Ecological Modelling 306 (2015) 305–317

Contents lists available at ScienceDirect

Ecological Modelling

journal homepage: www.elsevier.com/locate/ecolmodel

Learning ensembles of population dynamics models and their application to modelling aquatic ecosystems

Nikola Simidjievski[a,b,*], Ljupčo Todorovski[c], Sašo Džeroski[a,b]

[a] Department of Knowledge Technologies, Jožef Stefan Institute, Ljubljana, Slovenia
[b] Jožef Stefan International Postgraduate School, Ljubljana, Slovenia
[c] Faculty of Administration, University of Ljubljana, Slovenia

ARTICLE INFO

Article history:
Available online 5 September 2014

Keywords:
Aquatic ecosystems
Population dynamics
Phytoplankton growth
Process-based modelling
Ensembles
Bagging

ABSTRACT

Ensemble methods are machine learning methods that construct a set of models and combine their outputs into a single prediction. The models within an ensemble can have different structure and parameters and make diverse predictions. Ensembles achieve high predictive performance, benefiting from the diversity of the individual models and outperforming them.

In this paper, we develop a novel method for learning ensembles of process-based models. We build upon existing approaches to learning process-based models of dynamic systems from observational data, which integrates the theoretical and empirical paradigms for modelling dynamic systems. In addition to observed data, process-based modelling takes into account domain-specific modelling knowledge.

We apply the newly developed method and evaluate its utility on a set of problems of modelling population dynamics in aquatic ecosystems. Data on three lake ecosystems are used, together with a library of process-based knowledge on modelling population dynamics. Based on the evaluation results, we identify the optimal settings of the method for learning ensembles of process-based models, i.e., the optimal number of ensemble constituents (25) as well as the optimal way to select (using a separate validation set) and combine them (using simple average). Furthermore, the evaluation results show that ensemble models have significantly better predictive performance than single models. Finally, the ensembles of process-based models accurately simulate the current and predict the future behaviour of the three aquatic ecosystems.

1. Introduction

Mathematical models are widely used to describe the structure and predict the behaviour of dynamic systems under various conditions. Constructing such a model is a process that uses both expert knowledge and measured data about the observed system. The main challenge is integrating these two into an understandable model within the laws of nature.

Two major paradigms for constructing models of dynamic systems exist: theoretical (knowledge-driven) and empirical (data-driven) modelling. Following the first paradigm, domain experts establish an appropriate structure of the model and calibrate its parameters in an automatic fashion using measured data. The second approach uses measured data to search for such a combination of model structure and parameter values that leads to simulated behaviour that fits the measurements well. In both approaches, the models are often formulated as ordinary differential equations (ODEs).

Within the area of computational scientific discovery (Langley et al., 1987), a sub-field of equation discovery has emerged that studies methods for learning the model structure and parameter values of dynamic systems from observations (Džeroski and Todorovski, 2003; Bridewell et al., 2008). The state-of-the-art approaches in this area, referred to as process-based modelling (Bridewell et al., 2008; Čerepnalkoski et al., 2012), integrate the theoretical and the empirical paradigm to modelling dynamic systems. A process-based model (PBM) provides an abstraction of the observed system at two levels: qualitative and quantitative.

At the qualitative level, a process-based model comprises entities and processes. Entities correspond to agents involved in the modelled system, whereas processes represent the relations and interactions between the entities. This results in an interpretable model of a system, explaining the structure of the observed system. On the other hand, at the quantitative level, the entities define a set

* Corresponding author at: Department of Knowledge Technologies, Jožef Stefan Institute, Ljubljana, Slovenia. Tel.: +386 1 477 3635.
E-mail addresses: nikola.simidjievski@ijs.si (N. Simidjievski), ljupco.todorovski@fu.uni-lj.si (L. Todorovski), saso.dzeroski@ijs.si (S. Džeroski).

http://dx.doi.org/10.1016/j.ecolmodel.2014.08.019
0304-3800/

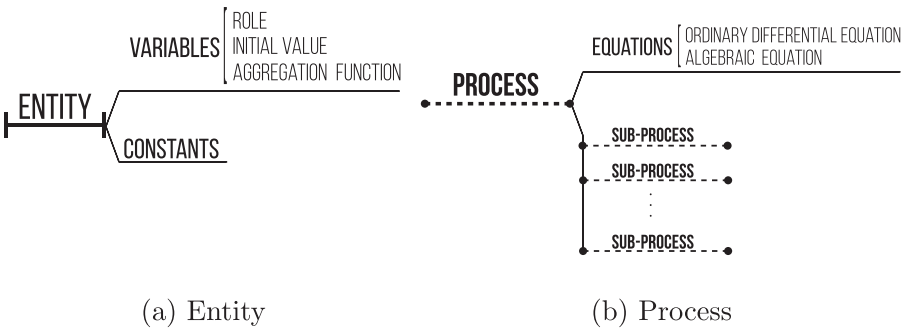


Fig. 1. The internal structure of entities and processes in process-based models.

of variables and constants, and the processes are annotated with equations modelling the underlying relations and interactions. At this level, we can transform a process-based model to a system of ODEs and simulate its behaviour.

Following the process-based modelling approach, we can generalize specific entities and processes into template entities and processes in a given modelling domain. A collection of such template entities and processes is called a library of process-based domain-specific knowledge. In modelling aquatic ecosystems, such a library of model components has been proposed by Atanasova et al. (2006b). The library defines a set of template entities, i.e., nutrients, primary producers, animals and environment, that typically occur in aquatic ecosystems (Luenberger, 1979). These entity templates are used to define template processes that provide recipes for modelling food-web interactions between the aquatic ecosystem entities. The knowledge encoded within the template entities and processes allow for automated modelling of population dynamics in aquatic ecosystems from measurements of system states (e.g., nutrients and species concentrations) through time. Process-based modelling software can then integrate the encoded knowledge with the measured system behaviour into a PBM of the observed system.

In our previous work, we have shown the utility of the process-based modelling approach for modelling population dynamics in a number of natural lakes (Čerepnalkoski et al., 2012) and marine ecosystems (Bridewell et al., 2008). Note however, that these studies focused on establishing descriptive, explanatory models of the population dynamics in aquatic ecosystems and the obtained models were analyzed and simulated on the same data that were used for learning them. In particular, they aimed to identify the limiting factors of the phytoplankton growth in the observed systems that are evident from the qualitative level of the learned process-based models. The generalization power of the obtained process-based models in terms of their ability to predict the future behaviour of the observed systems was not investigated in these studies.

In this study, we shift our focus towards the predictive performance of process-based models. The results of the preliminary experiments indicate the tendency of process-based models to overfit: While focusing on the provision of detailed and accurate descriptions of the observed systems, PBMs fail to accurately predict future system behaviour. To address this limitation of process-based models, we propose here a standard method for improving the predictive performance of models in machine learning, the use of ensembles. The idea of ensembles is to learn a set of predictive models (instead of a single one) and then combine their predictions. The prediction obtained with the ensemble is expected to be more accurate than the one obtained with a single model (Maclin and Opitz, 1999; Rokach, 2010).

The main contribution of this paper is a novel method for learning ensembles of process-based models. For tasks such as modelling the behaviour of ecosystems, the ensembles are usually employed in the context of learning tasks for classification and regression (Crisci et al., 2012; Knudby et al., 2010). However, to the best of our knowledge ensembles of process-based models have not yet been addressed in this context, and considered for tasks for modelling ecosystems.

We test the utility of the newly developed method for predictive modelling of population dynamics in lakes. To this end, we conjecture that ensembles of PBMs, similarly to other types of ensembles in machine learning, will improve the predictive performance of single models and lead to satisfactory prediction of future behaviour of the observed aquatic ecosystems. To test this hypothesis, we experiment on a series of tasks of modelling population dynamics in three lakes: Lake Bled, Lake Kasumigaura and Lake Zurich. From each lake we use seven yearly data sets, using six for learning and one for testing the predictive performance of the learned models. The aim of the experiments is two fold: Beside validating our central hypothesis (that ensembles perform better than single models), we also seek appropriate design choices related to our method for building ensembles of process-based models.

The remainder of this paper is organized as follows. Section 2 introduces the novel approach to learning ensembles of process-based models by discussing the task of automated modelling of dynamic systems – the process-based modelling approach, and focuses on a recent contribution to the area of automated process modelling, i.e., the ProBMoT tool. Section 3 describes ensemble methods in general and their adaptation for process-based modelling in particular. The design of the experiments, the evaluation measures and the data sets used are described in Section 4. Section 5 presents the results obtained the experimental evaluation. In Section 6, we discuss the contributions of this paper and overview the related work. Finally, Section 7 summarizes the work presented in this paper and discusses directions for further work.

2. Process-based modelling and ProBMoT

Equation discovery is the area of machine learning that aims at developing methods for learning quantitative laws, expressed in the form of equations, from collections of observed data. Recently, equation discovery methods have been used in the context of learning models of dynamic systems (Todorovski and Džeroski, 2007; Džeroski and Todorovski, 1993). The state-of-the-art equation discovery methods for modelling dynamic systems, referred to as process-based modelling (Bridewell et al., 2008; Džeroski and Todorovski, 2003) integrate domain-specific modelling knowledge and data into explanatory models of the observed systems. In the rest of this section, we briefly introduce the process-based modelling approach and then describe its particular implementation within the ProBMoT software platform.

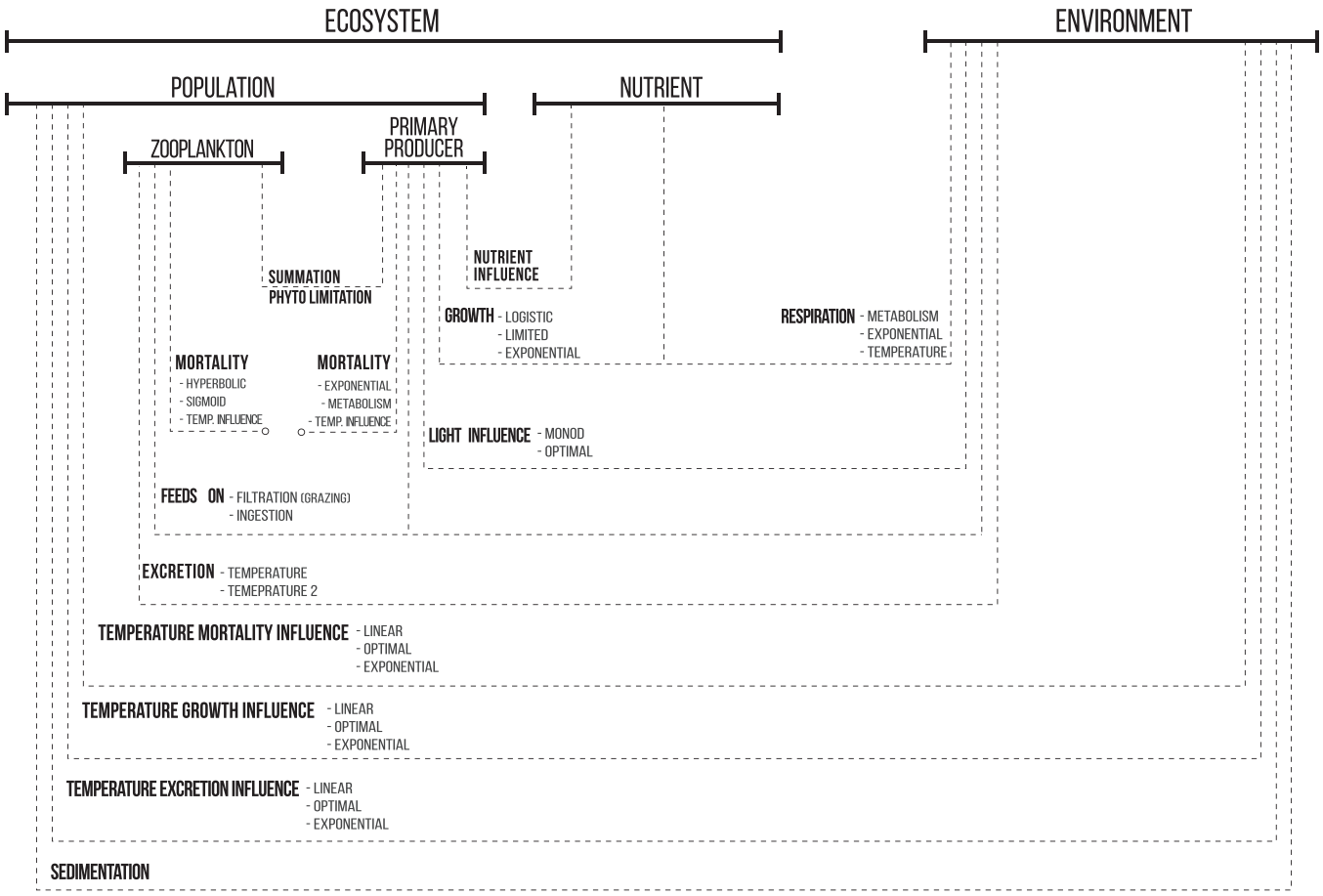


Fig. 2. The library of modelling knowledge comprising template entities (thick horizontal lines) and processes (dashed lines connecting the entities) for modelling population dynamics in aquatic ecosystems.

2.1. Process-based modelling

Process-based models provide a description of the observed system at two levels of abstraction. At the upper, a qualitative level, process-based model consists of entities and processes. The entities represent the main components of the observed system, whereas the processes correspond to the interactions between the system components. At the qualitative level, process-based models provide insight into the high-level conceptual structure of the system. However, this high-level description does not provide enough details that would allow for simulation of the system behaviour.

On the other hand, at the quantitative level, entities and processes provide further modelling details that allow for the transformation of PBMs to ODEs and therefore simulation of the system. Fig. 1 depicts the internal structure of entities and processes, which defines a number of properties as follows.

Entities comprise variables and constants related to the components of the observed system. For example, an entity representing phytoplankton in an aquatic ecosystem would include a variable corresponding to its concentration, that changes through time, and a constant, corresponding to its maximal growth rate. Each entity variable has three important properties: the role in the model, the initial value and the aggregation function. The role of the variable in the model can be endogenous, i.e., representing internal system state, or exogenous, i.e., representing an input external to the system (not modelled within the system). An example of an endogenous variable in an aquatic ecosystem is the concentration of phytoplankton, while the water temperature is often treated as exogenous. Initial values of endogenous variables are necessary for model simulation. Moreover, each endogenous variable has its constraints defined, which limit the set of feasible values of the variable (for example, the concentration of the phytoplankton can neither be negative nor exceed 100 gWM/m^3). Finally the aggregation function for a variable specifies how influences from multiple processes on the specific variable are need to be combined, e.g., additively or multiplicatively.

The processes include specifications of the entities that interact, equations, and sub-processes. Consider the process of phytoplankton growth. It involves the phytoplankton as well as the growth limiting factors of nutrients and the environment. Equations provide the model of the interaction represented by the process and contains variables and constants from the entities involved in the corresponding interaction. In the phytoplankton growth example, an equation would define the mathematical model for calculating the growth rate. Finally, each process can include a number of sub-processes related to different aspects of the interaction. For example, the mathematical term of temperature limitation of growth (or nitrogen/nutrient), can be specified in an appropriate temperature (or nitrogen) limitation sub-process of the growth process. Sub-processes improve both the interpretability and the modularity of process-based models (Bridewell et al., 2008).

The entities and processes represent specific components and interactions observed in the particular system at hand. The process-based modelling approach allows for a higher-level representation of domain-specific modelling knowledge, employing the concepts of entity and process templates. They both provide general modelling recipes that can be instantiated to any specific components or interactions in the system. The phytoplankton entity from the example above is an instance of the general template entity of primary producer. Similarly, particular model of phytoplankton

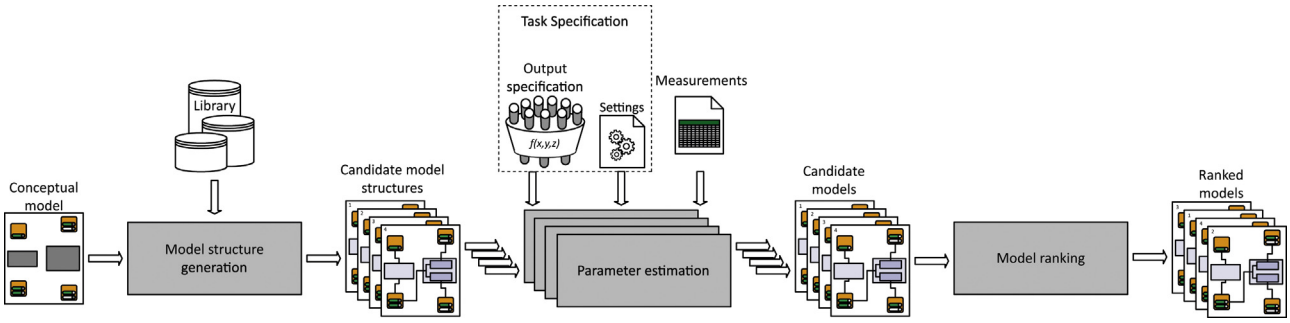


Fig. 3. The architecture of the ProBMoT platform for process-based modelling.

growth used in the above example process, is an instance of the more general growth process template. The template entities and processes are collected together into a library of components for modelling systems in a given domain of use.

Fig. 2 represents a high-level overview of the library for modelling population dynamics in aquatic ecosystems proposed by Atanasova et al. (2006b). The library organizes the templates in hierarchies. The hierarchy of entity templates (represented by thick horizontal lines in Fig. 2) in aquatic ecosystems includes the ecosystem entity and the environment entity templates at the highest level. The ecosystem entity template instantiates further on into population and nutrient entity templates, the first being further specialized into zooplankton and primary producer entity templates. Similarly, the process templates (represented by dashed lines, connecting involved entity templates) are organized into a hierarchy that defines the space of modelling alternatives. For example, the growth of a primary producer can be logistic, exponential or limited.

When learning process-based models, the entity and process templates from the library are instantiated to specific entities and processes corresponding to the observed system. These specific entities and processes represent model components that can be in turn used to define the set of candidate model structures. The algorithm for learning models employs knowledge-based methods from artificial intelligence to enumerate all candidate model structures. To evaluate a structure, the learning algorithm performs parameter estimations fitting the values of the constant model parameters that minimize the model error, i.e., the discrepancy between the model simulation and the observed system behaviour. The parameter estimation employs non-linear optimization to minimize the model error. Finally, the obtained models are sorted by decreasing model error and the best-ranked model is considered to be the result of the learning process.

Basic automated modelling algorithms perform exhaustive search through a constrained space of candidate process-based models, limiting the number of processes in the model (Bridewell et al., 2008). Advanced learning algorithms, such as Lagramge2.0 (Todorovski and Džeroski, 2007) and HIPM (Hierarchical Inductive Process Modelling) (Todorovski et al., 2005), perform heuristic search and allow for more sophisticated hierarchical constraints on the plausible process combinations. In the remainder of this section we will briefly describe the most recent implementation of the PBM approach, called ProBMoT, which stands for Process-Based Modelling Tool (Čerepnalkoski et al., 2012).

2.2. ProBMoT

Fig. 3 represents the architecture of the ProBMoT software platform for process-based modelling. ProBMoT supports the simulation, parameter estimation and automated learning of process-based models. ProBMoT follows the PBM approach described above.

The first input to ProBMoT is the conceptual model of the observed system. The conceptual model specifies the expected logical structure of the expected model in terms of entities and processes that we observe in the system at hand. ProBMoT combines the conceptual model with the library of modelling choices to obtain a list of candidate model structures. For each model structure, the parameter values are estimated to fit the observed behaviour of the modelled system.

The parameter estimation process is based on the metaheuristic optimization framework jMetal 4.4 (Durillo and Nebro, 2011) that implements a number of global optimization algorithms. In particular, ProBMoT uses the Differential Evolution (DE) (Storn and Price, 1997) optimization algorithm. For simulation purposes, each process-based model is first transformed to a system of ODEs. In turn, ProBMoT employs the CVODE (C-package for Variable-Coefficient ODE) solver from the SUNDIALS suite (Cohen and Hindmarsh, 1996).

ProBMoT implements a number of measures of model performance: the sum of square errors (SSE) between the simulated and observed behaviour, and several variants thereof. The latter include mean squared error (MSE), root mean squared error (RMSE), relative root mean squared error (ReRMSE) and weighted root mean squared error (WRMSE). The last two are used in the experiments presented here, and will be explained in greater detail later, together with the particular ProBMoT parameter settings.

3. Ensemble methods and ensembles of process-based models

Learning ensembles is an established method for improving the predictive performance of models in machine learning (Okun et al., 2011), however learning ensembles of process-based models has not been considered so far. In this section, we define ensembles of process-based models and corresponding methods for learning them. First, however, we provide a brief overview of classical ensemble methods in machine learning.

An ensemble is a set of models (referred to as base-models or ensemble constituents) that is expected to lead to predictive performance gain over a single model. The idea behind ensembles is to improve the overall predictive power by combing the predictions of individual base-models. An ensemble method consists of three main components: a technique for learning/generating a set of candidate base-models, a technique for selecting the base-models that constitute the ensemble, and a combining scheme specifying how the base-model predictions are aggregated into an ensemble prediction.

Based on how the candidate base-models are learned, the ensembles can be homogeneous or heterogeneous. In homogeneous ensembles, the base-models are learned with the same

learning algorithm, but from different samples of the training data. Commonly used sampling variants include: sampling of data instances (bagging Breiman, 1996a, boosting Freud and Schapire, 1999), sampling of data features/attributes (random subspaces Ho, 1998) or both (random forests Breiman, 2001). On the other hand, in heterogeneous ensembles, the candidate base-models are learned using different learning algorithms (e.g., stacking Wolpert, 1992).

After we have generated the candidate base-models, we have to select the ones to be included in the ensemble. Most classical ensemble methods would typically use all the candidate models as ensemble constituents. In contrast, ensemble pruning techniques can be used to learn small-size ensembles (thus reducing the computational complexity) and improve ensemble robustness, e.g., in the case of bagging (Zhou et al., 2002).

Finally, the combining scheme depends on the type of the base-models. In the case of classification models that predict qualitative values, different voting scheme are employed. In the case of regression models that predict numeric values, the alternatives include average, weighted average and weighted median (Drucker, 1997).

In this paper, we adapt the well-known bagging method for learning homogeneous ensembles where the training data is modified by sampling the data instances. The *bagging* method, introduced by Breiman (1996a), is one of the earliest and simplest ensemble learning methods. It first randomly samples data instances, with replacement, to obtain several bootstrap replicates of the training data. Next, a candidate base model is learned from each of the different bootstrap replicates. An important property of bagging is that it can be implemented as a parallel algorithm, which is due to the fact that it handles each bootstrap sample independently.

In the continuation of this section, we introduce a novel approach to learning ensembles of process-based models. This approach follows the bagging idea introduced above. We are going to introduce it following the three-components structure of ensemble methods as outlined above.

3.1. Learning individual process-based models

Using ProBMoT, we learn the individual candidate base-models from different samples of the observed behaviour at hand. The notable difference from bagging in the context of regression is that, in our case, the data instances have temporal ordering that has to be preserved in each data sample. To achieve this, we implement the sampling by introducing weights for each instance. The weight corresponds to the number of times the instance has been selected in the process of sampling with replacement. Instances that have not been selected (the ones with weight 0) are simply omitted from the sample. From each sample, a PBM is learned with ProBMoT.

To account for the instance weights when learning a model from the sample, we employ the weighted root mean squared error in ProBMoT:

$$WRMSE(m) = \sqrt{\frac{\sum_{t=0}^{n} \omega_t * (y_t - \hat{y_t})^2}{\sum_{t=0}^{n} \omega_t}}. \quad (1)$$

Here y_t and $\hat{y_t}$ correspond to the *measured* and *simulated* values (simulating the base model m) of the system variable y at time point t. n denotes the total number of instances in the data sample and ω_t denotes the weight of the data instance at time point t.

3.2. Selecting and combining process-based models into an ensemble

When learning a PBM from each data sample, ProBMoT selects the top-ranked model as a result. However, we can use two alternative data sets to calculate the error used to rank the models in ProBMoT. By default, ProBMoT ranks the models using the error on the training data sample; we refer to this selection method as *regular*. In contrast, the *validation* selection method employs a separate validation data set to calculate the error used for model ranking in ProBMoT.

In order to simulate an ensemble, we need to simulate every candidate base model. The resulting ensemble simulation is a combination of the predictions of all individual base-models in the respective time point. In our case, we use average, weighted average and weighted median as combining schemes: These are commonly used for tasks such as regression (Drucker, 1997). In the case of average, all base-models participate in the resulting simulation equivalently. For the weighted average and weighted median schemes a confidence is calculated for each of the base-models with respect to their performance error. The base-models with higher confidence will dominate in the resulting ensemble simulation.

However, some of these simulations may not be valid, i.e., may not satisfy the constraints given in the library of background knowledge. In this case, we perform ensemble pruning, i.e, we discard these base-models from the resulting ensemble. Bellow, we illustrate the necessity of using ensemble pruning by comparing two ensembles with 100 base-models.

Table 1 presents the results of the comparison of two ensembles, complete and pruned, and the number of candidate base-models discarded, in terms of performance error on the test data sample (see Section Section 4.3) for 15 different cases. From the table, we can see that in all but one experimental data set (K4), the pruned ensemble outperforms (or is equal in performance to) the complete ensemble. Note also that, in several cases (B1, B3, K1, K3, Z5) the performance of the ensemble is significantly improved. By discarding the base-models with unstable simulations from the ensemble, we can ensure valid ensemble prediction and a stable simulation. In this paper, we use ensemble pruning of this kind as a standard technique when selecting the ensemble constituents and simulating the ensemble prediction.

Table 1
Predictive performance (ReRMSE on the testing data) errors of the complete and pruned ensembles and the number of base-models pruned from the 100 model ensembles learned on 15 data sets, described in see Section 4.3.

Case	Complete	Pruned	# base-models pruned
B1	1.8E+03	1.055	53
B2	1.243	1.243	0
B3	17.037	1.046	11
B4	0.737	0.737	1
B5	0.625	0.625	0
K1	9.2E+01	0.927	19
K2	4.482	1.840	98
K3	4.0E+04	0.907	17
K4	0.823	0.988	9
K5	0.978	0.978	0
Z1	1.105	1.028	2
Z2	1.187	1.077	3
Z3	8.579	1.212	6
Z4	0.972	0.972	0
Z5	2.8E+01	1.390	24

4. Experimental setup

In this section, we present the setup of the experiments we performed to empirically evaluate the performance of the method for learning ensembles of process-based models. We perform the evaluation on tasks of modelling population dynamics in three aquatic ecosystems: Lake Bled in Slovenia, Lake Kasumigaura in Japan, and Lake Zurich in Switzerland. The goal of our empirical evaluation is twofold.

First, we are looking for a set of optimal design decisions related to the algorithm for learning ensembles. In particular, we want

Table 2
Overview of the data used for modelling population dynamics in the three lakes: Lake Bled, Lake Kasumigaura and Lake Zurich.

	Bled	Kasumigaura	Zurich
Environmental influence	Temperature Light	Temperature Light	Temperature Light
Nutrients	Phosphorus Nitrogen Silica	Phosphorus Nitrogen Ammonia	Phosphorus Nitrogen Silica
Primary producer	Phytoplankton	Phytoplankton	Phytoplankton
Zooplankton	*D. hyalina*	None	*D. hyalina*
Training data (labels)	1996–2000 (B1–B5)	1986–1990 (K1–K5)	1996–2000 (Z1–Z5)
Validation data	2001	1991	2001
Test data	2002	1992	2002

to perform a comparative analysis of using different methods for learning and the base-models to be included in the ensemble, different methods for combining the simulations of the base-models in the ensemble, and different numbers of based models in the ensemble. Based on the results of this comparative analysis, we make a set of choices that we use for learning ensembles of process-based models of aquatic ecosystems.

Second, we aim at analysing the predictive performance of ensembles of process-based models. In particular, we test the central hypothesis of this paper that ensembles of process-based models improve the predictive power of a single process-based model for a given aquatic ecosystem. Furthermore, we want to investigate whether the performance improvement is related to the diversity of the predictions of the ensemble constituents. Finally, in the last series of experiments, we visually compare the predictions of ensembles to those of single models in each of the three aquatic ecosystems.

4.1. Library of domain-specific knowledge and task of modelling aquatic ecosystems

In our experiments, we use the library of domain-specific knowledge for process-based modelling of aquatic ecosystems presented by Čerepnalkoski et al. (2012). Note that the library is based on the previous work presented by Atanasova et al. (2006b). The library, presented in Fig. 2, formalizes modelling knowledge in terms of a set of template entities and processes for modelling population dynamics in an arbitrary lake ecosystem. To reduce the computational complexity of the experiments performed in this paper, we used a simplified version of the library, where we omitted some of the alternatives for modelling individual processes. The simplified version of the library and the conceptual model, lead to 320 candidate model structure for Lake Kasumigarua and 128 candidates for the other two aquatic ecosystems used.

ProBMoT employs the Differential Evolution (DE, Storn and Price, 1997) method for parameter estimation with the following settings: population size of 50, *rand/1/bin* strategy, and the differential weight (F) and the crossover probability (Cr) both set to 0.6. The limit on the number of evaluations of the objective function is one thousand per parameter. For simulating the ODEs we used the CVODE simulator with absolute and relative tolerances set to 10^{-3}. The particular choice of parameters setting of DE is based on previous studies of the sensitivity of DE for estimating parameters of ODE models, which includes also modelling of the population dynamics in Lake Bled (Taškova et al., 2011, 2012).

4.2. Data

The data used in this study originates from three aquatic ecosystems: Lake Bled, Lake Kasumigaura and Lake Zurich.

Lake Bled is of glacial-tectonic origin, located in the Julian Alps in north-western Slovenia (46.3644° N, 14.0947° E). It occupies an area of 1.4 km^2, with a maximum depth of 30.1 m and an average depth of 17.9 m. The measurements, performed by the Slovenian Environment Agency, consist of physical, chemical and biological data for the period from 1996 to 2002. All the measurements were performed once a month and depth-averaged for the upper 10 m of the lake. To obtain daily approximations, the data was interpolated with a cubic spline algorithm and daily samples were taken from the interpolation (Atanasova et al., 2006c).

Lake Kasumigaura, is located 60 km to the north-east of Tokyo, Japan (36.0403° N, 140.3942° E). It has an average depth of 4 m, a volume of 662 million cubic metres, and a surface area of 220 km^2. The data set comprises monthly measurements in the period from 1986 to 1992. Again, to obtain daily approximations, the measurements were interpolated using linear interpolation and daily samples were taken from the interpolation (Atanasova et al., 2006a).

Lake Zurich is located in the south-western part of the canton of Zurich in Switzerland (42.1970° N, 88.0934° W). It has an average depth of 49 m, volume of 3.9 km^3 and a surface area of 88.66 km^2. The data comprises measurements performed by the Water Supply Authority of Zurich in the period from 1996 to 2002. The measurements, taken once a month, include profiles of physical, chemical and biological variables from 19 different sites. They were weight averaged to the respective epilimnion (upper ten metres) and hypolimnion (bottom ten metres) depths. The data was interpolated with a cubic spline algorithm and daily samples were taken from the interpolation (Dietzel et al., 2013).

We use the same structure of population dynamics model in all three aquatic ecosystems. It includes a single equation (ODE) for a system variable representing the phytoplankton biomass (measured as *chlorophyll-a* in Lake Kasumigaura). The exogenous variables include the concentration of zooplankton *Daphnia hyalina* (available only for Bled and Zurich), dissolved inorganic nutrients of nitrogen, phosphorus, and silica (ammonia in Lake Kasumigaura), as well as two input variables representing the environmental influence of water temperature and global solar radiation (light).

Table 2 provides a summary of the data sets we used in the experiments. For each aquatic ecosystem, we used seven data sets corresponding to the last seven years of available measurements. Five of these were used (one at a time) for training the base-models, one was used for validating the models in the process of selecting the ensemble constituents, and one was used to measure the predictive performance of the learned models and ensembles.

In each learning experiment, we take a single (year) training data set, learn a single model or an ensemble using the training and the validation data set, and test the predictive performance of the learned models on the test data set. We therefore perform 15 experiments. In the tables, we label them by the label of the training set used, which is comprised of the initial letter of the lake name, followed by a digit for 1 to 5 corresponding to each of the

Table 3
Comparison of the predictive performance (ReRMSE on test data) of the different combinations of methods for selecting (Validation and Regular) and methods for combining (Average, Weighted Average and Weighted Median) ensemble constituents applied to the 15 data sets. The numbers in bold are the best performance figures for the given data set.

	Validation			Regular		
	Average	Weighted average	Weighted median	Average	Weighted average	Weighted median
B1	1.08	1.09	1.08	**1.06**	1.09	**1.06**
B2	**1.10**	**1.10**	1.12	1.24	1.24	1.27
B3	0.97	0.97	**0.96**	1.05	1.04	1.05
B4	0.78	0.78	0.76	**0.74**	**0.74**	**0.74**
B5	0.87	0.88	0.87	0.63	0.63	**0.61**
K1	**0.73**	0.74	0.80	0.93	0.92	0.93
K2	1.64	**1.63**	1.64	1.84	1.80	1.60
K3	**0.87**	**0.87**	0.89	0.91	0.93	0.94
K4	**0.78**	**0.78**	0.79	0.99	0.99	0.98
K5	0.74	**0.73**	**0.73**	0.98	0.99	1.00
Z1	**0.78**	**0.78**	0.79	1.03	1.03	0.99
Z2	**0.88**	**0.88**	0.90	1.08	1.09	1.01
Z3	0.95	0.99	**0.91**	1.21	1.22	1.19
Z4	**0.96**	0.97	0.97	0.97	0.97	0.98
Z5	1.32	1.38	**1.26**	1.39	1.45	1.40

Table 4
Comparison of the descriptive performance (ReRMSE on training data) of the different combinations of methods for selecting (Validation and Regular) and methods for combining (Average, Weighted Average and Weighted Median) ensemble constituents applied to the 15 data sets. The numbers in bold are the best performance figures for the given data set.

Case	Validation			Regular		
	Average	Weighted average	Weighted median	Average	Weighted average	Weighted median
B1	0.27	0.36	0.27	**0.20**	0.29	0.21
B2	0.58	0.58	0.60	**0.24**	**0.24**	**0.24**
B3	0.37	0.40	0.41	**0.29**	**0.29**	**0.29**
B4	0.34	0.34	0.33	0.26	**0.25**	0.26
B5	0.56	0.56	0.56	**0.14**	**0.14**	**0.14**
K1	1.07	1.06	1.13	0.69	**0.68**	0.72
K2	0.83	0.84	0.84	**0.76**	**0.76**	**0.76**
K3	0.68	0.66	0.87	**0.54**	0.56	0.55
K4	0.68	0.74	0.84	**0.40**	**0.40**	**0.40**
K5	0.46	0.49	0.53	0.37	0.37	**0.38**
Z1	0.88	0.87	0.90	0.53	**0.52**	0.55
Z2	0.85	0.87	0.92	0.54	**0.53**	0.54
Z3	0.82	0.82	0.84	0.69	**0.68**	0.70
Z4	0.79	0.79	0.79	**0.77**	**0.77**	**0.77**
Z5	0.61	0.61	0.60	**0.51**	**0.51**	0.53

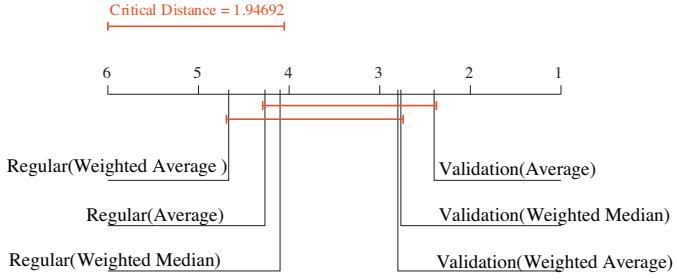


Fig. 4. Comparison of the average ranks of different combinations of methods for selecting and methods for combining ensemble constituents in terms of predictive performance (ReRMSE on test data) averaged over 15 data sets.

5 consecutive years of measurements. The labels are thus B1–B5, K1–K5 and Z1–Z5 and B5, for example, denotes the measurements for Lake Bled taken in the fifth year, i.e., the year 2000.

4.3. Evaluation methodology

To apprise the predictive performance of a given model m, we use the relative root mean squared error (*ReRMSE*) (Breiman, 1984), defined as:

$$ReRMSE(m) = \sqrt{\frac{\sum_{t=0}^{n}(y_t - \hat{y}_t)^2}{\sum_{t=0}^{n}(\bar{y} - \hat{y}_t)^2}}, \quad (2)$$

where n denotes the number of measurements in the test data set, y_t and $\hat{y}_t$ correspond to the *measured* and *predicted* value (obtained by simulating the model m) of the system variable y at time point t, and $\bar{y}$ denotes the mean value of the system variable y in the test data set. Note that the usual root mean squared error is normalized here with the standard deviation of the system variable in the test data, thus allowing us to compare the errors of models for system variables measured on different scales. The other interpretation of the normalization term is that it represents the error of a base-line model that predicts the average value of y at each time point t. Thus, the model with *ReRMSE* of 1 has performance equal to that of the base-line "average" predictor. Smaller values of *ReRMSE* indicate better predictive performance.

We observe the performance of different learning algorithms in terms of the predictive performance (*ReRMSE*) of the models learned on each of the 15 data sets. To assess the significance of the differences in performance between different learning algorithms, we use the corrected (Iman and Davenport, 1980) Friedman test (Friedman, 1940) and the post-hoc Nemenyi test (Nemenyi, 1963). This is a standard framework for comparing the predictive performance of different learning algorithms, superior to alternative frameworks as argued by Demšar (2006). The Friedman non-parametric test for multiple hypotheses testing first ranks the algorithms according to their performance (i.e., predictive performance of the trained models) on each combination of train/test data set, and then averages these ranks across all the data set

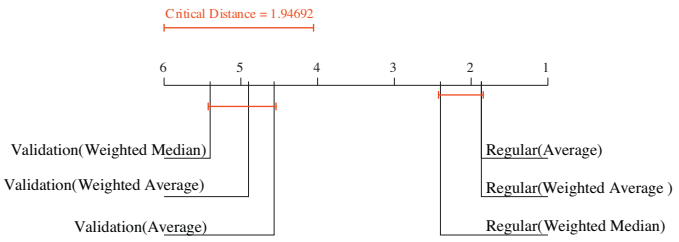


Fig. 5. Comparison of the average ranks of different combinations of methods for selecting and methods for combining ensemble constituents in terms of descriptive performance (ReRMSE on training data) averaged over 15 data sets.

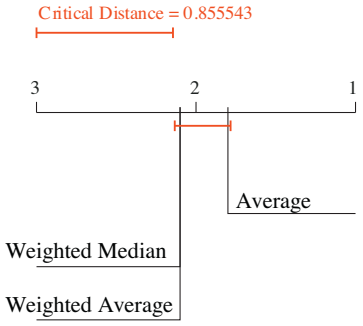


Fig. 6. Comparison of the average ranks of the three schemes for combining base-model simulations (Average, Weighted Average, and Weighted Median) in terms of predictive performance (averaged over 15 data sets). We compare the test errors of the ensembles with 100 base-models, selected by using a separate validation data set.

combinations. If the Friedman test indicates a statistically significant difference, we proceed with performing a Nemenyi test to identify which differences are significant.

The Nemenyi test computes the critical distance between the algorithm ranks at a given level of statistical significance (in our case, we set the significance level threshold at 95 %, $p = 0.05$). Only differences in the average ranks larger than the critical distance are considered significant; for those we can claim that one algorithm outperforms (i.e., performs significantly better than) the other. The results of the Friedman–Nemenyi tests are depicted by using average rank diagrams (Figs. 4–7). In these diagrams, we can see the name of each of the compared algorithms along with its average rank.

Finally, to test the conjecture that the power of ensembles is based on the exploitation of the diversity of the ensemble constituents, we measure the diversity of the ensemble constituents and correlate it to the performance improvement of ensembles over single models. To measure the diversity of the base-models in the ensemble e, we measure the average pairwise difference of the base model simulations

$$Diversity(e) = \frac{1}{\binom{|e|}{2}} \sum_{\{m_1,m_2\}\subset e} \sqrt{\frac{\sum_{t=0}^{n}(y_{1,t} - y_{2,t})^2}{n}}, \quad (3)$$

where $|e|$ denotes the number of base-models in the ensemble e, n the number of measurements in the data set, m_1 and m_2 two models from e, and $y_{1,t}$ and $y_{2,t}$ the simulated values of these models at time point t. To assess the performance improvement of the ensemble e over a single model m, we calculate

$$Improvement(e, m) = -\frac{ReRMSE(e) - ReRMSE(m)}{ReRMSE(m)}, \quad (4)$$

where the model m is learned from the complete training set, and the base-models from the ensemble e are learned on different bootstrap samples of the training set. We draw a scatter plot that depicts the correlation between ensemble diversity and performance improvement and calculate the Pearson Correlation Coefficient between them.

5. Results

In this section, we present and discuss the results of the empirical evaluation. We first explore some design decisions employed in the ensemble learning algorithm. We then analyze the improvement of performance obtained by replacing single models with ensembles and investigate the relation between the diversity of the ensemble constituents and the performance improvement. Finally, we visually compare the simulations of ensembles with the simulations of single models on both training and test data.

5.1. Selecting and combining ensemble constituents

One of the important decision when designing an algorithm for learning ensembles is how to select the models to be included in the ensemble. The base-line method (labelled *regular* in the tables and figures below) often employed in ensemble learning algorithms, is to select the models performing best on the bootstrap samples of the training data on which they were learned. Here, to avoid overfitting of the training data, we also consider an alternative method, labelled *validation* in the tables and figures below. We still build models on different bootstrap samples of the training data, but we evaluate their performance on a single separate validation set. In this first series of experiments, we construct ensembles of 100 base-models.

Table 5
Comparison of the predictive performance (ReRMSE on test data) of the single model and bagging ensembles that include 5, 10, 25, 50, and 100 base-models on the 15 data sets. The numbers in bold are the best performance figures for the given data set.

Case	single model	Ensemble 5	Ensemble 10	Ensemble 25	Ensemble 50	Ensemble 100
B1	1.22	**1.06**	**1.06**	1.07	1.09	1.08
B2	1.14	1.24	1.14	**1.09**	**1.09**	1.10
B3	1.07	1.03	0.99	**0.97**	**0.97**	**0.97**
B4	0.92	**0.75**	**0.75**	0.76	0.77	0.78
B5	0.92	0.87	0.89	**0.86**	0.87	0.87
K1	0.74	0.75	**0.73**	0.74	0.74	**0.73**
K2	2.20	1.50	**1.38**	1.43	1.52	1.64
K3	0.96	**0.86**	**0.86**	0.87	0.87	0.87
K4	0.78	0.77	**0.76**	0.77	0.78	0.78
K5	**0.72**	0.85	0.79	0.73	0.74	0.74
Z1	0.78	**0.77**	**0.77**	0.78	0.78	0.78
Z2	0.95	0.89	0.94	**0.88**	0.87	**0.88**
Z3	0.99	0.92	0.96	0.93	**0.92**	0.95
Z4	**0.94**	0.99	0.98	0.96	0.97	0.96
Z5	1.65	1.30	**1.11**	**1.11**	1.22	1.32

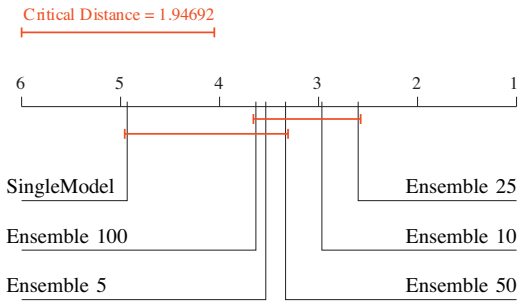


Fig. 7. Comparison of the average ranks of the single model and bagging ensembles that include 5, 10, 25, 50, and 100 base-models in terms of their predictive performance (ReRMSE on test data) averaged over the fifteen data sets.

Table 3 and Fig. 4 summarize the results of the comparison between the regular and the validation methods. From the table, we can see that for all but three data sets (B1, B4 and B5), the selection method based on a separate validation data set outperforms the regular base-line method. For four data sets (B3 and Z1–Z3), the use of validation lowered the ensemble *ReRMSE* below the value of 1, improving over the base-line "average" predictor. The Friedman–Nemenyi test (and the corresponding diagram in Fig. 4) confirms the significance of the observed superiority of the validation method: All methods that use validation are ranked better than those that use the training set, where the best method (validation used together with the average combining scheme) *significantly* outperforms the worst method (regular selection used with the weighted average combining scheme).

Note that we made the implicit conjecture that the regular selection method overfits the training data. The results presented in Table 4 and Fig. 5 confirm the validity of this conjecture. From the table, we can see that the regular method consistently leads to significantly smaller errors on the training data. Together with the results in Table 3/Fig. 4 these show a clear case of overfitting — while being superior on the training data, regular selection leads to inferior predictive performance as compared to validation-based selection.

Taken together, the above results show that the selection method based on validation is the right choice when learning ensembles of process-based models.

We next consider the choice of an appropriate method for combining the simulations/predictions of the models in the ensemble. Here, we choose among the three methods commonly used in learning ensembles of regression models: average, weighted average, and weighted median (Breiman, 1984; Drucker, 1997). Above, we considered these in combination with the regular and the validation selection methods: Here, we consider them in combination with the validation selection method only.

The results of the Friedman–Nemenyi test depicted in Fig. 6 show the lack of significant difference among the average ranks of the three combination schemes. However, the simplest among them, i.e., the 'average' method has also the best rank. Therefore, despite the lack of significant difference, we can make the decision to use 'average' as the most appropriate method for combining the predictions of the base-models in ensembles of PBMs.

In all the experiment performed so far, we learned ensembles of 100 base-models. However, the optimal number of base-models can depend on the type of the base-models in the ensemble. In the next series of experiments, we aim at identifying an optimal number of base-models to be included in the ensembles of PBMs for modelling aquatic ecosystems. To this end, we compare the predictive performance of ensembles consisting of 5, 10, 25, 50, and 100 models (learned on bootstrap samples) with the performance of a single model (learned on the complete data set).

Table 6
Diversity of the base-models and the relative improvement of the ensemble error over the error of the single model (given as percentage) for the fifteen data sets.

Case	Diversity	Relative improvement (%)
B1	0.354	12.27
B2	0.561	4.48
B3	0.230	9.24
B4	0.617	17.45
B5	0.270	6.81
K1	1.010	1.04
K2	1.030	35.06
K3	0.543	9.45
K4	0.598	1.02
K5	0.605	−1.53
Z1	0.089	0.69
Z2	0.234	7.23
Z3	0.223	5.72
Z4	0.125	−2.33
Z5	0.285	32.69
Pearson *r*	–	0.274

Table 5 and Fig. 7 summarize the results of these experiments. Comparing the predictive performance of thr different ensembles, we can see that for eight (out of fifteen) data sets, the ensemble of 10 base-models performs best, followed by the ensemble of 25 base-models, performing best for five data sets. The corresponding Friedman–Nemenyi diagram shows that the ensemble of 25 base-models is ranked best among all the ensembles. Note, however, that the critical distance on the same diagram shows that there is no significant difference in performance between any pair of ensembles with different numbers of constituents. Despite the lack of a significant difference, we are going to choose the ensembles with 25 base-models, which are ranked best, to be the subject of the further experiments.

When it comes to comparing the performance of the ensembles to that of a single model, the Friedman–Nemenyi test shows that the ensembles consisting of 10 and 25 models *significantly* outperform the single models. The tabular comparison of the predictive performance (Table 5) shows that, for all but two data sets (K5 and Z4), a single model performs worse than an ensemble. However, for these two data sets, the difference in the performance between the single model and an ensemble of 25 base-models is almost imperceptible.

These results clearly confirm the central hypothesis of this paper that the ensembles of PBMs significantly outperform a single PBM model. The optimal design choices to be used for learning ensembles are as follows: perform 25 iterations, in each of them select the best model with respect to the error measured on a separate validation set, and combine the ensemble constituents using the average combining scheme.

5.2. Ensemble diversity and performance improvement

The experimental results presented above show that ensembles of PBMs outperform single PBM models. Here, we further analyze the improvement and its relation to the diversity of the simulations of the ensemble constituents. To this end, we first measure the relative improvement of the performance obtained by using an ensemble as compared to using a single model. Then, we measure the diversity of base-models in the ensemble. Finally, we analyze the correlation between the two.

Table 6 and Fig. 8 summarize the results of these experiment. First, Table 6 confirm our previous finding: ensembles outperform single models in all but two data sets (K5 and Z4). Note that the loss of performance of the ensemble is minor (below 3%) for these two data sets. On the other hand, the gain in performance (performance

improvement) can be substantial and reach up to 17% for Lake Bled (B4), 35% for Lake Kasumigaura (K2) and 33% for Lake Zurich (Z5).

Furthermore, we observe a varying degree of diversity between ensemble constituents for different data sets – diversity varies from 0.125 to 1.030. The scatter plot in Fig. 8 shows weak positive correlation between ensemble diversity and relative improvement of performance. The measured Pearson correlation coefficient of 0.274 is certainly neither high nor significant. Still, the positive correlation is in line with the ensemble literature assumption that ensembles perform well by exploiting the diversity of their constituents (Kuncheva and Whitaker, 2003).

5.3. Simulating ensembles

Finally, in the last series of experiments, we visually inspect the difference between the simulations of ensembles and simulation of single models. We selected one data set for each of the three aquatic ecosystems considered in the experiments and simulated the ensemble and the single model on both training and test data. The simulation on test data is in line with the predictive modelling setting used throughout the experiments presented in this paper. The simulation on the training data is in line with the previous work on building descriptive models of aquatic ecosystems, where only the performance on training data is considered (Čerepnalkoski et al., 2012; Taškova et al., 2012; Atanasova et al., 2006c).

In Fig. 9, we present the simulations of ensembles and single models in both the predictive (graphs on the left-hand side) and the descriptive scenario (graphs on right-hand side), for each of the three lakes. The first row presents the simulations for Lake Bled, the second for Lake Kasumigaura and the last for Lake Zurich. The visual comparison confirms the superiority of the ensembles in the predictive scenario. Note that only ensembles lead to acceptable reconstruction of the population dynamics for the test data sets. But more importantly, given the nature of the ensembles (that avoid overfitting), they do not seem to have lower descriptive performance; they still capture the population dynamics of the phytoplankton on training data. Thus, we can conclude that the ensembles can be applied in both predictive and descriptive scenarios.

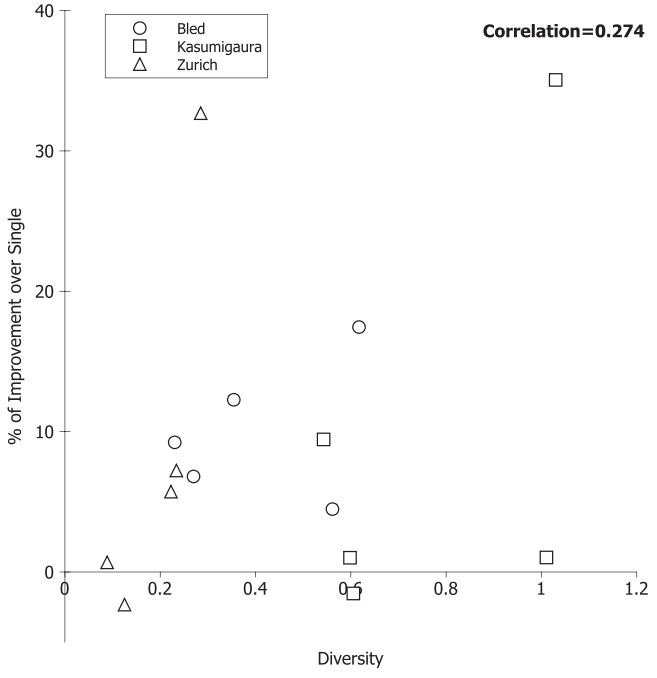


Fig. 8. Scatter plot depicting the correlation between the diversity of the base-model predictions and the relative improvement of error between an ensemble and a single model for the 15 data sets.

One interesting case is Lake Zurich (Fig. 9e and f), where despite the high relative error (above 1), we can see that the ensemble accurately captures the phytoplankton dynamics with a slight phase shift.

5.4. Summary

We can summarize the results of our experiments as follows. When learning ensembles of PBMs, one should use a separate validation data set in addition to the training one when learning the base-models included in the ensemble. For combining the simulation of constituent process-based models, one should use the simplest combining scheme, i.e., averaging. The optimal ensembles of PBMs consist of a relatively low number of constituent models, ranging between 10 and 25.

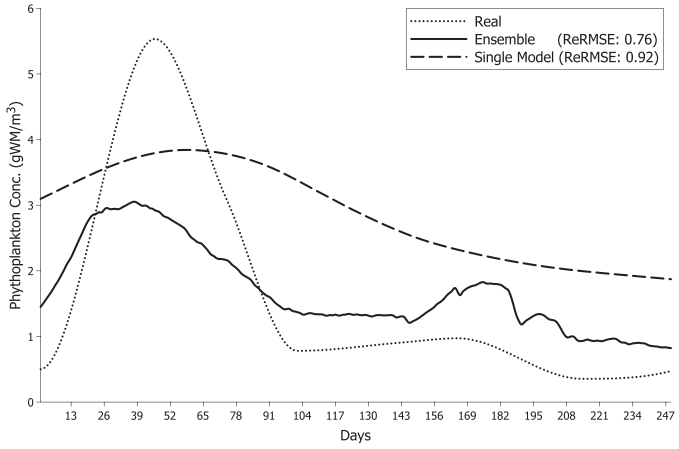
Furthermore, the ensembles of 25 base-models selected using a validation data set and combined with the average combining scheme significantly outperform single models of population dynamics in aquatic ecosystems. The improvement of performance between an ensemble and a single model is positively related to the diversity of the ensemble constituents – the higher the diversity, the greater the improvement. However, the correlation between the diversity and the performance gain is weak, as a consequence of the modest diversity between the base-models. Finally, the simulations show that ensembles are applicable for both predictive modelling tasks, where prediction of the future system behaviour (test data error estimates) is of central interest, as well as descriptive modelling tasks, where the focus is on explaining the observed behaviour (training data error estimates).

6. Discussion

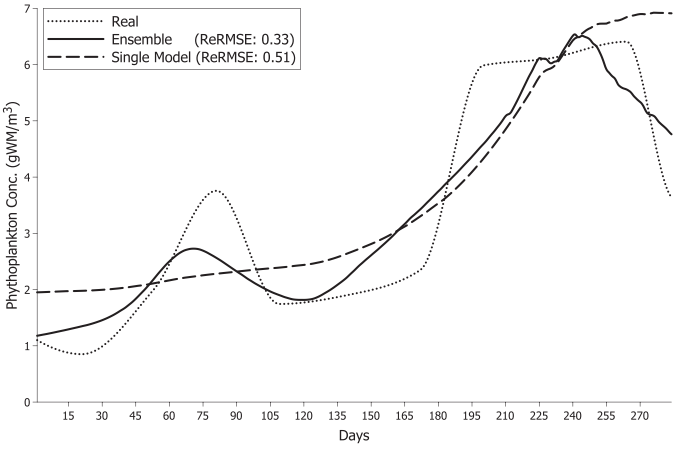
In this section, we discuss the method we propose and its results, and put them in the context of related work.

The work presented in this paper follows two different lines of research. First, it extends the state-of-the-art in the paradigm of equation discovery. More specifically, we build upon previous methods for learning process-based models, that have proven successful for automated modelling of population dynamics in a number of aquatic ecosystems (Todorovski and Džeroski, 2007; Čerepnalkoski et al., 2012; Bridewell et al., 2008). Second, it follows the basic principles of ensemble learning, and translates them into a methodology for modelling dynamic systems. Our work is closely related to that of Bridewell et al. (2005), where the authors use ensemble methods to establish better descriptive models by tackling the over-fitting problem. Their approach is based on integrating the model structures of ensemble constituents into a single model. This model still provides a process-based explanation of the observed system structure, while being more robust in terms of over-fitting observed data. The evaluation of overfitting is performed by a variant of the general cross-validation method, where samples of data are kept out of the training set and are used to estimate the model error. While this method provides estimates of model error on unseen data, these estimates are not related to the predictive performance of the model, i.e., its ability to predict future system behaviour beyond the time-period captured in training data.

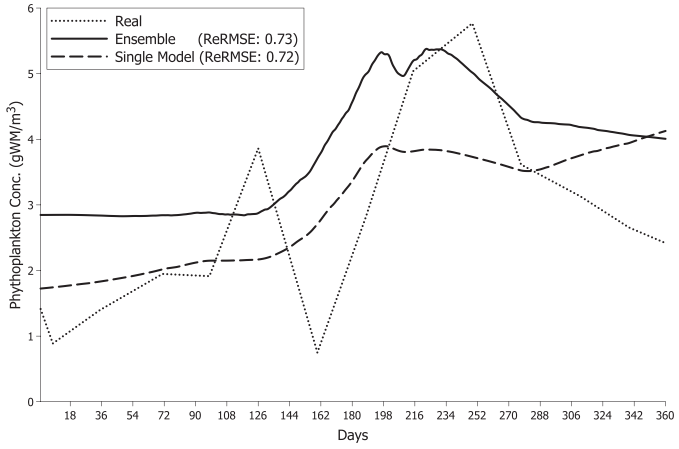
The studies of Whigham and Recknagel (2001) and Cao et al. (2008) are also related to our work, as they use differential equations to model the dynamics of lake ecosystems. However, they start from a modelling assumption that includes a fixed structure of model equations and employ genetic algorithms to estimate the values of the model parameters. While Whigham and Recknagel



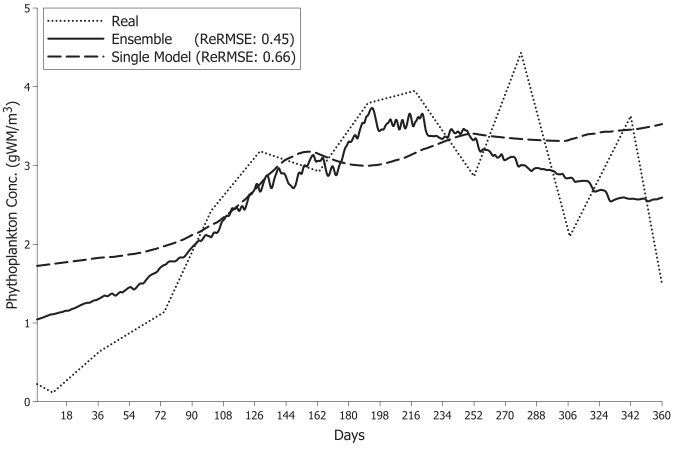
(a) Test Bled 4



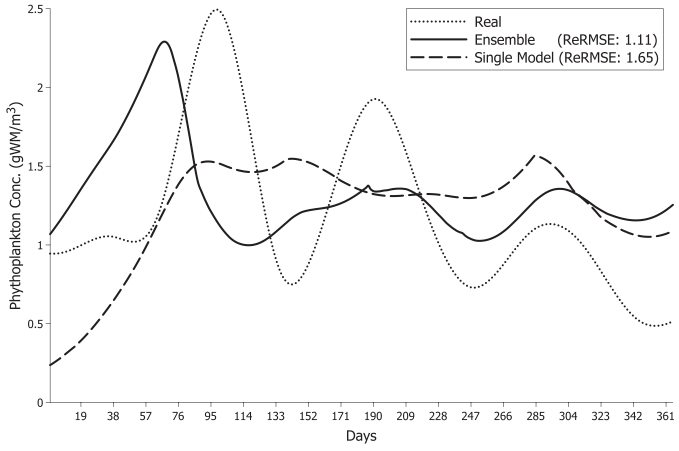
(b) Train Bled 4



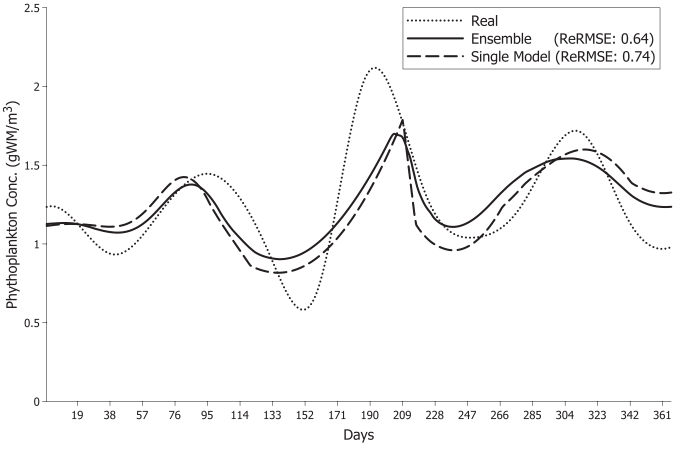
(c) Test Kasumigaura 5



(d) Train Kasumgaura 5



(e) Test Zurich 5



(f) Train Zurich 5

Fig. 9. Simulations of single models and ensembles compared to the measured data, for three pairs of test and training data sets.

(2001) also consider the use of genetic programming to explore a number of different model structures, the obtained equations are not cast in the form of process-based models and therefore do not provide insight into the processes and entities that govern the dynamics of the observed systems.

Muttil and Chau (2006) also use genetic programming and artificial neural networks to model algal blooms in coastal marine ecosystems. While their aim is similar to ours, i.e., to obtain predictive models of algal biomass dynamics, they focus exclusively on black-box models of population dynamics. The models they

consider are based on equations and neural networks and fail to provide insight into the structure of the observed system. They also consider a different time scale for predicting the future system behaviour: While we focus in our experiments on a whole-year prediction the population dynamics, Muttil and Chau (2006) use their models for making one-week-ahead predictions of algal blooms.

The ensemble method proposed in this paper aims at improving the generalization power of process-based models, in terms of achieving predictive performance gain over the state-of-the-art process-based modelling approaches. However, when learning ensembles of process-based models, there is a trade-off between two conflicting requirements: predictive performance and interpretability. The increase of predictive accuracy comes at the cost of losing the interpretability of the learned ensemble model. Moreover, Breiman (1996a) states that bagging can improve the predictive performance when the ensemble is comprised of unstable base-models, such as decision trees, whose predictions sufficiently vary with small variations in the training set. In this case, high diversity of the ensemble constituents can be easily achieved. Our empirical evaluation shows that the ensemble constituents have only modest diversity: This may limit the potential for performance gain, even though the diversity is only weakly correlated with predictive performance.

7. Conclusion

7.1. Summary

In this paper, we address the task of learning ensembles of process-based models of dynamic systems and develop a methodology to solve it. Note that the task of learning ensembles of process-based models is a novel task, and has not been considered so far. For this purpose, we extend the state-of-the-art approaches to process-based modelling: We take the notion of ensembles – a collection of base-models, whose predictions are combined to improve the collective performance. In traditional machine learning, this has proved to be an effective method for gaining predictive power.

More specifically, we propose a methodology that adapts the key design principles from learning ensembles for classical machine learning tasks to tasks of modelling dynamic systems. Our approach constructs homogeneous ensembles of process-based models, using bagging as an underlying ensemble method. We identify the main components of the method for learning an ensemble of PBMs (a technique for learning a set of candidate base-models, a technique for selecting the base-models, and a combining scheme specifying how the base-model predictions are combined) and the related design choices.

We conduct an extensive experimental evaluation to identify the appropriate design choices for the proposed ensemble learning method and to test its utility for modelling dynamic systems. We analyze the improvement of performance obtained by ensembles relative to single models on 15 different data sets. Moreover, we investigate the relation between the diversity of the ensemble constituents and the performance improvement. Finally, we visually compare the simulations of ensembles with the simulations of single models in both descriptive and predictive scenarios.

We conduct the empirical evaluation on the task of modelling population dynamics in aquatic ecosystems. The case studies considered concern modelling phytoplankton growth, a complex non-linear dynamic process, in three different aquatic ecosystem domains. These include: Lake Bled in Slovenia, Lake Kasumigaura in Japan and Lake Zurich in Switzerland.

The results of the empirical evaluation confirm our central hypothesis. For predictive modelling tasks, ensembles of process-based models perform better than a single model. More precisely, ensembles with a relatively low number of constituents (10–25), chosen on a separate validation data set and combined by averaging, outperform the single model. Moreover, the diversity of the constituents in the ensemble is positively (weakly) correlated with the performance improvement. Finally, after visual inspection of the simulations, we found that the ensembles are applicable to both predictive and descriptive modelling tasks.

7.2. Future work

We have identified a number of limitations of our approach that can be addressed in further work. First, the diversity of the ensemble constituents is modest. One reason for this might be the fact that we use a library with a limited number of modelling alternatives. An extended library of domain-specific knowledge should be used in future experiments in order to reach higher diversity and further study the relation between the diversity and predictive performance in ensembles of process-based models.

Future work can also include the development of alternative methods for data and knowledge sampling that would lead to higher diversity. These include sampling the data variables (in addition to sampling the data instances considered in this paper) and sampling the alternative modelling choices included in the library of domain-specific modelling knowledge. While we limited our focus in this paper on adapting the method of bagging, further work should adapt other ensemble methods to the task of learning process-based models: Methods to be adapted include *boosting* (Drucker, 1997; Freund, 1999; Schapire, 2003) and *random subspaces* (Ho, 1998).

Finally, the experiments performed in this paper were limited to modelling population dynamics in lake ecosystems from historical data. Future experiments can be based on more recent data of the same ecosystems. Also, future work should confirm the results presented in this paper by learning ensembles of process-based models of population dynamics in other aquatic environments, such as marine ecosystems or water-treatments plants (Škerjanec et al., 2014). Other application domains (such as systems neuroscience and systems biology) should also be considered.

Acknowledgements

We thank Panče Panov and Dragi Kocev for the helpful discussions and comments on the work presented in this paper. Thanks to Nataša Atanasova for providing the extensive library for modelling aquatic ecosystems and the data sets for Lake Bled and Lake Kasumigaura. We also would like to thank Anne Dietzel for providing data on Lake Zurich. Finally, we acknowledge the financial support of the European Commission through the project SUMO – Supermodelling by combining imperfect models (grant number ICT-2009-266722) and MAESTRA – Learning from Massive, Incompletely annotated, and Structured Data (grant number ICT-2013-612944).

References

Čerepnalkoski, D., Taškova, K., Todorovski, L., Atanasova, N., Džeroski, S., 2012. The influence of parameter fitting methods on model structure selection in automated modeling of aquatic ecosystems. Ecol. Model. 245, 136–165.

Škerjanec, M., Atanasova, N., Čerepnalkoski, D., Džeroski, S., Kompare, B., 2014. Development of a knowledge library for automated watershed modeling. Environ. Model. Softw. 54, 60–72.

Atanasova, N., Recknagel, F., Todorovski, L., Džeroski, S., Kompare, B., 2006a. Computational assemblage of Ordinary Differential Equations for Chlorophyll-a using a lake process equation library and measured data of Lake Kasumigaura. In: Recknagel, F. (Ed.), Ecological Informatics. Springer, pp. 409–427.

Atanasova, N., Todorovski, L., Džeroski, S., Kompare, B., 2006b. Constructing a library of domain knowledge for automated modelling of aquatic ecosystems. Ecol. Model. 194 (1–3), 14–36.
Atanasova, N., Todorovski, L., Džeroski, S., Remec, R., Recknagel, F., Kompare, B., 2006c. Automated modelling of a food web in Lake Bled using measured data and a library of domain knowledge. Ecol. Model. 194 (1–3), 37–48.
Breiman, L., 1984. Classification and Regression Trees. Chapman & Hall, London, UK.
Breiman, L., 1996a. Bagging predictors. Mach. Learn. 24 (2), 123–140.
Breiman, L., 2001. Random forests. Mach. Learn. 45 (1), 5–32.
Bridewell, W., Asadi, N.B., Langley, P., Todorovski, L.,2005. Reducing overfitting in process model induction. In: Proceedings of the 22nd International Conference on Machine Learning (ICML '05). ACM, pp. 81–88.
Bridewell, W., Langley, P.W., Todorovski, L., Džeroski, S., 2008. Inductive process modeling. Mach. Learn. 71, 1–32.
Cao, H., Recknagel, F., Cetin, L., Zhang, B., 2008. Process-based simulation library SALMO-OO for lake ecosystems. Part 2: Multi-objective parameter optimization by evolutionary algorithms. Ecol. Inform. 3 (2), 181–190.
Cohen, S.D., Hindmarsh, A.C., 1996. CVODE, a stiff/nonstiff ODE solver in C. Comput. Phys. 10 (March (2)), 138–143.
Crisci, C., Ghattas, B., Perera, G., 2012. A review of supervised machine learning algorithms and their applications to ecological data. Ecol. Model. 240, 113–122.
Džeroski, S., Todorovski, L.,1993. Discovering dynamics. In: Proceedings of Tenth International Conference on Machine Learning. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 97–103.
Džeroski, S., Todorovski, L., 2003. Learning population dynamics models from data and domain knowledge. Ecol. Model. 170, 129–140.
Demšar, J., 2006. Statistical comparisons of classifiers over multiple data sets. J. Mach. Learn. Res. 7 (December), 1–30.
Dietzel, A., Mieleitner, J., Kardaetz, S., Reichert, P., 2013. Effects of changes in the driving forces on water quality and plankton dynamics in three Swiss lakes – long-term simulations with BELAMO. Freshw. Biol. 58 (1), 10–35.
Drucker, H.,1997. Improving regressors using boosting techniques. In: Proceedings of the 14th International Conference on Machine Learning (ICML '97). Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 107–115.
Durillo, J.J., Nebro, A.J., 2011. jMetal: A Java framework for multi-objective optimization. Adv. Eng. Softw. 42, 760–771.
Freud, Y., Schapire, R.E., 1999. A short introduction to boosting. J. Jpn. Soc. Artif. Intell. 14 (5), 771–780.
Freund, Y.,1999. An adaptive version of the boost by majority algorithm. In: Proceedings of the 12th Annual Conference on Computational Learning Theory (COLT '99). ACM, New York, NY, USA, pp. 102–113.
Friedman, M., 1940. A comparison of alternative tests of significance for the problem of m rankings. Ann. Math. Stat. 11 (1), 86–92.
Ho, T.K., Aug 1998. The random subspace method for constructing decision forests. IEEE Trans. Pattern Anal. Mach. Intell. 20 (8), 832–844.
Iman, R.L., Davenport, J.M., 1980. Approximations of the critical region of the Friedman statistic. Commun. Stat. – Theory Methods 9 (6), 571–595.
Knudby, A., Brenning, A., LeDrew, E., Feb. 2010. New approaches to modelling fish–habitat relationships. Ecol. Model. 221 (3), 503–511.
Kuncheva, L.I., Whitaker, C.J., 2003. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. Mach. Learn. 51 (May (2)), 181–207.
Langley, P.W., Simon, H.A., Bradshaw, G., Zytkow, J.M., 1987. Scientific Discovery: Computational Explorations of the Creative Processes. The MIT Press, Cambridge, MA, USA.
Luenberger, D., 1979. Introduction to Dynamic Systems: Theory, Models, and Applications. Wiley, NJ, USA.
Maclin, R., Opitz, D., 1999. Popular ensemble methods: An empirical study. J. Artif. Intell. Res. 11, 169–198.
Muttil, N., Chau, K.W., 2006. Neural network and genetic programming for modelling coastal algal blooms. Int. J. Environ. Pollut. 28 (3–4), 223–238.
Nemenyi, P.B., 1963. Distribution-free Multiple Comparisons. Princeton University, Princeton, NJ, USA (Ph.D. thesis).
Okun, O., Valentini, G., Ré, M. (Eds.), 2011. Ensembles in Machine Learning Applications. Vol. 373 of Studies in Computational Intelligence. Springer, Berlin.
Rokach, L., Feb. 2010. Ensemble-based classifiers. Artif. Intell. Rev. 33 (1–2), 1–39.
Schapire, R.E., 2003. The boosting approach to machine learning: an overview. In: Denison, D., Hansen, M., Holmes, C., Mallick, B., Yu, B. (Eds.), Nonlinear Estimation and Classification. Vol. 171 of Lecture Notes in Statistics. Springer, New York, pp. 149–171.
Storn, R., Price, K., 1997. Differential Evolution - A simple and efficient heuristic for global optimization over continuous spaces. J. Glob. Optim. 11 (4), 341–359.
Taškova, K., Korošec, P., Šilc, J., Todorovski, L., Džeroski, S., 2011. Parameter estimation with bio-inspired meta-heuristic optimization: modeling the dynamics of endocytosis. BMC Syst. Biol. 5 (1), 159.
Taškova, K., Šilc, J., Atanasova, N., Džeroski, S., 2012. Parameter estimation in a nonlinear dynamic model of an aquatic ecosystem with meta-heuristic optimization. Ecol. Model. 226, 36–61.
Todorovski, L., Džeroski, S., 2007. Integrating domain knowledge in equation discovery. In: Džeroski, S., Todorovski, L. (Eds.), Computational Discovery of Scientific Knowledge. Vol. 4660 of Lecture Notes in Computer Science. Springer, Berlin, pp. 69–97.
Todorovski, L., Bridewell, W., Shiran, O., Langley, P.W.,2005. Inducing hierarchical process models in dynamic domains. In: Proceedings of the 20th National Conference on Artificial Intelligence. AAAI Press, Pittsburgh, USA, pp. 892–897.
Whigham, P., Recknagel, F., 2001. Predicting Chlorophyll-a in freshwater lakes by hybridising process-based models and genetic algorithms. Ecol. Model. 146 (1–3), 243–251.
Wolpert, D.H., 1992. Stacked generalization. Neural Netw. 5, 241–259.
Zhou, Z.-H., Wu, J., Tang, W., 2002. Ensembling neural networks: Many could be better than all. Artif. Intell. 137 (1–2), 239–263.