained can be larger as compared to presences which are positioned around the mean value of the environmental variable. Therefore, during evaluation and comparison of different models by using the *%D* it should be taken into account that a part of the deviance explained occurs spontaneously and that there is a connection between the number of presences and the location of these presences. Therefore the main use of the %D is to compare different models for a single species.
Using the maximal Kappa statistic and assuming that a Kappa of between 0.4 and 0.8 indicates a moderate or substantial agreement and above 0.8 an almost perfect agreement between observed and predicted presence, 290 species models have a moderate or substantial agreement between observed and predicted presence and two species models have an agreement between observed and predicted presence that is almost perfect. However since the level of the kappa statistic depends on the number of presences, and the cut-off points are arbitrarily chosen, no definite conclusions can be drawn from this.

Dit stuk discussie is niet opgenomen in de discussie vanwege foute aannames (blijkt niet uit de resultaten) en een conclusie over variabelen-selectie die niet besproken is, die niet uit het onderzoek volgt. Relevante punten eruit zijn wel in de discussie opgenomen, maar dan gestaafd met gegevens van de modellen.De aantallen modellen met lineaire en kwadratische termen waren destijds niet bekend bij Paul.

16. Er wordt automatische modelselectie uitgevoerd met in totaal 137 mogelijke termen, waarvan 22 hoofdeffecten (waarbij kwadratische termen ook als hoofdeffecten worden geteld). Het grote aantal termen lijkt geen probleem omdat de dataset zeer groot is (N=108826, zie pp 10). Echter voor de meer zeldzame soorten, met zeg 50-200 presenties en dus een overmaat aan nullen, kunnen termen geselecteerd worden die toevallig samenhangen met de presenties. Zulke termen, zonder werkelijke relatie met de response, komen de voorspelkracht van een model niet ten goede. Dit probleem wordt gedeeltelijk ondervangen door het gebruik van de BIC als selectie criterium. Dit strenge criterium impliceert dat een term geselecteerd wordt indien de deviance na toevoeging van de term met minimaal Ln(108826) = 11.60 afneemt. Dit komt overeen met een p-waarde van 0.00066 bij toetsing op die term. Dit lijkt echter onnodig streng met name ten aanzien van de hoofdeffecten.
    Opgenomen in de discussie.
17. Bovengenoemde problemen (in 15, 16 en 17) kunnen wellicht gedeeltelijk ondervangen worden door het onderstaande stappenplan te volgen
    a. De drie mogelijke modellen voor een Ellenberg variabele zijn 1) geen opname van de variabele in het model, 2) een lineaire relatie en 3) een kwadratische relatie. Met vier Ellenberg variabelen geeft dat 3^4 = 81 mogelijke modellen. Deze modellen worden allemaal aangepast en het beste model, volgens een of ander criterium, wordt

geselecteerd. Dit model wordt verder als startmodel gebruikt en wordt niet meer gewijzigd.

b. Aan dit model worden, met een stapsgewijze procedure, VEG en FGR termen toegevoegd.
c. Dan worden de lineaire interacties tussen de in a) geselecteerde Ellenberg variabelen en de in b) geselecteerde VEG en FGR termen bekeken en weer met een stapsgewijze procedure aan het model toegevoegd.
d. Voor de in c) geselecteerde interacties worden de overeenkomstige interacties met de kwadratische Ellenberg variabele berekend, en wordt met deze interacties wederom stapsgewijze selectie uitgevoerd.

In elke stap kan een ander selectie criterium worden gebruikt, bijvoorbeeld met toenemende strengheid zodat de interacties in c) en d) minder snel geselecteerd worden dan de hoofdeffecten in a) en b). Deze procedure respecteert marginaliteit van lineaire en kwadratische termen en van interacties. Het zou interessant zijn om het gebruik van meer en minder strenge selectie criteria te vergelijken.
In een volgende versie van MOVE kan hier rekening mee worden gehouden

18. Pagina 16: het percentage deviance explained hangt mijns inziens niet af van de grootte van de dataset. Bovenaan pagina 17 moet “relevés with extreme numbers of zeros” vervangen worden door “species with”…. Ik betwijfel dat soorten met veel nullen een hoger percentage deviance explained hebben dan soorten met een meer uniforme verdeling. Als ik 0/1 waarnemingen simuleer conform de grafiek linksboven in Bijlage A, met pmax = 0.9, 0.7, 0.5, 0.3 en 0.1 en dus soorten met een toenemend aantal nullen dan vind ik juist kleiner wordende percentages 29, 17, 15, 8 en 3.
    In formule [1] op pagina 17 ontbreekt “x 100”. Onder deze formule staat de volgende beschrijving van het null model “(y = 1; all species are present in all relevés)”. Dit is onzin, het null model is het model met alleen een constante.
    Tekst aangepast

19. Het gebruik van de kappa statistic voor de bepaling van de grenswaarde (voor de conversie van kans op aanwezigheid naar harde aan- of afwezigheid) is nieuw. De eigenschappen van deze methode zijn onduidelijk. Om wat gevoel te krijgen heb ik een kwadratische responsiecurve in één dimensie genomen en daar waarnemingen bij gesimuleerd. Vervolgens is het kwadratische model op de gesimuleerde waarnemingen aangepast en is de grenswaarde met maximale kappa berekend. Dit is per responsiecurve 8 maal herhaald. De resultaten zijn opgenomen in Bijlagen A, B, C en D ; het betreft identieke responsiecurven met alleen een verschillende maximale presentiekans. De grafiek linksboven is de gebruikte responsiecurve waarbij voor de x-as de pH in gedachten kan worden genomen. De streepjes op de x-as geven aan bij welke pH waarden de aan- of afwezigheid is gesimuleerd. In de vervolggrafieken, genummerd 1-8 wordt kappa gegeven als functie van de gebruikte grenswaarde. In Bijlage A is de maximale responsie gelijk aan 0.7. In de 8 simulaties is de grenswaarde (crit COO) steeds duidelijk kleiner dan het geschatte maximale kans (max COO). Er zijn regelmatig meerdere lokale maxima, soms met nagenoeg dezelfde waarden zoals in simulaties 2, 4 en 7. In Bijlagen B en C, met maximale kans 0.5 respectievelijk 0.3 , is er één simulatie met nummer 4 resp 2 waarvoor de kritieke waarde bijna gelijk is aan de maximale kans. In Bijlage D zijn er 4 simulaties waarvoor dit geldt. Dit wordt veroorzaakt doordat er in de

simulatie relatief veel responsies gesimuleerd worden in de buurt van het optimum. Echter als de gevonden grenswaarde toegepast zou worden op een meer uniforme verdeling van waarden langs de x-as dan zouden relatief veel kansen onder de grenswaarde vallen en dus geïnterpreteerd worden als afwezig. Andersom geredeneerd, als in de calbratieset de x-waarden meer uniform verdeeld zijn dan in de toepassingsset, dan wordt een te lage grenswaarde gevonden met bijgevolg teveel presenties bij het toepassen van het model. Anders gezegd, deze methode kan goed werken als de verdeling van de x-waarden in de dataset overeenkomt met de verdeling van de x-waarden waarvoor voorspeld gaat worden. Of MOVE hieraan voldoen is onduidelijk ook al omdat in MOVE een niet aselecte dataset wordt gebruikt om het model aan te passen.
Dank voor de suggesties, niet in het rapport opgenomen

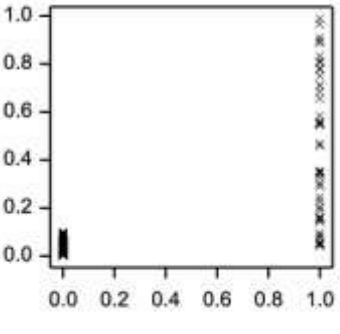
20. Pagina 17 De volgende zin begrijp ik niet “In fact the link with the original presence/absence of the different species in the dataset is maintained by the critical COO only”.
    Tekst a*angepast*

21. De indeling in Table 6 is arbitrair. Uit de simulatie in Bijlagen 1 blijkt bijvoorbeeld ook dat de maximale kappa afneemt met afnemende pmax. De gemiddelde maximale waarde van kappa over de 8 simulaties is voor pmax=0.7, 0.5, 0.3 en 0.1 respectievelijk 0.49, 0.42, 0.29 en 0.20. De referentie naar Svanholm in de title van Table 6 ontbreekt.
    Tekst aangepast

22. Over de Hosmer-Lemeshow test wordt opgemerkt (pag 18) “…tests the hypothesis that the difference between observed and predicted responses is zero for all the groups.” Dit is geen goede formulering omdat hypotheses altijd geformuleerd worden in termen van een model en nooit in termen van waarnemingen. De toets is een zogenaamde lack-of-fit toets, dat wil zeggen of het model correct is. Naast de referentie naar BIO (2000), zou een korte uitleg over de manier waarop de cut-off points worden berekend welkom zijn.
    Aangepast, geen extra uitleg gegeven.

23. De derde alinea in § 2.6, “Move calculates … per response model” is slecht geformuleerd (sum of squared deviations wordt niet geminimaliseerd) en kan gevoegelijk verwijderd worden.
    Aangepast

24. In § 2.6 worden 4 maten voorgesteld om het model te valideren.
    Validatie is inherent lastig omdat voorspelde kansen nu eenmaal slecht vergeleken kunnen worden met 0/1 waarnemingen en dit wordt ook opgemerkt in het rapport. Voor berekening van de validatie maten worden opnames geclusterd op basis van hun Ellenberg waarden. Dit geeft 504 clusters. Dan kan per cluster de gesommeerde presentie y_c vergeleken worden met de gesommeerde voorspelde kans p_c waarbij subscript c de cluster aangeeft. Ik zou graag een aantal grafieken zien waarin y_c uitgezet is tegen p_c om te kunnen beoordelen of de voorgestelde validatiematen relevant zijn. Voor een soort met 50 presenties hebben we minimaal 454 clusters met

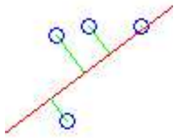


y_c=0 en kleine positieve kans p_c. De meeste van de andere clusters hebben y_c=1 en enkelen zullen een grotere waarden van y_c hebben. Een gestileerd voorbeeld van een dergelijke situatie wordt hier rechts gegeven. Ook voor soorten met grotere aantallen presenties zal voor relatief veel clusters y_c=0 gelden met kleine positieve kans p_c.
We hebben nu opgenomen dat validatie lastig is en daar discussie aan gewijd.

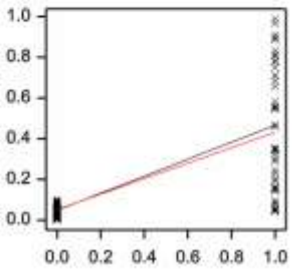
25. De formule voor de eerste validatie maat (WeightedDeviation) suggereert dat de grootheden y_c en p_c separaat gesommeerd worden over de clusters en dan worden vergeleken, in plaats van per cluster vergeleken en dan gesommeerd. Ik neem toch aan dat het laatste is gedaan, dus dat er eigenlijk een sommatieteken in de formule moet staan. Dan nog is het een merkwaardige maat omdat positieve verschillen weg kunnen vallen tegen negatieve verschillen. In het hypothetische geval dat er evenveel presenties als absenties zijn én dat elke waarneming 0 voorspeld wordt door een kans 1, en elke waarneming 1 voorspeld wordt door een kans 0 dan is de voorgestelde maat gelijk aan 0.
Aangepast (formule)

26. De tweede validatie maat is de Pearson correlatie. Deze maat verondersteld een homogene variantie en daar wordt duidelijk niet aan voldaan. Voor soorten met weinig presenties ligt een hele kluit clusters waarschijnlijk dicht bij de oorsprong (zie punt 25); ik kan niet overzien wat dat betekent voor de correlatie. Voor het gestileerde voorbeeld hierboven is de correlatie overigens gelijk aan 0.76. De gebruikte eerste drempelwaarde (0.0077) is gebaseerd op een normale verdeling met homogene variantie en daar is niet aan voldaan; deze drempelwaarde is ook zo laag dat hij nauwelijks waarde heeft. De tweede drempelwaarde (0.7) wordt niet gemotiveerd.
Aangpast.

27. De derde en vierde validatie maat zijn gebaseerd op een regressie van y_c op p_c. Hiervoor wordt de mij onbekende methode van "Model II linear regression" gebruikt. Googelen leert dat hieronder wordt verstaan "major axis regression" of "orthogonal regression" waarbij niet verticale gekwadrateerde residuen worden geminimaliseerd maar gekwadrateerde residuen loodrecht op de gefitte lijn, zoals hiernaast is weergegeven. In deze methode wordt, neem ik aan, geen rekening gehouden met het feit dat de cluster waarnemingen niet normaal verdeeld zijn. Eigenlijk zou een GLM-achtige regressie uitgevoerd moeten worden waarbij de variatie toeneemt met het niveau.



Voor het hierboven geïntroduceerde gestileerde voorbeeld met een soort met weinig presenties is hiernaast de orthogonale regressie gegeven middels de zwarte lijn en de gewone regressie middels de rode lijn; in dit geval zijn er slechts marginale verschillen. Het is duidelijk dat in dit gestileerde voorbeeld het intercept significant groter is dan 0 en de regressiecoëfficiënt significant kleiner dan 1. Ik vermoed dat dit voor veel soorten met weinig presenties zal gelden, maar wellicht ook voor soorten met meer presenties.



De figuren 17 en 19 geven aan dat inderdaad voor slechts weinig soorten het intercept significant kleiner is dan 0 of de regressiecoeffcient significant groter dan 1.
De derde maat is nu gebaseerd op het betrouwbaarheidsinterval voor het intercept van de orthogonale regressie. Indien dit interval de waarde 0 bevat (dat wil zeggen regressie

door de oorsprong) dan wordt dit als goed beschouwd. De vierde maat is gebaseerd op het betrouwbaarheidsinterval voor de regressiecoëfficiënt. Indien dit interval de waarde 1 bevat, dan wordt dit als goed beschouwd. Het hier besproken gestileerde voorbeeld maakt intuïtief duidelijk dat de waarde van deze maten onduidelijk is.
Niet verwerkt, wel opgenomen dat validatie moeilijk is en dat we daar nog een keer echt werk van moeten maken.

28. De vier maten worden tenslotte omgevormd naar scores die weer opgeteld worden in een "Total validation score". Ik zie niet in wat de toegevoegde waarde hiervan is temeer omdat de drempelwaarden die gebruikt worden om de scores te definiëren vrij arbitrair lijken en ook niet gemotiveerd worden. In de definitie van deze score op pagina 21 zijn de nummers in "(ad 1)" , "(ad 2)" , "(ad 3)" en "(ad 4)" foutief. Veel soorten met weinig presenties zullen een score hebben van 0 voor de derde en de vierde maat, en dus een maximale validatie score van 4; zie ook figuur 21. Alle oneven "total validation scores" worden veroorzaakt door een correlatie die tussen de twee drempelwaarden in ligt.
De methode kan in dit stadium niet meer worden veranderd, verder zo gelaten. Er is een discussiestuk opgenomen over validatie.

29. Welke maat is dan wel geschikt om de validatie uit te voeren? Een goed startpunt voor het beantwoorden van deze vraag is het artikel van Hosmer et al (1997) waarin een aantal goodness of fit maten worden vergeleken zoals de Hosmer Lemeshow statistic. Daarin wordt ook verwezen naar de methode van Tsiatis (1980) die gebruik maakt van een clustering op basis van de x-variabelen vergelijkbaar met wat hier wordt voorgesteld. Voor elke maat geldt dat een uitsplitsing naar soorten met weinig en met veel presenties behulpzaam kan zijn in de interpretatie van de maat. Voor een maat gebaseerd op geclusterde opnames is het nog interessant te weten in welke regionen van Ellenberg waarden de voorspelling juist goed of slecht is.
Overigens geldt dat de deviance en Pearson's statistic, dit zijn de gebruikelijke goodness-of-fit statistics in GLMs, geen chi-kwadraat verdelingen volgen voor 0/1 gegevens, zie McCullagh and Nelder (1989)
De methode kan in dit stadium niet meer worden veranderd, verder zo gelaten. Er is een discussiestuk opgenomen over validatie.

30. Met betrekking tot de vorming van de clusters wordt opgemerkt dat "Too small clusters (<10 relevés) were joined to the nearest, larger cluster." Ik kan niet overzien of dit altijd een unieke indeling in clusters geeft. Wellicht is het beter om samen te voegen met de kleinste naastliggende cluster om vergelijkbare cluster groottes te krijgen.
De methode kan in dit stadium niet meer worden veranderd, verder zo gelaten.

31. Ik werd op pagina 20 halverwege op het verkeerde spoor gezet door de zin "Besides … between the observations and the calculations". Pas na tweede herlezing realiseerde ik me dat het ook hier gaat om "clustered observations". Het gebruik van het woord "calculations" is hier ongelukkig.
tesks aangepast

32. In het Results hoofdstuk op pagina 21 wordt gesteld dat "Rare species are however underrepresented in the database". Dit lijkt in tegenspraak met het gestelde op pagina 10 "It is known that there is a relative overrepresentation of relevés on locations with high

species diversity or with rare species.” De vraag is ook hoe je weet dat er sprake is van onder-representatie.
tekst aangepast

33. Op pagina 22 worden statistics gegeven voor het aantal termen in het model. Ik zou graag ook weten of dit afhangt van het aantal presenties en ook hoeveel kwadratische termen en interacties worden geselecteerd. Er wordt op pagina 21 gesteld dat veel geselecteerde modellen relatief simpel zijn, en dat verklaard wellicht de geringe verschillen met MOVE 3.2 zoals gesteld in de discussie.
    Marjolein: grafiek in rapport en tabel in bijlage gezet

    Vegetatie types 3, 4 en 5 (allen bos) worden minder vaak geselecteerd dan types 1 en 2. Wellicht is het zinvol om deze typen samen te voegen tot één enkel type en dan weer over te gaan naar de factor VEG in plaats van de VEG dummy variabelen. Dit zou de modelselectie vereenvoudigen. Voor bijna alle soorten worden één of meerdere FGR dummies geselecteerd. Is daar iets over te zeggen in relatie met de opmerkingen in § 2.2 over het niet random zijn van de dataset?
    De methode kan in dit stadium niet meer worden veranderd, verder zo gelaten.

34. Bovenaan pagina 24 wordt wellicht bedoeld “under all circumstances”.
    Marjolein: Klopt, misschien nog beter is: “This means that a relatively high number of modelled species is never very dominant in the calibration set.”

35. Bovenaan pagina 25 wordt gesteld dat “critical COO is always lower than the maximum COO” maar dat is per definitie waar. De zin “At a low COO the variable ranges are usually wider than at a high COO.” en de daarop volgende zin begrijp ik niet.
    Aangepast

36. In Figuur 11 worden alle waarden van kappa in één grafiek gegeven. Zoals eerder betoogd hangt de interpretatie van kappa af van het aantal presenties. Een onderverdeling naar het aantal presenties is dan ook welkom. Tevens is de waarde van de kappa statistic tot stand gekomen door optimalisatie en zal daarom wellicht een overschatting zijn van de werkelijke onderliggende waarde. Ook hier kan kruisvalidatie uitkomst bieden.
    OK, verder nu niets mee doen

37. Bij de bespreking van de resultaten voor de Hosmer-Lemeshow test is het onduidelijk of de resultaten tot tevredenheid stemmen of juist niet. Ook hier graag een onderverdeling van de resultaten naar het aantal presenties

    Alleen iets over opgenomen in de discussie, verder nu te veel werk om het aan te passen

38. De paragrafen § 3.2 en § 3.3 heb ik niet uitgebreid bestudeerd omdat de gebruikte maten problematisch zijn en omdat het geen resultaten voor kruisvalidatie zijn. In deze paragrafen wordt wel een uitsplitsing gemaakt naar het aantal presenties.
    -

39. Formule [4] op pagina 28 verschilt van de formule onderaan pagina 19 omdat in de noemer de macht 0.5 ontbreekt.
    Aangepast

40. Uit figuur 22 en 23 blijkt dat validatie scores 5 en 7 weinig voorkomen. Blijkbaar is voor soorten die 4 of 6 scoren op basis van de maten 1, 3 en 4, de score voor maat 2 gelijk aan 0 of 2.
    -

41. Waarom zijn de figuren 30 en 47 niet identiek aan figuur 6?
    grafiek 6, H3 laten staan en andere twee grafieken weggehaald.

42. De hoofdconclusie is dat "In total one may conclude that the model performance, as far as tested is adequate, but not more than that." (bovenaan pagina 52). Het is onduidelijk wat er onder adeuqate moet worden verstaan.
    Zin uitgebreid met dat het model gebruikt kan worden

43. In de discussie op pagina 51
    a. Wordt gesproken over de maximum COO; dit is neem ik aan de maximum fitted COO.
    b. "Re-ordering of the categorical variabels" is foutief geformuleerd.
    c. "P-value=0.05" moet zijn "p-value>0.05"
    d. "Hosmer-Lemeshow … has a major influence on the measured calibration". Wat wordt bedoeld met de laatste twee woorden?
    e. Er wordt, zonder referentie, gesproken over problemen met de Hosmer-Lemeshow test bij small samples maar het aantal waarnemingen is steeds hetzelfde en erg groot.
    f. De laatste twee zinnen onderaan "Considering all cross-validation … additional constraint." begrijp ik niet helemaal. Wordt hier betoogd dat het afronden van de voorspelde kansen op 0/1 een slechtere maten van overeenstemmingmaten geven?
    punten zijn overgenomen en de tekst is aangepast

44. In de discussie op pagina 52
    a. "Compared to the earlier version … significantly." Dit wordt in de discussie gesteld maar er zijn geen resultaten die de lezer zelf kan interpreteren. Ik zou in deze zin overigens "significantly" weglaten omdat dit woord een vooral statistische connotatie heeft.
    Significantly weggelaten.
    b. "One of the underlying assumptions … biased in themselves." Wat wordt hier bedoeld met flawless? wat is nu de eindconclusie ten aanzien van de representativiteit van de data?
    aangepast, flawless verwijderd en uitleg toegevoegd.
    c. "This does not affect the internal uncertainty of MOVE 4". Waarom is dat zo? Is dat niet in tegenspraak met "may influence the performance … tremendously". En waarom zou het gebruik van echte N, pH en vocht een betere validatie geven?
    De verwachting is dat de onzekerheid kleiner wordt, maar zeker is dat inderdaad niet. Verder tekst aangepast door eerste deel weg te laten.
    d. Er wordt gesuggereerd om management intensity in het model op te nemen. Zijn er voldoende opnames waarvoor die (historische) informatie beschikbaar is?
    Dat gaat waarschijnlijk wel lukken en het beheer kan uitgerekend worden, behoeft wel nog nadenk werk.

45. In de discussie op pagina 53

a. “Is is advisable not to validate MOVE4 … above.” Dit begrijp ik niet. Wordt hier gesteld dat MOVE altijd gevalideerd moet worden samen met DIMO? En zo ja waarom wordt dat hier niet gedaan?
Tekst veranderd, in een aantal toepassingen van MOVE4 is dit belangrijk.

b. “The correlation between n … “. Ik zou de inclusie van beiden in het model niet zonder meer af laten hangen van de correlatie.
onduidelijk achteraf wat hier werd bedoeld, ook voor Paul Goedhart.

46. De bijlagen heb ik nauwelijks bekeken. Listing 1.3b bevat wel drie keer dezelfde functie; tevens lijkt de functienaam stepBIC in deze listing te ontbreken.
niets mee gedaan.

Commentaar van Cajo ter Braak

1. The document describes that MOVE is linked to SMART2 and SUMO2. The critical step of this link is the “P2E” model that converts acidity to Ellenberg r and nutrient status to Ellenberg n. As the P2R model is only needed because of an Ellenberg-based MOVE model, a logical place to describe and validate it is this document.
P2E wordt ergens anders getest, er is wel een stuk hierover opgenomen

2. I could not understand section 2.6 Cross-validation of the results. The word cross-validation means to me that a part of the data is set aside (e.g. 10%), the regression calculations are done on the remaining data (90%) and the regression model so obtained is applied to the data set aside, giving predictions (COO) for the set aside data. In 10 fold cross-validation, this procedure is done 10 times so that each data point is set aside once. Cross-validation is important if the data set size and number of predictors is of the same order (many predictors and few data points) and not important if the data set size is huge compared to the number of predictors. As such MOVE4 does not seem to need cross-validation as N = 108826. However, for a presence/absence response and logistic regression, as in MOVE4, the relevant data set size is the number of presences (if small) *or* the number of absences (if small). The minimum number of presences is set to 50 whereas the (maximum) number of predictors is of the same order.
Tekst is aangepast

3. The conversion of COO to presence/absence predictions appear to me a great loss of information. For a rare species with maximum COO of 0.1 it is strange to say it is always present in the situation that COO = 0.1 In my view DIMO (and BIODIV) should be able to work with COO directly.
In principe kan dit. De keuze is gemaakt om met 0-1 waarden te werken, maar is geen noodzaak, ook niet voor DIMO

4. The choice of the kappa statistic to set the critical COO is original, but requires motivation and validation.
Eens, maar valt buiten de huidige mogelijkheden om dit nu te doen

5. COO and observed presence-absence can be compared directly using ROC curves and, in one statistic, the AUC, the area under the curve (Sing *et al.* 2005, Fawcett 2006).
Zie ook commentaar hierboven. Er is echter geen ruimte om dit nu te doen

6. I do not see the value of the statistics used in section 2.6. Is it not true that for the model without predictors the weighted deviation is 0, for the reason that the report just sums deviations across clusters. Plus and minus cancel in this case.
   Tekst hierover is aangepast

7. I have some more minor comments and corrections to the text in the word document.

Commentaar van Arjen van Hinsberg

Ik heb 2 belangrijke inhoudelijke opmerkingen en daarnaast enkele meer tekstuele opmerkingen. Die laatste heb ik op papier aangegeven. Ik leg dit morgenochtend op je bureau. De 2 inhoudelijke punten zijn: (1) de cross-validatie; (2) de matige beschrijving van de toepassing van het model en daarmee de beperkte toetsing van dat gebruik.

Ad 1). Eigenlijk is er geen cross-validatie uitgevoerd, terwijl de kopjes van verschillende paragrafen dit wel lijken te suggereren. Er is geen onafhankelijke dataset gebruikt om toepassing van move te checken. De eenvoudigste oplossing is dan ook niet te spreken over cross-validatie maar meer van 'check of the regressionfunctions' or 'evaluation of the regressionfunctions'. andere optie -inhoudelijk beter en bij extra inspanning uitvoerbaar- is cross-validatie op de rest van de data (inmiddels zijn er meer dan 500.000 opnamen: terwijl er 'maar' 150.000 zijn gebruikt voor het afleiden van move). Als je spreekt over 'evaluation' is het ook goed om de evaluation-aspects ook wat kritisch te bekijken. Je gebruikt er immers ook meer omdat er geen 1 alles omvattende is. Zo is het mi logisch dat zeldzame soorten op sommige evaluation-aspects slechts scoren (bijv de aspecten op pag 34 en 35: een zeldzame soort met een smalle range zit per definitie in een een beperkte range van ellenberg-klassen. lineaire regressie van de frequentie van die soort over alle klassen werkt dan niet. Daarnaast kan je zeggen dat een 100% fit ook noot te verwachten is. Immers je bekijkt aanwezigheid van 'volwassen individuen'. Die aanwezigheid is niet alleen bepaald obv abiotische conditie, maar ook op aanwezigheid van zaden. Het is immers niet voor niets dat je DIMO inzet obv florbase!!!! Afrekenen op volledige fit, hoge maximum COO, 100% verklaarde variantie ed is dan ook veel te scherp). Door minder zwaar te spreken van cross-validatie en minder zwaar op te geven van de validatie-aspecten kan je mi wel een veel gebalanceerder verhaal neerzetten, zonder te veel actie.
De formulering is aangepast.

Ad 2). In de beschrijving (inleiding, hfdst 7, discussie) wordt eigenlijk nergens uitgelegd hoe modelresultaten gebruikt worden (uitzondering in wellicht beschrijving biodiv op pag 12). Het rapport gaat steeds in op beschrijving van voorspellingen van kans op voorkomen van individuele soorten, dit zijn echter nooit de eind-uitkomsten. Het gaat soms om gemiddelde kans op voorkomen van een groep van soorten en veelal om de gemiddelde verandering van de kans op voorkomen van een set van soorten ten opzichte vanuit een ook gemodelleerde referentiesituatie. Het gebruik voor deze eind-uitkomsten zijn eigenlijk niet getoetst. Door te werken met meerdere soorten is het idee dat fouten uit middelen. Door te werken met veranderingen tov een gemodelleerde referentie geldt datzelfde. Mi zou een iets uitgebreidere beschrijving van toepassing op zijn plaats zijn in de inleiding, maar zeker ook in de discussie. Beter zou natuurlijk zijn om toetsing/validatie op die eind uitkomsten te richten maar dat zal meer inspanning vergen. Nu zou je kunnen beschrijven dat je primair de tussen-resultaten en de regressiefuncties zelf evalueert en dat je rekeninghoudend met de beperkingen van de regressiefuncties voorstelt om uitspraken te -blijven- doen voor groepen van soorten en relatieve veranderingen!!!).

Het effect op de einduitkomsten is belangrijk, echter geen onderdeel van MOVE4. De onzekerheidsanalyse die inmiddels in concept klaar is gaat daar wel op in. De opdracht was om MOVE4 en de werking daarvan te beschrijven.

Referenties

Fawcett T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, **27**, 861-874.

Hosmer, D.W., Hosmer, T., Le Cessie, S. and Lemeshow, S. (1997). A comparison of goodness-of-fit tests for the logistic regression model. *Statistics in Medicine*, **16**, 965-980.

McCullagh, P. and Nelder, J.A. (1989). *Generalized linear models, second edition*. Chapman and Hall. New York.

Sing T., Sander O., Beerenwinkel N., and Lengauer T. (2005). ROCR: visualizing classifier performance in R. *Bioinformatics*, **21**, 3940-3941.

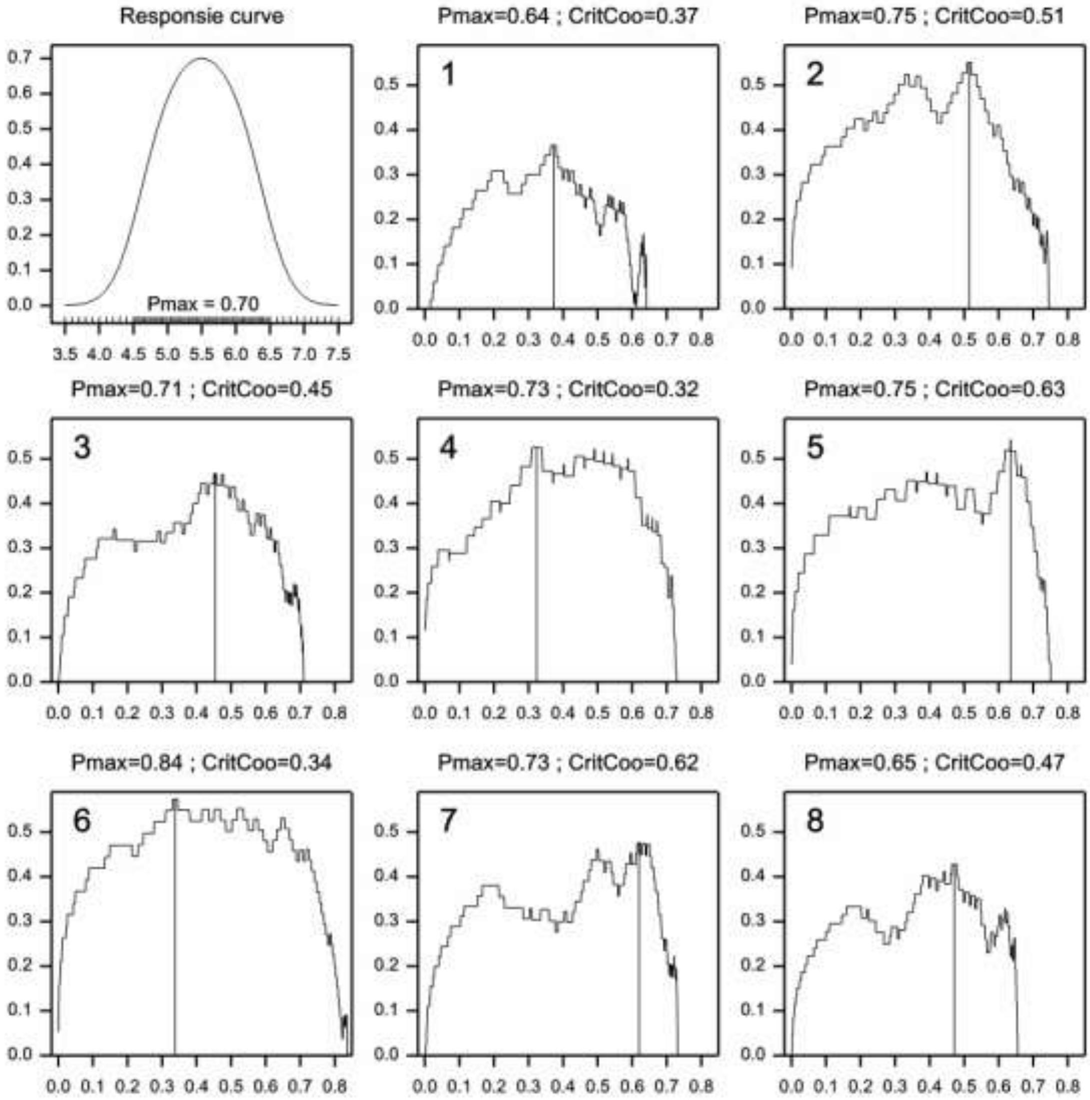
Tsiatis, A. A. (1980) A note on a goodness-of-fit test for the logistic regression model, *Biometrika*, **67**, 250-251.

Venables, W.N. (2008). StepAIC and polynomial terms. http://www.mail-archive.com/r-help@r-project.org/msg15403.html

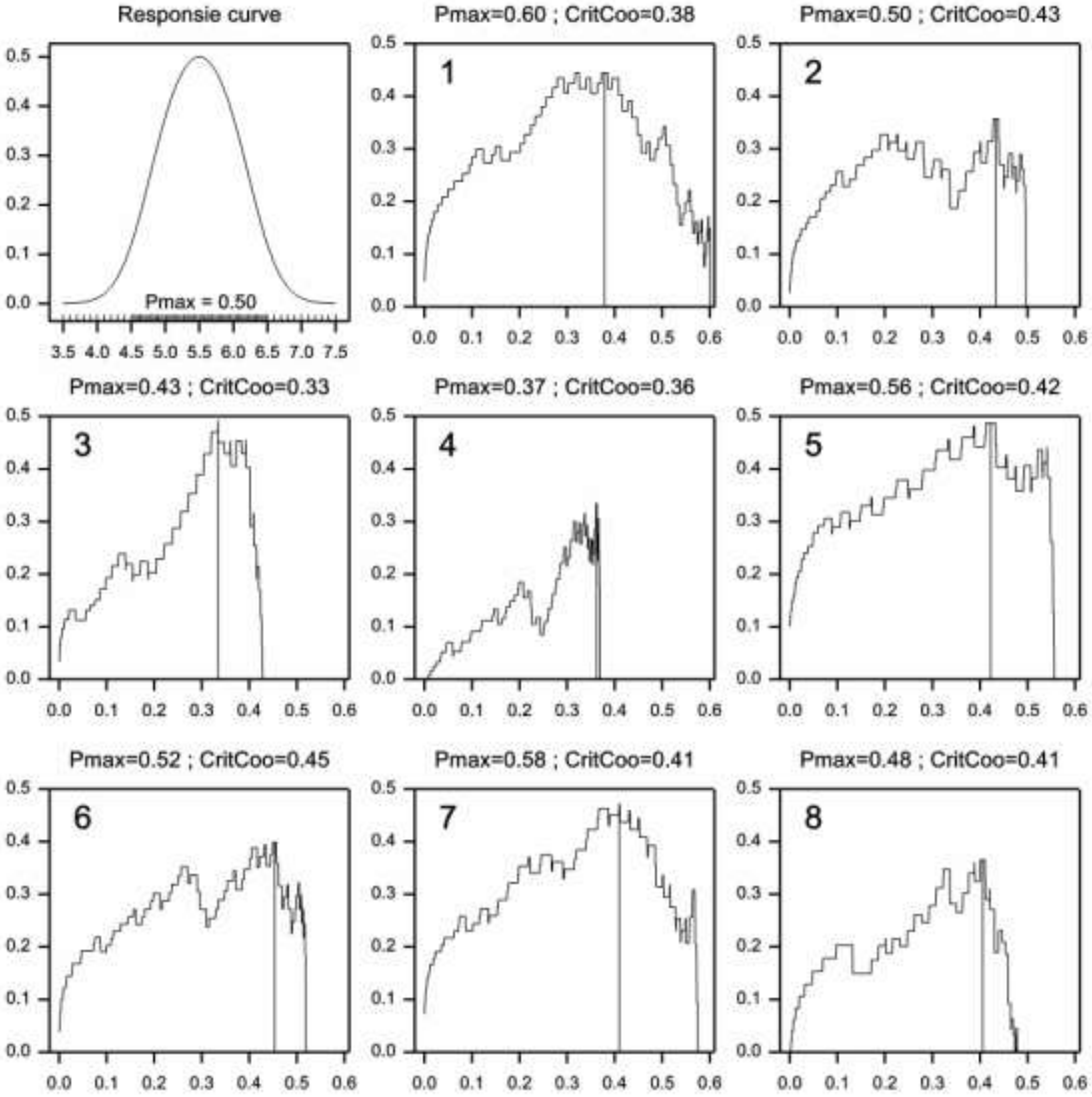
Venables, W.N. and Ripley, B.D. (1997). *Modern Applied Statistics with S-PLUS, second edition*. Springer-Verlag, New York.

Bijlage bij appendix 7

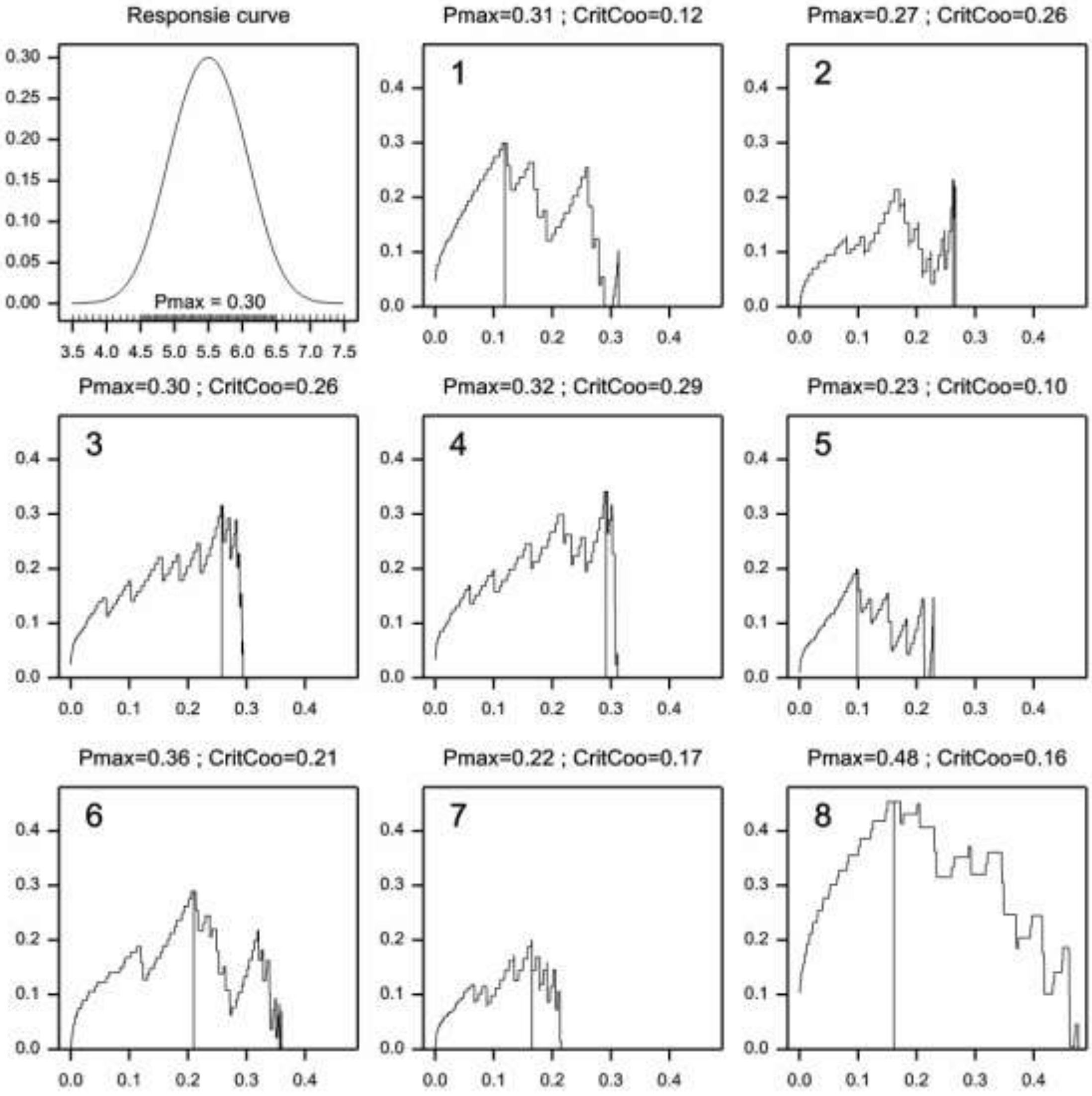
A Kappa als functie van de grenswaarde voor een responsiecurve met Pmax=0.7. De linkerboven grafiek geeft de responsiecurve die gebruikt is om 0/1 gegevens te simuleren bij die x-waarden die op de x-as zijn aangegeven. De grafieken genummerd 1-8 tonen, voor 8 gesimuleerde datasets, de waarde van kappa langs de y-as tegen de gebruikte grenswaarde.



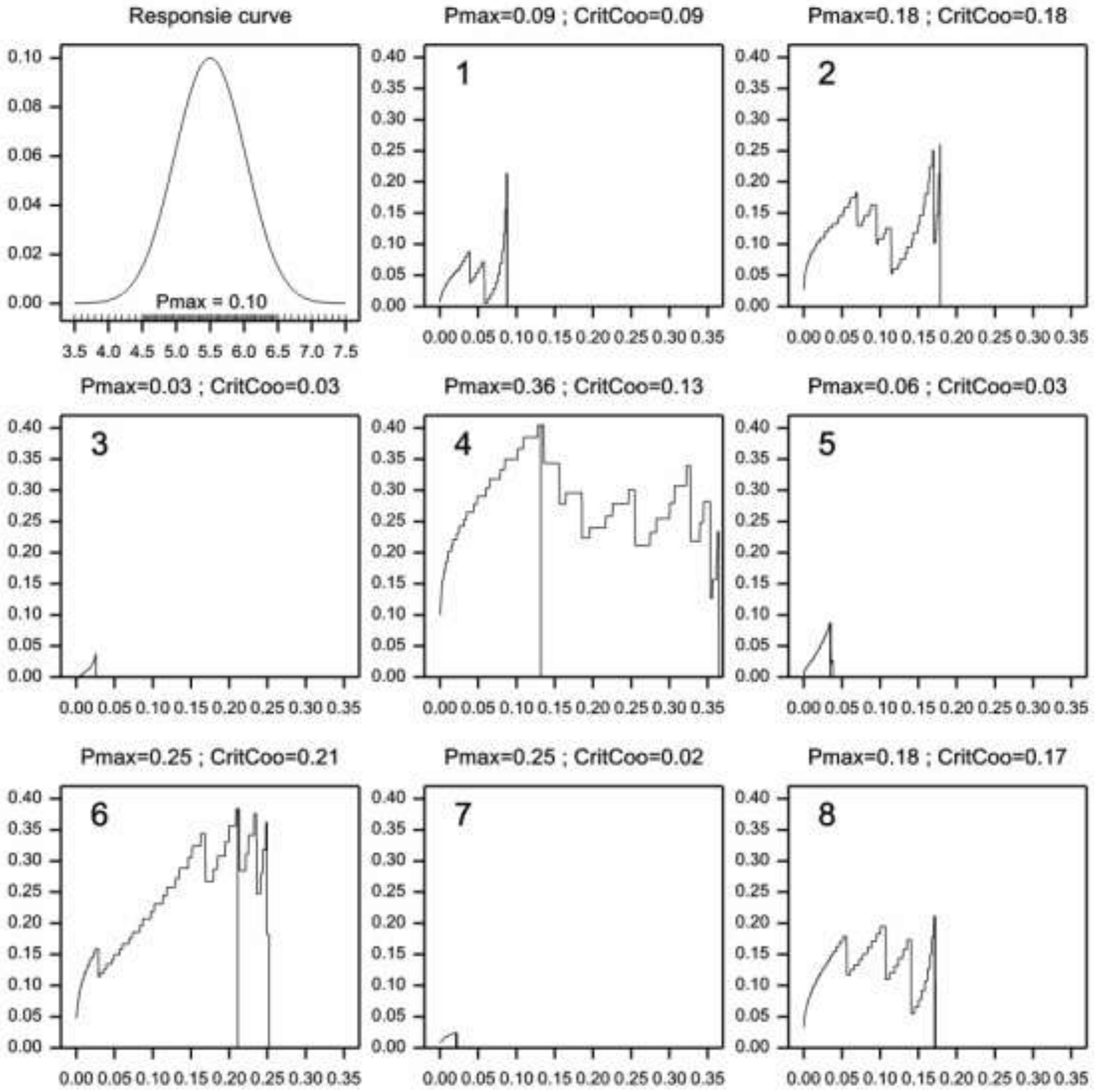
B Kappa als functie van de grenswaarde voor een responsiecurve met Pmax=0.5. De linkerboven grafiek geeft de responsiecurve die gebruikt is om 0/1 gegevens te simuleren bij die x-waarden die op de x-as zijn aangegeven. De grafieken genummerd 1-8 tonen, voor 8 gesimuleerde datasets, de waarde van kappa langs de y-as tegen de gebruikte grenswaarde.



C Kappa als functie van de grenswaarde voor een responsiecurve met Pmax=0.3. De linkerboven grafiek geeft de responsiecurve die gebruikt is om 0/1 gegevens te simuleren bij die x-waarden die op de x-as zijn aangegeven. De grafieken genummerd 1-8 tonen, voor 8 gesimuleerde datasets, de waarde van kappa langs de y-as tegen de gebruikte grenswaarde.



D Kappa als functie van de grenswaarde voor een responsiecurve met Pmax=0.1. De linkerboven grafiek geeft de responsiecurve die gebruikt is om 0/1 gegevens te simuleren bij die x-waarden die op de x-as zijn aangegeven. De grafieken genummerd 1-8 tonen, voor 8 gesimuleerde datasets, de waarde van kappa langs de y-as tegen de gebruikte grenswaarde.



Verschenen documenten in de reeks Werkdocumenten van de Wettelijke Onderzoekstaken Natuur & Milieu vanaf 2007

Werkdocumenten zijn verkrijgbaar bij het secretariaat van Unit Wettelijke Onderzoekstaken Natuur & Milieu, te Wageningen. T 0317 – 48 54 71; F 0317 – 41 90 00; E info.wnm@wur.nl
De werkdocumenten zijn ook te downloaden via de WOt-website www.wotnatuurenmilieu.wur.nl

2007

47 *Ten Berge, H.F.M., A.M. van Dam, B.H. Janssen & G.L. Velthof.* Mestbeleid en bodemvruchtbaarheid in de Duin- en Bollenstreek; Advies van de CDM-werkgroep Mestbeleid en Bodemvruchtbaarheid in de Duin- en Bollenstreek
48 *Kruit, J. & I.E. Salverda.* Spiegeltje, spiegeltje aan de muur, valt er iets te leren van een andere plannings-cultuur?
49 *Rijk, P.J., E.J. Bos & E.S. van Leeuwen.* Nieuwe activiteiten in het landelijk gebied. Een verkennende studie naar natuur en landschap als vestigingsfactor
50 *Ligthart, S.S.H.* Natuurbeleid met kwaliteit. Het Milieu- en Natuurplanbureau en natuurbeleidsevaluatie in de periode 1998-2006
51 *Kennismarkt 22 maart 2007; van onderbouwend onderzoek Wageningen UR naar producten MNP in 27 posters*
52 *Kuindersma, W., R.I. van Dam & J. Vreke.* Sturen op niveau. Perversies tussen nationaal natuurbeleid en besluitvorming op gebiedsniveau.
53.1 *Reijnen, M.J.S.M.* Indicators for the 'Convention on Biodiversity 2010'. National Capital Index version 2.0
53.3 *Windig, J.J., M.G.P. van Veller & S.J. Hiemstra.* Indicatoren voor 'Convention on Biodiversity 2010'. Biodiversiteit Nederlandse landbouwhuisdieren en gewassen
53.4 *Melman, Th.C.P. & J.P.M. Willemen.* Indicators for the 'Convention on Biodiversity 2010'. Coverage protected areas.
53.6 *Weijden, W.J. van der, R. Leewis & P. Bol.* Indicatoren voor 'Convention on Biodiversity 2010'. Indicatoren voor het invasieproces van exotische organismen in Nederland
53.7a *Nijhof, B.S.J., C.C. Vos & A.J. van Strien.* Indicators for the 'Convention on Biodiversity 2010'. Influence of climate change on biodiversity.
53.7b *Moraal, L.G.* Indicatoren voor 'Convention on Biodiversity 2010'. Effecten van klimaatverandering op insectenplagen bij bomen.
53.8 *Fey-Hofstede, F.E. & H.W.G. Meesters.* Indicators for the 'Convention on Biodiversity 2010'. Exploration of the usefulness of the Marine Trophic Index (MTI) as an indicator for sustainability of marine fisheries in the Dutch part of the North Sea.
53.9 *Reijnen, M.J.S.M.* Indicators for the 'Convention on Biodiversity 2010'. Connectivity/fragmentation of ecosystems: spatial conditions for sustainable biodiversity
53.11 *Gaaff, A. & R.W. Verburg.* Indicators for the 'Convention on Biodiversity 2010' Government expenditure on land acquisition and nature development for the National Ecological Network (EHS) and expenditure for international biodiversity projects
53.12 *Elands, B.H.M. & C.S.A. van Koppen.* Indicators for the 'Convention on Biodiversity 2010'. Public awareness and participation
54 *Broekmeyer, M.E.A. & E.P.A.G. Schouwenberg & M.E. Sanders & R. Pouwels.* Synergie Ecologische Hoofdstructuur en Natura 2000-gebieden. Wat stuurt het beheer?
55 *Bosch, F.J.P. van den.* Draagvlak voor het Natura 2000-gebiedenbeleid. Onder relevante betrokkenen op regionaal niveau
56 *Jong, J.J. & M.N. van Wijk, I.M. Bouwma.* Beheerskosten van Natura 2000-gebieden
57 *Pouwels, R. & M.J.S.M. Reijnen & M. van Adrichem & H. Kuipers.* Ruimtelijke condities voor VHR-soorten
58 Niet verschenen/ vervallen
59 *Schouwenberg, E.P.A.G.* Huidige en toekomstige stikstofbelasting op Natura 2000-gebieden
60 Niet verschenen/ vervallen

61 *Jaarrapportage 2006.* WOT-04-001 – ME-AVP
62 *Jaarrapportage 2006.* WOT-04-002 – Onderbouwend Onderzoek
63 *Jaarrapportage 2006.* WOT-04-003 – Advisering Natuur & Milieu
64 *Jaarrapportage 2006.* WOT-04-385 – Milieuplanbureaufunctie
65 *Jaarrapportage 2006.* WOT-04-394 – Natuurplanbureaufunctie
66 *Brasser E.A., M.F. van de Kerkhof, A.M.E. Groot, L. Bos-Gorter, M.H. Borgstein, H. Leneman* Verslag van de Dialogen over Duurzame Landbouw in 2006
67 *Hinssen, P.J.W.* Wettelijke Onderzoekstaken Natuur & Milieu. Werkplan 2007
68 *Nieuwenhuizen, W. & J. Roos Klein Lankhorst.* Landschap in Natuurbalans 2006; Landschap in verandering tussen 1990 en 2005; Achtergronddocument bij Natuurbalans 2006.
69 *Geelen, J. & H. Leneman.* Belangstelling, motieven en knelpunten van natuuraanleg door grondeigenaren. Uitkomsten van een marktonderzoek.
70 *Didderen, K., P.F.M. Verdonschot, M. Bleeker.* Basiskaart Natuur aquatisch. Deel 1: Beleidskaarten en prototype
71 *Boesten, J.J.T.I, A. Tiktak & R.C. van Leerdam.* Manual of PEARLNEQ v4
72 *Grashof-Bokdam, C.J., J. Frissel, H.A.M. Meeuwsen & M.J.S.M. Reijnen.* Aanpassing graadmeter natuurwaarde voor het agrarisch gebied
73 *Bosch, F.J.P. van den.* Functionele agrobiodiversiteit. Inventarisatie van nut, noodzaak en haalbaarheid van het ontwikkelen van een indicator voor het MNP
74 *Kistenkas, F.H. en M.E.A. Broekmeyer.* Natuur, landschap en de Wet algemene bepalingen omgevingsrecht
75 *Luttik, J., F.R. Veeneklaas, J. Vreke, T.A. de Boer, L.M. van den Berg & P. Luttik.* Investeren in landschapskwaliteit; De toekomstige vraag naar landschappen om in te wonen, te werken en te ontspannen
76 *Vreke, J.* Evaluatie van natuurbeleidsprocessen
77 *Apeldoorn, R.C. van,* Working with biodiversity goals in European directives. A comparison of the implementation of the Birds and Habitats Directives and the Water Framework Directive in the Netherlands, Belgium, France and Germany
78 *Hinssen, P.J.W.* Werkprogramma 2008; Unit Wettelijke Onderzoekstaken Natuur & Milieu (WOT-04). Onderdeel Planbureaufuncties Natuur en Milieu.
79 *Custers, M.H.G.* Betekenissen van Landschap in onderzoek voor het Milieu- en Natuurplanbureau; een bibliografisch overzicht
80 *Vreke,J., J.L.M. Donders, B.H.M. Elands, C.M. Goossen, F. Langers, R. de Niet & S. de Vries.* Natuur en landschap voor mensen Achtergronddocument bij Natuurbalans 2007
81 *Bakel, P.J.T. van, T. Kroon, J.G. Kroes, J. Hoogewoud, R. Pastoors, H.Th.L. Massop, D.J.J. Walvoort.* Reparatie Hydrologie voor STONE 2.1. Beschrijving reparatie-acties, analyse resultaten en beoordeling plausibiliteit.

2008

82 *Kistenkas, F.H. & W. Kuindersma..* Jurisprudentie-monitor natuur 2005-2007; Rechtsontwikkelingen Natura 2000 en Ecologische Hoofdstructuur

83 *Berg, F. van den, P.I. Adriaanse, J. A. te Roller, V.C. Vulto & J.G. Groenwold.*. SWASH Manual 2.1; User's Guide version 2

84 *Smits, M.J., M.J. Bogaardt, D. Eaton, P. Roza & T. Selnes.*. Tussen de bomen het geld zien. Programma Beheer en vergelijkbare regelingen in het buitenland (een quick-scan)

85 *Dijk, T.A. van, J.J.M. Driessen, P.A.I. Ehlert, P.H. Hotsma, M.H.M.M. Montforts, S.F. Plessius & O. Oenema.*. Protocol beoordeling stoffen Meststoffenwet; versie 1.0

86 *Goossen, C.M., H.A.M. Meeuwsen, G.J. Franke & M.C. Kuyper.* Verkenning Europese versie van de website www.daarmoetikzijn.nl.

87 *Helming, J.F.M. & R.A.M. Schrijver.* Economische effecten van inzet van landbouwsubsidies voor milieu, natuur en landschap in Nederland; Achtergrond bij het MNP-rapport 'Opties voor Europese landbouw-subsidies

88 *Hinssen, P.J.W.* Werkprogramma 2008; Unit Wettelijke Onderzoekstaken Natuur & Milieu (WOT-04). Programma 001/003/005

90 *Kramer, H.* Geografisch Informatiesysteem Bestaande Natuur; Beschrijving IBN1990t en pilot ontwikkeling BN2004

92 *Jaarrapportage 2007.* WOT-04-001 – Koepel

93 *Jaarrapportage 2007.* WOT-04-002 – Onderbouwend Onderzoek

94 *Jaarrapportage 2007.* WOT-04-003 – Advisering Natuur & Milieu

95 *Jaarrapportage 2007.* WOT-04-005 – M-AVP

96 *Jaarrapportage 2007.* WOT-04-006 – Natuurplanbureaufunctie

97 *Jaarrapportage 2007.* WOT-04-007 – Milieuplanbureaufunctie

98 *Wamelink, G.W.W.* Gevoeligheids- en onzekerheids-analyse van SUMO

99 *Hoogeveen, M.W., H.H. Luesink, L.J. Mokveld & J.H. Wisman.* Ammoniakemissies uit de landbouw in Milieubalans 2006: uitgangspunten en berekeningen

100 *Kennismarkt 3 april 2008; Van onderbouwend onderzoek Wageningen UR naar producten MNP*

101 *Mansfeld, M.J.M. van & J.A. Klijn.* "Balansen op de weegschaal". Terugblik op acht jaar Natuurbalansen (1996-2005)

102 *Sollart, K.M. & J. Vreke.* Het faciliteren van natuur- en milieueducatie in het basisonderwijs; NME-ondersteuning in de provincies

103 *Berg, F. van den, A. Tiktak, J.G. Groenwold, D.W.G. van Kraalingen, A.M.A. van der Linden & J.J.T.I. Boesten*, Documentation update for GeoPEARL 3.3.3

104 *Wijk, M.N., van (redactie).* Aansturing en kosten van het natuurbeheer. Ecologische effectiviteit regelingen natuurbeheer

105 *Selnes, T. & P. van der Wielen.* Tot elkaar veroordeeld? Het belang van gebiedsprocessen voor de natuur

106 *Annual reports for 2007; Programme WOT-04*

107 *Pouwels, R. J.G.M. van der Greft, M.H.C. van Adrichem, H. Kuiper, R. Jochem & M.J.S.M. Reijnen.* LARCH Status A

108 *Wamelink, G.W.W.* Technical Documentation for SUMO2 v. 3.2.1,

109 *Wamelink, G.W.W., J.P. Mol-Dijkstra & G.J. Reinds.* Herprogrammeren van SUMO2. Verbetering in het kader van de modelkwaliteitsslag

110 *Salm, C. van der, T. Hoogland & D.J.J. Walvoort.* Verkenning van de mogelijkheden voor de ontwikkeling van een metamodel voor de uitspoeling van stikstof uit landbouwgronden

111 *Dobben H.F. van & R.M.A. Wegman.* Relatie tussen bodem, atmosfeer en vegetatie in het Landelijk Meetnet Flora (LMF)

112 *Smits, M.J.W. & M.J. Bogaardt.* Kennis over de effecten van EU-beleid op natuur en landschap

113 *Maas,G.J. & H. van Reuler.* Boomkwekerij en aardkunde in Nederland,

114 *Lindeboom, H.J., R. Witbaard, O.G. Bos & H.W.G. Meesters.* Gebiedsbescherming Noordzee, habitattypen, instandhoudingdoelen en beheermaatregelen

115 *Leneman, H., J. Vader, L.H.G. Slangen, K.H.M. Bommel, N.B.P. Polman, M.W.M. van der Elst & C. Mijnders.* Groene diensten in Nationale Landschappen- Potenties bij een veranderende landbouw,

116 *Groeneveld, R.A. & D.P. Rudrum.* Habitat Allocation to Maximize BiOdiversity, A technical description of the HAMBO model

117 *Kruit, J., M. Brinkhuijzen & H. van Blerck.* Ontwikkelen met kwaliteit. Indicatoren voor culturele vernieuwing en architectonische vormgeving

118 *Roos-Klein Lankhorst, J.* Beheers- en Ontwikkelingsplan 2007: Kennismodel Effecten Landschap Kwaliteit; Monitoring Schaal; BelevingsGIS

119 *Henkens, R.J.H.G.* Kwalitatieve analyse van knelpunten tussen Natura 2000-gebieden en waterrecreatie

120 *Verburg, R.W., I.M. Jorritsma & G.H.P. Dirkx.* Quick scan naar de processen bij het opstellen van beheerplannen van Natura 2000-gebieden. Een eerste verkenning bij provincies, Rijkswaterstaat en Dienst Landelijk Gebied

121 *Daamen, W.P.* Kaart van de oudste bossen in Nederland; Kansen op hot spots voor biodiversiteit

122 *Lange de, H.J., G.H.P. Arts & W.C.E.P. Verberk.* Verkenning CBD 2010-indicatoren zoetwater. Inventarisatie en uitwerking relevante indicatoren voor Nederland

123 *Vreke, J., N.Y. van der Wulp, J.L.M. Donders, C.M. Goossen, T.A. de Boer & R. Henkens.* Recreatief gebruik van water. Achtergronddocument Natuurbalans 2008

124 *Oenema, O. & J.W.H. van der Kolk.* Moet het eenvoudiger? Een essay over de complexiteit van het milieubeleid

125 *Oenema, O. & A. Tiktak.* Niets is zonder grond; Een essay over de manier waarop samenlevingen met hun grond omgaan

2009

126 *Kamphorst, D.A.* Keuzes in het internationale biodiversiteitsbeleid; Verkenning van de beleidstheorie achter de internationale aspecten van het Beleidsprogramma Biodiversiteit (2008-2011)

127 *Dirkx, G.H.P. & F.J.P. van den Bosch.* Quick scan gebruik Catalogus groenblauwe diensten

128 *Loeb, R. & P.F.M. Verdonschot.* Complexiteit van nutriëntenlimitaties in oppervlaktewateren

129 *Kruit, J. & P.M. Veer.* Herfotografie van landschappen; Landschapsfoto's van de 'Collectie de Boer' als uitgangspunt voor het in beeld brengen van ontwikkelingen in het landschap in de periode 1976-2008

130 *Oenema, O., A. Smit & J.W.H. van der Kolk.* Indicatoren Landelijk Gebied; werkwijze en eerste resultaten

131 *Agricola, H.J.A.J. van Strien, J.A. Boone, M.A. Dolman, C.M. Goossen, S. de Vries, N.Y. van der Wulp, L.M.G. Groenemeijer, W.F. Lukey & R.J. van Til.* Achtergrond-document Nulmeting Effectindicatoren Monitor Agenda Vitaal Platteland

132 *Jaarrapportage 2008.* WOT-04-001 – Koepel

133 *Jaarrapportage 2008.* WOT-04-002 – Onderbouwend Onderzoek

134 *Jaarrapportage 2008.* WOT-04-003 – Advisering Natuur & Milieu

135 *Jaarrapportage 2008.* WOT-04-005 – M-AVP

136 *Jaarrapportage 2008.* WOT-04-006 – Natuurplanbureaufunctie

137 *Jaarrapportage 2008.* WOT-04-007 – Milieuplanbureaufunctie

138 *Jong de, J.J., J. van Os & R.A. Smidt.* Inventarisatie en beheerskosten van landschapselementen

139 *Dirkx, G.H.P., R.W. Verburg & P. van der Wielen.* Tegenkrachten Natuur. Korte verkenning van de weerstand tegen aankopen van landbouwgrond voor natuur

140 *Annual reports for 2008; Programme WOT-04*

141 *Vullings, L.A.E., C. Blok, G. Vonk, M. van Heusden, A. Huisman, J.M. van Linge, S. Keijzer, J. Oldengarm & J.D. Bulens.* Omgaan met digitale nationale beleidskaarten

142 *Vreke, J.,A.L. Gerritsen, R.P. Kranendonk, M. Pleijte, P.H. Kersten & F.J.P. van den Bosch.* Maatlat Government – Governance

143 *Gerritsen, A.L., R.P. Kranendonk, J. Vreke, F.J.P. van den Bosch & M. Pleijte.* Verdrogingsbestrijding in het tijdperk van het Investeringsbudget Landelijk Gebied. Een verslag van casusonderzoek in de provincies Drenthe, Noord-Brabant en Noord-Holland.

144 *Luesink, H.H., P.W. Blokland, M.W. Hoogeveen & J.H. Wisman.* Ammoniakemissie uit de landbouw in 2006 en 2007

145 *Bakker de, H.C.M. & C.S.A. van Koppen.* Draagvlakonderzoek in de steigers. Een voorstudie naar indicatoren om maatschappelijk draagvlak voor natuur en landschap te meten

146 *Goossen, C.M.,* Monitoring recreatiegedrag van Nederlanders in landelijke gebieden. Jaar 2006/2007

147 *Hoefs, R.M.A., J. van Os & T.J.A. Gies.* Kavelruil en Landschap. Een korte verkenning naar ruimtelijke effecten van kavelruil.

148 *Klok, T.L., R. Hille Ris Lambers, P. de Vries, J.E. Tamis & J.W.M. Wijsman.* Quick scan model instruments for marine biodiversity policy.

149 *Spruijt, J., P. Spoorenberg & R. Schreuder.* Milieueffectiviteit en kosten van maatregelen gewasbescherming.

150 *Ehlert, P.A.I. (rapporteur).* Advies Bemonstering bodem voor differentiatie van fosfaatgebruiksnormen.

151 *Wulp van der, N.Y.* Storende elementen in het landschap: welke, waar en voor wie? Bijlage bij WOt-paper 1 – Krassen op het landschap

152 *Oltmer, K., K.H.M. van Bommel, J. Clement, J.J. de Jong, D.P. Rudrum & E.P.A.G. Schouwenberg.* Kosten voor habitattypen in Natura 2000-gebieden. Toepassing van de methode Kosteneffectiviteit natuurbeleid.

153 *Adrichem van, M.H.C., F.G. Wortelboer & G.W.W. Wamelink **(2010).*** MOVE. Model for terrestrial Vegetation. Version 4.0

154 *Wamelink, G.W.W., R.M. Winkler & F.G. Wortelboer.* User documentation MOVE4 v 1.0

155 *Gies de, T.J.A., L.J.J. Jeurissen, I. Staritsky & A. Bleeker.* Leefomgevingsindicatoren Landelijk gebied. Inventarisatie naar stand van zaken over geurhinder, lichthinder en fijn stof.

156 *Tamminga, S., A.W. Jongbloed, P. Bikker, L. Sebek, C. van Bruggen & O. Oenema.* Actualisatie excretiecijfers landbouwhuisdieren voor forfaits regeling Meststoffenwet

157 *Van der Salm, C., L. .M. Boumans, G.B.M. Heuvelink & T.C. van Leeuwen.* Protocol voor validatie van het nutriëntenemissiemodel STONE op meetgegevens uit het Landelijk Meetnet effecten Mestbeleid

158 *Bouwma, I.M.* Quickscan Natura 2000 en Programma Beheer. Een vergelijking van Programma Beheer met de soorten en habitats van Natura 2000

159 *Gerritsen, A.L., D.A. Kamphorst, T.A. Selnes, M. van Veen, F.J.P.van den Bosch, L. van den Broek, M.E.A. Broekmeyer, J.L.M. Donders, R.J. Fontein, S. van Tol, G.W.W. Wamelink & P. van der Wielen.* Dilemma's en barrières in de praktijk van het natuur- en landschapsbeleid; Achtergronddocument bij Natuurbalans 2009.

160 *Fontein R.J, T.A. de Boer, B. Breman, C.M. Goossen, R.J.H.G. Henkens, J. Luttik & S. de Vries.* Relatie recreatie en natuur; Achtergronddocument bij Natuurbalans 2009

161 *Deneer, J.W. & R. Kruijne. **(2010).*** Atmosferische depositie van gewasbeschermingsmiddelen. Een verkenning van de literatuur verschenen na 2003.

162 *Verburg, R.W., M.E. Sanders, G.H.P. Dirkx, B. de Knegt & J.W. Kuhlman.* Natuur, landschap en landelijk gebied. Achtergronddocument bij Natuurbalans 2009.

163 *Doorn van, A.M. & M.P.C.P. Paulissen.* Natuurgericht milieubeleid voor Natura 2000-gebieden in Europees perspectief: een verkenning.

164 *Smidt, R.A., J. van Os & I. Staritsky.* Samenstellen van landelijke kaarten met landschapselementen, grondeigendom en beheer. Technisch achtergronddocument bij de opgeleverde bestanden.

165 *Pouwels, R., R.P.B. Foppen, M.F. Wallis de Vries, R. Jochem, M.J.S.M. Reijnen & A. van Kleunen,* Verkenning LARCH: omgaan met kwaliteit binnen ecologische netwerken.

166 *Born van den, G.J., H.H. Luesink, H.A.C. Verkerk, H.J. Mulder, J.N. Bosma, M.J.C. de Bode & O. Oenema,* Protocol voor monitoring landelijke mestmarkt onder een stelsel van gebruiksnormen, versie 2009.

167 *Dijk, T.A. van, J.J.M. Driessen, P.A.I. Ehlert, P.H. Hotsma, M.H.M.M. Montforts, S.F. Plessius & O. Oenema.* Protocol beoordeling stoffen Meststoffenwet- Versie 2.1

168 *Smits, M.J., M.J. Bogaardt, D. Eaton, A. Karbauskas & P. Roza.* De vermaatschappelijking van het Gemeenschappelijk Landbouwbeleid. Een inventarisatie van visies in Brussel en diverse EU-lidstaten.

169 *Vreke, J. & I.E. Salverda.* Kwaliteit leefomgeving en stedelijk groen.

170 *Hengsdijk, H. & J.W.A. Langeveld.* Yield trends and yield gap analysis of major crops in the World.

171 *Horst, M.M.S. ter & J.G. Groenwold.* Tool to determine the coefficient of variation of DegT50 values of plant protection products in water-sediment systems for different values of the sorption coefficient

172 *Boons-Prins, E., P. Leffelaar, L. Bouman & E. Stehfest (2010)* Grassland simulation with the LPJmL model

173 *Smit, A., O. Oenema & J.W.H. van der Kolk.* Indicatoren Kwaliteit Landelijk Gebied

2010

174 *Boer de, S., M.J. Bogaardt, P.H. Kersten, F.H. Kistenkas, M.G.G. Neven & M. van der Zouwen.* Zoektocht naar nationale beleidsruimte in de EU-richtlijnen voor het milieu- en natuurbeleid. Een vergelijking van de implementatie van de Vogel- en Habitatrichtlijn, de Kaderrichtlijn Water en de Nitraatrichtlijn in Nederland, Engeland en Noordrijn-Westfalen

175 *Jaarrapportage 2009.* WOT-04-001 – Koepel

176 *Jaarrapportage 2009.* WOT-04-002 – Onderbouwend Onderzoek

177 *Jaarrapportage 2009.* WOT-04-003 – Advisering Natuur & Milieu

178 *Jaarrapportage 2009.* WOT-04-005 – M-AVP

179 *Jaarrapportage 2009.* WOT-04-006 – Natuurplanbureaufunctie

180 *Jaarrapportage 2009.* WOT-04-007 – Milieuplanbureaufunctie

181 *Annual reports for 2009; Programme WOT-04*

182 Oenema, *O., P. Bikker, J. van Harn, E.A.A. Smolders, L.B. Sebek, M. van den Berg, E. Stehfest & H. Westhoek.* Quickscan opbrengsten en efficiëntie in de gangbare en biologische akkerbouw, melkveehouderij, varkenshouderij en pluimveehouderij. Deelstudie van project 'Duurzame Eiwitvoorziening'.

183 *Smits, M.J.W., N.B.P. Polman & J. Westerink.* Uitbreidingsmogelijkheden voor groene en blauwe diensten in Nederland; Ervaringen uit het buitenland

184 *Dirkx, G.H.P. (red.).* Quick responsefunctie 2009. Verslag van de werkzaamheden.

185 *Kuhlman, J.W., J. Luijt, J. van Dijk, A.D. Schouten & M.J. Voskuilen.* Grondprijskaarten 1998-2008

186 *Slangen, L.H.G., R.A. Jongeneel, N.B.P. Polman, E. Lianouridis, H. Leneman & M.P.W. Sonneveld.* Rol en betekenis van commissies voor gebiedsgericht beleid.

187 *Temme, A.J.A.M. & P.H. Verburg.* Modelling of intensive and extensive farming in CLUE

188 *Vreke, J.* Financieringsconstructies voor landschap

189 *Slangen, L.H.G.* Economische concepten voor beleidsanalyse van milieu, natuur en landschap

190 *Knotters, M., G.B.M. Heuvelink, T. Hoogland & D.J.J. Walvoort.*

A disposition of interpolation techniques

191 *Hoogeveen, M.W., P.W. Blokland, H. van Kernebeek, H.H. Luesink & J.H. Wisman.* Ammoniakemissie uit de landbouw in 1990 en 2005-2008

192 *Beekman, V., A. Pronk & A. de Smet.* De consumptie van dierlijke producten. Ontwikkeling, determinanten, actoren en interventies.

193 *Polman, N.B.P., L.H.G. Slangen, A.T. de Blaeij, J. Vader & J. van Dijk.* Baten van de EHS; De locatie van recreatiebedrijven

194 *Veeneklaas, F.R. & J. Vader.* Demografie in de Natuurverkenning 2011; Bijlage bij WOt-paper 3

195 *Wascher, D.M., M. van Eupen, C.A. Mücher & I.R. Geijzendorffer,* Biodiversity of European Agricultural landscapes. Enhancing a High Nature Value Farmland Indicator

196 *Apeldoorn van, R.C., I.M. Bouwma, A.M. van Doorn, H.S.D. Naeff, R.M.A. Hoefs, B.S. Elbersen & B.J.R. van Rooij.* Natuurgebieden in Europa: bescherming en financiering

197 *Brus, D.J.,, R. Vasat, G. B. M. Heuvelink, M. Knotters, F. de Vries & D. J. J. Walvoort.* Towards a Soil Information System with quantified accuracy; A prototype for mapping continuous soil properties

198 *Groot, A.M.E.& A.L. Gerritsen, m.m.v. M.H. Borgstein, E.J. Bos & P. van der Wielen.* Verantwoording van de methodiek Achtergronddocument bij 'Kwalitatieve monitor Systeeminnovaties verduurzaming landbouw'

199 *Bos, E.J. & M.H. Borgstein.* Monitoring Gesloten voer-mest kringlopen. Achtergronddocument bij 'Kwalitatieve monitor Systeeminnovaties verduurzaming landbouw'

200 *Kennismarkt 27 april 2010;* Van onderbouwend onderzoek Wageningen UR naar producten Planbureau voor de Leefomgeving.

201 *Wielen van der, P.* Monitoring Integrale duurzame stallen. Achtergronddocument bij 'Kwalitatieve monitor Systeeminnovaties verduurzaming landbouw'

202 *Groot, A.M.E.& A.L. Gerritsen.* Monitoring Functionele agrobiodiversiteit. Achtergrond-document bij 'Kwalitatieve monitor Systeeminnovaties verduurzaming landbouw'

203 *Jongeneel, R.A. & L. Ge.* Farmers' behavior and the provision of public goods: Towards an analytical framework.

204 *Vries, S. de, M.H.G. Custers & J. Boers.* Storende elementen in beeld; de impact van menselijke artefacten op de landschapsbeleving nader onderzocht.

205 *Vader, J. J.L.M. Donders & H.W.B. Bredenoord.* Zicht op natuur- en landschapsorganisaties; Achtergronddocument bij Natuurverkenning 2011.

206 *Jongeneel, R.A., L.H.G. Slangen & N.B.P. Polman.* Groene en Blauwe Diensten; Een raamwerk voor de analyse van doelen, maatregelen en instrumenten

207 *Letourneau, A.P, P.H. Verburg & E. Stehfest.* Global change of land use systems; IMAGE: a new land allocation module

208 *Heer, M. de.* Het Park van de Toekomst. Achtergronddocument bij Natuurverkenning 2011

209 *Knotters, M., J. Lahr, A.M. van Oosten-Siedlecka & P.F.M. Verdonschot, 2010.* Aggregation of ecological indicators for mapping aquatic nature quality. Overview of existing methods and case studies.

CONCEPTEN 2010

210 *Verdonschot, P.F.M. & A.M. van Oosten-Siedlecka* Graadmeters Aquatische natuur. Analyse gegevenskwaliteit Limnodata

211 *Linderhof, V.G.M. & Hans Leneman, 2010.* Quickscan kosteneffectiviteitsanalyse aquatische natuur

212 *Leneman, H. V.G.M. Linderhof & R. Michels, 2010.* Mogelijkheden voor het inbrengen van informatie uit de 'KRW database' in de 'KE database'

213 *Schrijver, R.A.M., A. Corporaal, W.A. Ozinga & D.P. Rudrum* Naar een nieuwe methode voor het bepalen van effecten van maatregelen voor de verhoging van de biodiversiteit in landbouwgebieden. Een test in twee gebieden in Twente en Zeeuws-Vlaanderen

214 *Hoogland, T., R.H. Kemmers, D.G. Cirkel & J. Hunink.* Standplaatsfactoren afgeleid van hydrologische model uitkomsten; Methode ontwikkeling en toetsing in het Drentse-Aa gebied.

215 *Agricola, H.J., R.M.A. Hoefs, A.M. van Doorn, R.A. Smidt & J. van Os.* Landschappelijke effecten van ontwikkelingen in de landbouw

216 *Kramer, H., J. Oldengarm en L.F.S. Roupioz.* Nederland is groener dan kaarten laten zien. De mogelijkheden om beter groen te inventariseren en monitoren met de automatische classificatie van digitale luchtfoto's

217 *Raffe, J.K. van, J.J. de Jong & G.W.W. Wamelink.* Scenario's voor de kosten van natuurbeheer en stikstofdepositie; Kostenmodule v 1.0 voor de Natuurplanner

218 *Hazeu, G.W., Kramer, H. & J. Clement.* Basiskaart Natuur 1990rev; Vervaardiging en monitoring van veranderingen

219 *Boer, T.A. de.* Waardering en recreatief gebruik van Nationale Landschappen door haar bewoners

220 *Leneman, H., A.D. Schouten & R.W. Verburg.* Kosten van varianten van natuurbeleid; Voorbereiding voor de Natuurverkenningen

221 *Knegt, B. de, J. Clement, P.W. Goedhart, H. Sierdsema, Chr. van Swaay & P. Wiersma.* Natuurkwaliteit van het agrarisch gebied

222 *Kamphorst, D.A. en Mark van Oorschot*. Kansen en barrières voor verduurzaming van houtketens

xx *Bikker, P., M.M. van Krimpen& G.J. Remmelink.* Stikstof-verteerbaarheid in voeders voor landbouwhuisdieren; Berekeningen ten behoeve van de TAN-excretie

xx *Klijn, J.A.* Wisselend getij. Omgang met en beleid voor natuur en landschap in verleden en heden; een essayistische beschouwing