LIMNOLOGY
and
OCEANOGRAPHY: METHODS

ASLO

Limnol. Oceanogr.: Methods 00, 2016, 00–00

doi: 10.1002/lom3.10151

Validation methods for plankton image classification systems

Pablo González,[1] Eva Álvarez,[2] Jorge Díez,[1] Ángel López-Urrutia,[2] Juan José del Coz[1]*
[1]Artificial Intelligence Center, University of Oviedo, Gijón, Spain
[2]Centro Oceanográfico de Gijón, Instituto Español de Oceanografía, Gijón, Asturias, Spain

Abstract

In recent decades, the automatic study and analysis of plankton communities using imaging techniques has advanced significantly. The effectiveness of these automated systems appears to have improved, reaching acceptable levels of accuracy. However, plankton ecologists often find that classification systems do not work as well as expected when applied to new samples. This paper proposes a methodology to assess the efficacy of learned models which takes into account the fact that the data distribution (the plankton composition of the sample) can vary between the model building phase and the production phase. As opposed to most validation methods that consider the individual organism as the unit of validation, our approach uses a validation-by-sample, which is more appropriate when the objective is to estimate the abundance of different morphological groups. We argue that, in these cases, the base unit to correctly estimate the error is the sample, not the individual. Thus, model assessment processes require groups of samples with sufficient variability in order to provide precise error estimates.

Since the advent of plankton-imaging systems, there has been a clear need to automate the classification of these images into taxonomic and functional categories. Despite the complexity of the problem from a learning perspective, automatic plankton classification seems to be quite good in terms of accuracy and close to that achieved by professional taxonomists (Benfield et al. 2007). The methods used when building automatic plankton recognition systems differ in many aspects, including the capture device used, image preprocessing, the considered taxonomy, the construction of the training, and test sets, the algorithm used for learning and the validation methods applied to estimate the accuracy of the overall approach. It is therefore virtually impossible to compare the results from different studies and it is not easy to extract general conclusions, except some obvious ones, like the conclusion that accuracy tends to decrease when the number of classes increases. For example, Tang et al. (1998) report accuracies of up to 92% when classifying between six classes, while other authors, like Culverhouse et al. (1996), report 83% accuracy using neural networks and classifying between 23 classes. Table 1 summarizes the diversity of methods used.

However, most of the authors of the papers listed in Table 1 would probably agree with respect to a worrying fact: the performance of plankton recognition systems degrade when they are deployed and have to work in real conditions (Bell and Hopcroft 2008). This means that the model assessment strategies employed are not able to correctly estimate the future performance of these systems. Yet, the techniques applied are those proposed in the statistical literature, like cross-validation. Acknowledging that solving this issue is difficult, the present paper exhaustively discusses it from a formal point of view and proposes a validation methodology that may help to mitigate the problem, suggesting further directions of research. Our proposal is designed to deal with the particular characteristics of plankton recognition systems, focusing on those cases in which the goal is to obtain estimates for complete samples, e.g., the abundance of different groups in unseen samples.

Why do traditional model assessment methods not work in plankton recognition systems? In our opinion, there are two main reasons why the performance of plankton recognition systems is not accurately estimated by model assessment methods.

The first has to do with an imprecise definition of what the actual prediction task is from the learning point of view and how its performance should be assessed. In many cases, error estimates during learning are provided in terms of the classification accuracy at an individual level. Basically, they estimate the probability of classifying an individual example correctly. However, many of these studies are designed to predict the total abundance of the different taxonomic or functional groups. Hence, the actual performance of the model/algorithm should be assessed in terms of the estimated abundance for each group in a sample. We believe that this dichotomy in evaluating the learned model at an

*Correspondence: juanjo@uniovi.es

Table 1. Summary of the training sets and validation methodologies used in several papers. Note that results may not be comparable due to the variety of datasets and methods used in the experiments. The abbreviations used are the following: manually selected (man. sel.) examples (ex.), classes (cl.), samples (sa.), phytoplankton (phyto.), zooplankton (zoo.), cross-validation over training sets (CV), Hold-out applied over testing sets (HO), Resubstitution (R), Accuracy (ACC), Precision (P), Recall (RE), True Positives (TP), False Positives (FP), Confusion Matrices (CM), Abundance estimate (AE), Abundance comparison with graphics (AC), Regression analysis (RA) and Kullback–Leibler Divergence (KLD).

Paper	Datasets	Validation method	Performance metrics
Jeffries et al. (1984)	315 man. sel. ex., 8 cl. (zoo.)	HO (265 ex. for training and 50 ex. for testing)	ACC (89%)
Gorsky et al. (1989)	3 cl. (phyto). 30 mL of each cl. for testing	HO (50 ex./cl. for training)	AE
Simpson et al. (1991)	100 man. sel. ex., 2 cl. (phyto.)	HO	ACC (90%)
Boddy et al. (1994)	42 cl. (phyto.) (200 man. sel. ex./cl.)	HO (100 ex./cl. for testing)	ACC (half of the cl. over 70%)
Culverhouse et al. (1996)	5000 man. sel. ex., 23 cl. (phyto.)	HO (100 ex. for training, rest for testing)	ACC (83%)
Frankel et al. (1996)	6000 man. sel. ex., 6 cl. (phyto.)	R, HO (1,000 extra ex. for testing)	ACC (98%), CM
Tang et al. (1998)	2000 man. sel. ex., 6 cl. (zoo. and phyto.)	HO (1/2 training, 1/2 testing)	ACC (95%)
Boddy et al. (2000)	1st) 61 cl. (phyto.) 2nd) 52 cl. (phyto.)	HO (500 ex./cl. for training and 500 ex./cl. for testing)	ACC (77% 1st. dataset, 73% 2nd. dataset)
Embleton et al. (2003)	235 ex., 4 cl. (phyto.)	HO (235 for training, 500 ex. for testing)	CM, AC
Luo et al. (2003)	1st) 1,258 man. sel. ex., 5 cl. 2nd) 6,000 man. sel. ex., 6 cl. (zoo. and phyto.)	10-fold CV	ACC (90% 1st. dataset, 75% 2nd. dataset), CM
Beaufort and Dollfus (2004)	4150 man. sel. ex., 11 cl. (150 ex./cl. + 2500 ex. in class others)	HO (50 ex./cl. for testing)	AC (91%), RA
Davis et al. (2004)	D_1 1,920(5cl.) D_2 1,527(7) D_3 1,671(7) D_4 1,400(7) 200ex/cl T_1 19,521(7) T_2 20,000(7) T_3 time series (zoo. and phyto.)	R, CV, HO	ACC (93% for R, 84% for CV and 63% for HO), CM, AC, RA
Grosjean et al. (2004)	1st) 1,035 man. sel. ex, 8 cl. (zoo.) 2nd) 1,127 man. sel. ex., 29 cl. (zoo.)	HO (2/3 training, 1/3 testing, 100 repetitions)	ACC (1st 85%, 2nd 75%)
Luo et al. (2004)	1st) 1,285 man. sel. ex., 5 cl. 2nd) 6,000 man. sel. ex., 6 cl. (phyto. and zoo.)	10-fold CV over both datasets	ACC (1st 90%, 2nd 75.6%), CM
Blaschko et al. (2005)	982 man. sel. ex., 13 cl. (zoo. and phyto.)	10-fold CV	ACC (71%)
Hu and Davis (2005)	20,000 man. sel. ex., 7 cl. (zoo. and phyto.)	HO (200 ex. for training, 200 ex. for testing)	ACC (72%), KLD
Lisin et al. (2005)	1826 man. sel. ex., 14 cl. (phyto. and zoo.)	10-fold CV	ACC (65.5%), CM
Luo et al. (2005)	8440 man. sel. ex., 5 cl. (phyto. and zoo.)	HO (7,440 ex. for training, 1,000 ex. for testing)	ACC (88%)
Hu and Davis (2006)	20,000 man. sel. ex., 7 cl. (zoo. and phyto.)	HO	TP, FP, CM, AC
Tang et al. (2006)	3147 man. sel. ex., 7 cl. (phyto. and zoo.)	R	ACC (91%), CM
Sosik and Olson (2007)	D 3,300 ex. (22 cl.) 150 ex./cl. T_1 3,300(22) T_2 19,000 (phyto.)	HO T_3 15 sa.	ACC (88%), R, P, AC (T_3)
Bell and Hopcroft (2008)	63 cl. 10-30 ex./cl. (zoo.)	CV, HO	ACC (82%), CM, AE, RA

TABLE 1. Continued

Paper	Datasets	Validation method	Performance metrics
Gislason and Silva (2009)	D_1 1,135 ex. (34 cl.), D_2 1,139 ex. (25 cl.), D_3 1,174 ex. (19 cl.), T 17sa. (zoo.)	10-fold CV, HO	ACC, CM, P, AE, RA (T)
Gorsky et al. (2010)	5–35 cl. (phyto. and zoo.), 300 ex./cl.	CV	TP, FP, CM, AC
Zhao et al. (2010)	3119 man. sel. ex., 7 cl. (phyto. and zoo.)	10-fold CV	ACC (93.27%), CM
Ye et al. (2011)	154,289 ex., 26 cl. (zoo.)	HO (50% for training, 50% for testing)	ACC (69%), AC
Álvarez et al. (2012)	526 sa., 86 sa. for training, 17 sa. for testing (61,700 ex.), 6 cl. (phyto. and zoo.)	HO	ACC (86%), CM, P, RE, AC
Vandromme et al. (2012)	14 cl. 9668 ex. for training (zoo.)	HO (26,027 ex. in 22 sa. for testing)	CM, AC, RA
González et al. (2013)	5145 man. sel. ex., 5 cl. (phyto. and zoo.)	Fivefold CV (repeated twice)	ACC (93,6%), P, RE
Lindgren et al. (2013)	50 sa., 5 depths, 17 cl. (zoo.)	CV	ACC (81.6%), P, AE (1, 5 sa.)
Ellen et al. (2015)	725,516 ex. (46 sa.), 24 cl. (phyto. and zoo.)	HO (80% for training, 20% for testing	RE (88% with 8 cl.), CM
Orenstein et al. (2015)	3.4 million ex., 70 cl. (phyto. and zoo.)	HO (20% for training, 80% for testing)	ACC (93.8%)
Dai et al. (2016)	9460 man. sel. ex., 13 cl. (zoo.)	HO (80% for training, 20% for testing)	ACC (93.7%)
Faillettaz et al. (2016)	1.5 million ex., 14 cl. (phyto. and zoo.)	HO (5,979 man. sel. ex. for training)	ACC (56.3%), P. for biological groups (84%), AC

individual level during training, but using it to estimate total group abundance per sample during "production" should be considered when validating plankton recognition systems. When the goal is to obtain an accurate estimate of the abundance per class, the learning problem is different to when the goal is to classify each image correctly. The former is not a classification task, as the model should return simply an estimate for the whole sample. The performance at an individual level is secondary in such cases.

There are, in fact, some methods whose final estimations are not just based on the number of examples classified for each class (Solow et al. 2001; Lindgren et al. 2013). Unfortunately, most experimental studies focus only on obtaining error estimates for individual predictions. Only a few papers analyze the performance of the model when a global magnitude, typically abundance, is predicted. Different techniques are applied in these papers:

- Confusion matrix. The abundance of each group can be estimated from a confusion matrix (Gislason and Silva 2009; Vandromme et al. 2012; Lindgren et al. 2013). The problem is that the information of the confusion matrix comprises just one sample, the complete testing set. This is equivalent to estimating the classification accuracy at an individual level using only one example.
- Graphically. Some papers use graphs to compare the actual and the predicted magnitude for a set of samples (Davis et al. 2004; Sosik and Olson 2007; Lindgren et al. 2013). The problem is that performance cannot be measured numerically using only graphs.
- Regression analysis. This is carried out to analyze the relationship between both values, observing whether they are well correlated; see, for instance, (Davis et al. 2004; Bell and Hopcroft 2008; Gislason and Silva 2009). R^2 is a good measure to assess fit accuracy, but does not measure prediction accuracy so well.

In addition of these techniques, a precise estimate of the error for the target magnitude should be provided using a group of samples. This estimate will be more useful once the model is deployed. Therefore, our unit in the model assessment process is not the individual example, but the sample, i.e., a group of individual examples. The most important element in our proposal is that the datasets should be composed of a collection of actual complete samples.

The second problem arises from another intrinsic property of plankton recognition problems: changes in data distribution (Haury et al. 1978), also called dataset shift (Moreno-Torres et al. 2012). This drift occurs when the joint distribution of inputs (description of the individuals) and outputs (classes) differs between training and test stages. For instance, when the probability of a class (e.g., diatoms) changes or when the characteristics of the individuals of such class change (e.g., the size distribution of diatoms

varies) or both things together (e.g., the proportion of diatoms changes and also their size distribution). Data drifts occur in many practical applications for a number of different reasons. However, there are two well documented situations: (1) the sample selection bias introduced in the dataset used during training and/or the validation process, for instance, when the training set is manually built without representing the true underlying probability distribution, and (2) because it is impossible to reproduce the testing conditions at training time, mainly because the testing conditions vary over time and are unknown when the training set is built. Both situations may be found, at different levels, in plankton recognition studies. Focussing on the latter, plankton composition shows natural variability. The concentration of different morphological groups usually varies over space and time and this variation depends on numerous causes. However, this is precisely what the model must capture. In order to achieve this goal and also to assess its future performance, the collection of samples that compose the dataset should contain sufficient variability. So once again, variability in terms of individuals, which is the current trend, should shift to variability in terms of samples. Otherwise, it is impossible to obtain accurate estimates.

This paper makes two main contributions to the literature. The first is that of studying how changes in distribution affect the performance of classifiers and assessment strategies. The second is to put forward some guidelines and propose an appropriate model assessment methodology designed to deal with the characteristics of the aforementioned plankton recognition tasks. A relatively large dataset, composed of 60 different samples and 39,613 examples, was used to analyze both aspects. The dataset was captured using a FlowCAM (Sieracki et al. 1998) in the Bay of Biscay and off the northern coast of the Iberian Peninsula.

Material and methods

Learning task

Supervised classification tasks require as input a dataset $D=\{(x_i, y_i) : i=1 \dots n\}$, in which x_i is the representation of an individual in the input space $\mathcal{X}$ and $y_i \in \mathcal{Y}=\{c_1, \dots, c_l\}$ is its corresponding class. The goal of a classification task is to induce from D a hypothesis or model

$$h : \mathcal{X} \rightarrow \mathcal{Y}=\{c_1, \dots, c_l\}, \tag{1}$$

that correctly predicts the class of unlabeled query instances, x. A typical example of this kind of learning problem is the prediction of a disease. The input space, $\mathcal{X}$, would be the symptoms of the patient and h returns the most probable disease from $\mathcal{Y}$. Obviously, patients are interested in knowing how accurate h is. Hence, the assessment strategy must estimate the probability that h correctly predicts the disease of a random patient, x.

Most approaches solve plankton recognition tasks using a classifier, including those aimed at returning aggregate estimates. For instance, predicting the abundance per unit of volume for class c_j of a dataset, D, can be computed using a classifier, h:

$$\bar{h}(D, c_j)=\frac{1}{v}\sum_{x_i \in D} I(h(x_i)=c_j), \tag{2}$$

where v is the volume and $I(p)$ is the indicator function that returns 1 if p is true and 0 otherwise. This approach is called "classify and count" in the context of quantification learning (Forman 2008) as individual instances are first classified by h and then counted to compute the estimate for the whole sample, D. Formally, the aforementioned learning task takes the form $\bar{h} : \mathcal{X}^n \times \mathcal{Y} \rightarrow \mathbb{R}$, if we wish to predict one magnitude for a given class, or the form $\bar{H} : \mathcal{X}^n \rightarrow \mathbb{R}^l$, if we wish to make a prediction for all classes together. Notice that $\bar{H}$ can be computed using $\bar{h}$ in (2) because $\bar{H}(D)=(\bar{h}(D, c_1), \dots, \bar{h}(D, c_l))$. Notice that both, $\bar{h}$ and $\bar{H}$, do not require an individual example as input, but a sample denoted as $\mathcal{X}^n$ representing a set of a variable number of instances from the original input space $\mathcal{X}$.

There are two reasons why the classify and count approach is so popular. First, it is a straightforward solution using any off-the-shelf classifier. However, it is not the only possible approach; there exist other alternatives whose analysis falls outside the scope of this paper. One such method was proposed by Solow et al. (2001) and applied by Lindgren et al. (2013). In fact, the classify and count approach is outperformed by other methods according to the quantification literature (Forman 2008; Barranquero et al. 2013).

The second reason is the false belief that if you build the best possible classifier, then you will also have the most accurate estimates at an aggregated level too. This is simply not true (Forman 2008). The only case when it is true is when you have a perfect classifier (accuracy 100%), but this never occurs in real-world applications as difficult as plankton recognition problems. Imagine, for instance, a two-class problem (positive class and negative class) with 200 examples, 100 of each class, and two classifiers, h_1 and h_2. h_1 produces 0 false positives and 20 false negatives, while for h_2, these values are 20 false positives and 20 false negatives. Classifier h_1 has an accuracy of 90%, but it does not estimate the abundance of both classes exactly. While h_2 is a worse classifier, with an accuracy of 80%, the abundance estimates are perfect. Several examples can also be found in plankton recognition papers. For instance, in Lindgren et al. (2013), according to Table 2, page 77, the precision classifying Nonionella examples is 96.7%, with a 3% of error in estimating abundance, while the precision classifying examples of the Multiparticles class is just 68.4% but the error in estimating the abundance is only 1%. In the experiments, we shall see similar examples in the case study (see Fig. 6).

Optimizing precision at an individual level does not mean improving precision at an aggregate level. The performance metrics for both problems are different, so the optimal model for one of them, is rarely also optimal for the other. The perfect classifier is simply an exception. The performance measures for samples require a kind of compensation among the whole sample, as occurs for classifier h_2 in the previous example. There are classifiers that select this kind of model for binary quantification; see, for instance, (Barranquero et al. 2015).

Our advice is that the experiments should focus on estimating the error at an aggregate level at which the ecological question is posed, typically analyzing samples for a target region. This, of course, requires datasets composed of several samples taken for such region. Recall that most often, the ecological unit of analysis is the sample and therefore the classification accuracy at an individual level should be somewhat secondary.

Datasets: representing the underlying probability distribution

One factor that has a major influence on the validation process is the way in which datasets are constructed. Learning theory establishes that training (and validation) datasets must be generated independently and identically according to the probability distribution, $P(x, y)$, on $\mathcal{X}\times\mathcal{Y}$. This is the so-called independent and identically distributed (i.i.d.) assumption, which is the main assumption made for the learning processes of most algorithms (Duda et al. 2012). When this assumption is not fulfilled, the model obtained is suboptimal with respect to the true underlying distribution, $P(x, y)$, and the performance function optimized by the algorithm (for instance, accuracy). Unfortunately, the datasets used in many plankton studies are biased. Several authors *design* their training sets, selecting *ideal* examples or fixing the number of examples manually for each class in an attempt to improve the overall accuracy, especially when some morphological groups are scarce. This is a clear case of *sampling bias* and the training set does not represent the underlying probability distribution.

Although these kinds of databases are sometimes only used for training the models, which are subsequently validated using a different testing set, sampling bias is still dangerous for the training process. Learning a model is in fact a searching process in which the algorithm selects the best model from a model space according to: (1) the training dataset, which is the representation of the probability distribution, and (2) a target performance measure, including some regularization mechanism to avoid overfitting. If the training set is biased, then the learning process is ill-posed. Furthermore, if the same dataset is also employed in the validation phase (for instance, when a cross-validation is performed), then the estimate of the error obtained is clearly biased. Thus, the first thing to bear in mind is the golden rule of building an unbiased dataset.

A frequent practice in plankton recognition studies is to look for the *best* training dataset. This is partly motivated by the scarcity of labeled examples and imbalanced classes; there are groups that have much fewer examples than others. Researchers usually build training datasets with the same number of individuals for each class to avoid this issue. This is a bad practice.

It is true that imbalanced situations can make some classifiers misclassify the examples of the minority classes. Nonetheless, the solution from a formal point of view is not to select the examples for the training dataset manually, thereby biasing the sampling process. The correct procedure is just the opposite. In supervised learning, the training data comes first. It is the most important element and should be obtained obeying the i.i.d. assumption as far as possible, without introducing sampling bias of any kind. Then, we may work with three elements to boost the performance of our model: (1) enhancing the representation of the input objects (e.g., using advanced computer vision techniques robust to rotation or obstruction), (2) selecting a classifier well tailored to the characteristics of the training data and the learning task (e.g., using algorithms for imbalanced data (Chawla et al. 2004) if required), and (3) tuning the parameters of the learning algorithm (e.g., algorithms usually have a regularization parameter to avoid overfitting, like parameter C in the case of Support Vector Machines).

Selecting the training dataset manually (Culverhouse et al. 1996; Luo et al. 2003; Grosjean et al. 2004; Hu and Davis 2005) is counterproductive for a number of reasons. Balancing the number of training examples for all classes may mean that a large class does not have sufficient diversity, for instance, when such class is complex and it has different types of individuals. Limiting the number of examples for such classes reduces the desired diversity of the training data. The i.i.d. assumption guarantees that, if the sample is large enough, all the individuals will be represented in the training set, making the learning process more reliable.

The previous argument is supported by statistical learning theory. Over the past few years, learning theory papers have established generalization error bounds for different classifiers, including Support Vector Machines (SVM) and ensemble methods like Boosting (Bartlett and Shawe-Taylor 1999; Schapire and Singer 1999; Cristianini and Shawe-Taylor 2000; Vapnik and Chapelle 2000). These bounds decrease (i.e., the probability of error is lower) when the number of examples in the training set increases, among other factors. This is a quite intuitive result; when the model has been trained with more information (examples), its ability to classify unseen examples is greater. Hence, if the total number of examples is reduced in order to balance all classes, the risk of the generalization error of the classifier increases.

Basically, supervised learning requires a large collection of examples, one that is as large as possible, sampled without bias from the underlying population. This in turn guarantees the required diversity of the examples. Note that diversity in this context refers to the different types of objects that the model has to work with. This includes not only the different types of individuals from a biological or morphological point of view, but also the diversity produced by the capturing device or any other element of the processing system that may mean that the same type of individual is represented differently. This general principle has the drawback that obtaining a large collection of examples is usually expensive. Our case is even worse, because, if the unit is the sample and not the individual, a large collection of diverse samples is required. The problem is that obtaining sufficient diversity at the sample level is difficult, but makes diversity at an individual level less problematic.

Data distribution drift

Understanding data distribution drift is important to obtain a better solution to the plankton recognition problem. Formally, this occurs when the joint distribution of inputs and outputs changes. Given two datasets, D and T, captured at different times or places, drift occurs when their joint probability distributions differ; in symbols, $P_D(x,y) \neq P_T(x,y)$. Several factors can be the cause of this drift and the joint probability can be expressed in different ways depending on the type of learning problem. Fawcett and Flach (2005) proposed a taxonomy to classify learning problems according to the causal relationship between class labels and covariates (or inputs). The interest of this taxonomy lies in the fact that it determines the kind of changes in the distribution that a particular task may experience. The authors distinguished between two different kinds of problems: $\mathcal{X} \rightarrow \mathcal{Y}$ problems, in which the class label is causally determined by the values of the inputs; and $\mathcal{Y} \rightarrow \mathcal{X}$ problems, where the class label causally determines the covariates. Spam detection constitutes an example of the first type of problem; the content of the mail and other characteristics determine whether the mail is spam or not. On the other hand, a medical diagnosis task is a typical example of $\mathcal{Y} \rightarrow \mathcal{X}$ problems; suffering from a particular disease, y, causes a series of symptoms, x, to appear, and not the other way around. Plankton recognition is a $\mathcal{Y} \rightarrow \mathcal{X}$ problem. An individual will have some characteristics because it belongs to a particular species or morphological group. These characteristics are a consequence of its class.

In this type of problem, the joint distribution, $P(x, y)$, can be written as $P(x,y)=P(x|y)P(y)$, in which $P(y)$ represents the probability of a class and $P(x|y)$ is the probability of an object x, although knowing that the class is y. We know that $P(x, y)$ changes, so we need to determine whether both terms in the expression change or just one of them.

In abundance related problems, it is evident that $P(y)$ changes, because it is precisely the magnitude that must be estimated for the model. However, does $P(x|y)$ change? The answer to this question is more complex. Lets imagine that we represent each individual using only one characteristic: the particle physical size. If $P(x|y)$ remains constant, it means that the distribution of sizes in each class does not change. Notice that this is a quite strong condition that depends on several factors, basically the representation of the input space, the taxonomy and the classes considered in each particular plankton recognition problem. If we only have a small number of top-level classes, it is almost certain that $P(x|y)$ changes as these classes are formed by different subclasses, whose probabilities will not change in proportion to the main class. Conversely, if the problem distinguishes between classes at the bottom of a taxonomy, then $P(x|y)$ changes are less probable.

Knowledge of all these factors for a given problem, mainly the behavior of $P(x|y)$, is crucial in order to design new algorithms that are robust against the expected changes in the joint probability distribution. For instance, the algorithm proposed by Solow et al. (2001) is based on the assumption that $P(x|y)$ is constant. This is also the main assumption made by several quantification algorithms (Forman 2008).

A way to measure changes in the distribution between two datasets is the Hellinger distance (HD). This measure has been used in classification methods to detect failures in classifier performance due to shifts in data distribution (Cieslak and Chawla 2009). In this paper, HD will be used to study the dataset shift between training and test datasets and how this relates to the accuracy of the classifier and the corresponding validation methods. The Hellinger distance is a type of f-divergence initially proposed to quantify the similarity between two probability distributions. Based on the continuous case formulation, the Hellinger distance can also be computed for the discrete case. Given two datasets, D and T, from the same input space, $\mathcal{X}$, their HD is calculated as

$$\mathrm{HD}(D,T)=\frac{1}{d}\sum_{f=1}^{d}\mathrm{HD}_f(D,T)=\frac{1}{d}\sum_{f=1}^{d}\sqrt{\sum_{k=1}^{b}\left(\sqrt{\frac{|D_{f,k}|}{|D|}}-\sqrt{\frac{|T_{f,k}|}{|T|}}\right)^2}, \quad (3)$$

in which d is the dimension of the input space (the number of attributes or features), $\mathrm{HD}_f(D,T)$ represents the Hellinger distance for feature f, b is the number of bins used to construct the histograms, $|D|$ is the total number of examples in dataset $|D|$, and $|D_{f,k}|$ is the number of examples whose feature f belongs to the k-th bin (the same definitions apply to dataset T).

Performance measures

In order to compare the performance of several methods over a group of samples, two different types of results can be studied. First, the goal may be to analyze the error for a

particular class across all samples and obtain the error rate. However, it may also be necessary to calculate the precision for all classes across all samples, a kind of general error of the model.

To compute the error rate for a particular class, we need to compare the predicted count data or frequencies, $\{n'_{c_i,j} : j=1,\ldots,m\}$, with the ground-truth count of class c_i over m labeled samples $\{n_{c_i,j} : j=1,\ldots,m\}$. Three performance measures are usually employed in similar regression problems:

- Bias: $Bias(c_i)=\frac{1}{m}\sum_{j=1}^{m} n_{c_i,j}-n'_{c_i,j}$
- Mean Absolute Error: $MAE(c_i)=\frac{1}{m}\sum_{j=1}^{m} |n_{c_i,j}-n'_{c_i,j}|$
- Mean Square Error: $MSE(c_i)=\frac{1}{m}\sum_{j=1}^{m} (n_{c_i,j}-n'_{c_i,j})^2$

The drawback of *Bias* is that negative and positive biases are neutralized. For instance, a method that guesses the same number of units as too high or too low will have zero bias on average. *MAE* and *MSE* are probably the most widely used loss functions in regression problems, although *MAE* is more intuitive and easier to interpret than *MSE*. Nonetheless, all these metrics present some issues in this case. First, if the frequencies are expressed in terms of another variable, typically volume, the density must be similar in order to average the errors across samples, otherwise the samples with a higher density have a greater influence on the final score. More importantly, these metrics do not allow us to determine the magnitude of the errors. Averaging across samples with different frequencies has certain implications that should be carefully taken into account. For instance, an error of 10 units produced when the actual value is 100 is not the same as when the actual value is 20. In the latter case, the error can be considered worse. The problem of these measures in the context of plankton studies is that it is quite commonplace for a given sample not to contain examples for some classes. Any error in theses cases is high in relative terms, even when the absolute error is low. These factors decrease the usefulness of these performance metrics.

There are several measures for evaluating the importance of errors. Mean Absolute Percentage Error, $MAPE=1/m\sum_{i=1}^{m}(|n_{c_i,j}-n'_{c_i,j}|/n_{c_i,j})$, also called Mean Relative Error *MRE*, is probably the most popular. However, this measure presents some issues: it is asymmetric, unbounded and undefined when $n_{c_i,j}=0$. Moreover, recent papers (Tofallis 2015) have shown that *MAPE* prefers those models that systematically under-forecast when it is used in model selection processes. The log of the accuracy ratio, i.e., $\ln(n'_{c_i,j}/n_{c_i,j})$, has been introduced to select less biased models. However, this measure presents the same problem as *MAPE*: it is undefined when $n_{c_i,j}$ is 0 for one sample, which it is quite common when the number of classes is large. Symmetric MAPE (Armstrong 1978) seems the best alternative considering all the above factors:

$$SMAPE(c_i)=\frac{1}{m}\sum_{i=1}^{m}\frac{|n_{c_i,j}-n'_{c_i,j}|}{n_{c_i,j}+n'_{c_i,j}}. \quad (4)$$

It is a percentage, it is always defined and its reliability for model selection purposes is comparable to that of $\ln(n'_{c_i,j}/n_{c_i,j})$ according to Tofallis (2015).

In order to compute a kind of overall result, an initial performance metric that can be applied is Bray–Curtis dissimilarity (Bray and Curtis 1957). This is commonly used to analyze abundance data collected at different sampling locations in ecological studies. The Bray–Curtis dissimilarity is defined as:

$$\mathrm{BC}=1-2\frac{\sum_{i=1}^{l}\min(n_{c_i},n'_{c_i})}{\sum_{i=1}^{l} n_{c_i}+n'_{c_i}}=\frac{\sum_{i=1}^{l}|n_{c_i}-n'_{c_i}|}{\sum_{i=1}^{l} n_{c_i}+n'_{c_i}}, \quad (5)$$

where l is the number of classes. A good thing here is that both samples obviously have the same total size. This means that the score it is the same whether counts or frequencies are used. The Bray–Curtis dissimilarity is bound between 0 and 1, with 0 meaning that the prediction is perfect. Although the Bray–Curtis dissimilarity metric is able to quantify the difference between samples, it is not a true distance because it does not satisfy the triangle inequality axiom.

A possible alternative to the Bray–Curtis dissimilarity is the Kullback–Leibler Divergence, also known as normalized cross-entropy. In this case, it is usual to compare a set of frequencies:

$$\mathrm{KLD}(n,n')=\sum_{i=1}^{l}\frac{n_{c_i}}{\sum_{i=1}^{l} n_{c_i}}\cdot\log\left(\frac{n_{c_i}}{n'_{c_i,j}}\right). \quad (6)$$

The main advantage of KLD is that it may be more suitable for averaging over different test prevalences. However, a drawback of KLD is that it is less interpretable than other measures, such as the Bray–Curtis dissimilarity or *MAE*. Moreover, it is not defined when a frequency is 0 or 1, which is quite common in plankton recognition, particularly when the number of classes is large. In order to resolve these situations, KLD can be normalized via the logistic function: $\mathrm{NKLD}(n,n')=2/(1+\exp(\mathrm{KLD}(n,n')))$.

Assessment methods for a collection of samples

As stated previously, several studies have found it difficult to expose their algorithms to changes in the distribution of plankton populations, reaching the conclusion that

traditional assessment methods significantly overestimate models accuracy. Our goal is to propose an assessment methodology that ensures that training and testing datasets change, introducing the data distribution variations that will occur under real conditions. Moreover, the test should not be carried out only with one test set. Ideally, testing should be carried out with different samples presenting different distributions, covering the actual variations due to seasonal factors or the location of sampling stations as much as possible. Here, we shall discuss how to extend traditional assessment methods, namely hold-out and cross-validation, to the case of working with a group of samples, highlighting both their drawbacks and strengths.

The extension of hold-out is fairly straightforward. As always, we need a training dataset, obtained without sampling bias, that represents the probability distribution of the study. Additionally, a collection of samples must be collected to constitute the testing set. The performance of the model is assessed just in this collection, computing a sample-based measure, like the ones discussed previously.

The drawback of hold-out is that the effort involved in collecting data is doubled because we need two separate datasets. This is much most costly when working with samples. The labeled data is usually limited, so in some studies one of the datasets will be smaller than it should be. If the training dataset is reduced in size, useful information to build the model is lost. If we limit the size of the testing dataset, the assessment of the model will be poor. It seems that shifting the unit of the study from the individual to the sample makes hold-out less suitable.

The other alternative is to apply cross-validation (CV). The difference with respect to traditional CV is that the folds are composed of a number of complete samples. The key parameter in CV is the number of folds. Selecting a low number of folds once again means that the training dataset for each run is smaller, with the same drawbacks as mentioned previously. Thus, the best way to have the maximum amount of training data is to conduct a leave-one-out (LOO) cross-validation of samples (see Fig. 1). Given a set of m samples, m training and test iterations are performed ($E_1 \dots E_m$). In each iteration, all but one sample (the gray sample in each iteration in Fig. 1), is selected as the testing set, performing training with the remaining samples (the white samples in each iteration in Fig. 1).

Notice that this kind of LOO is computationally less expensive than in the case of LOO at the individual level, because the number of individuals is much larger than the number of samples ($n \gg m$); in the case under study, 39,613 examples vs. 60 samples. The other main advantage is that the method operates under similar conditions to real ones when the model is deployed: it has been trained using a group of samples, then it has to make a prediction for a new, unseen sample. It also guarantees a realistic degree of variation between training and test sets. We shall analyze this factor in the experiments.

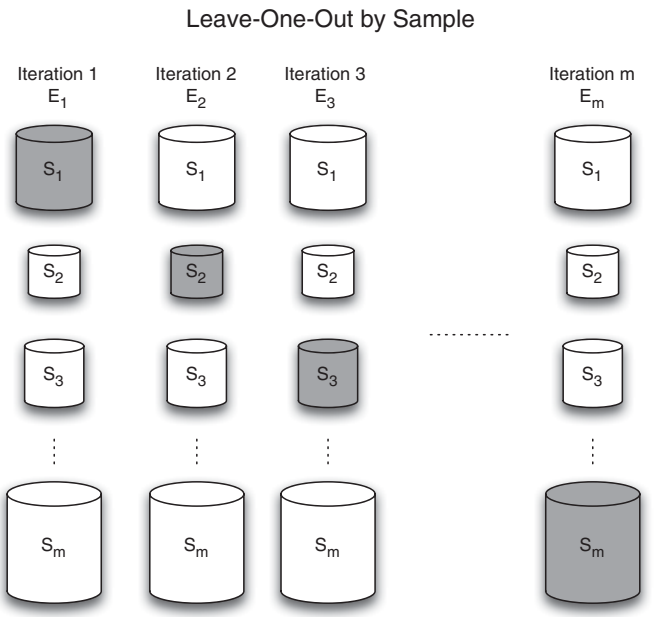

Fig. 1. Given m samples, m training and test iterations are performed ($E_1 \dots E_m$). In each iteration, the gray sample is selected as the testing set, while white samples are used for training. The samples may have different sizes, as represented in the figure.

The advantages of LOO over hold-out are twofold: (1) LOO uses as many training examples as possible, and (2) the estimate of the error is theoretically more precise. However, it also presents an important drawback: it cannot be applied when the samples present some kind of correlation among themselves; for instance, when they come from a series of samples obtained in a short period of time. In such cases, hold-out is the best option: the model is trained with a separate training set and tested on such collection of testing samples. Any sort of cross-validation using this collection of samples will over-estimate the performance of the model. In order to apply cross-validation to a collection of correlated samples, the division in folds must guarantee that those samples correlated among themselves should belong to the same fold. This may, however, be impossible in some cases; for instance, when the size of the fold is just one sample, which is the case of LOO.

Finally, if the collection of samples is large, which is the ideal situation, then instead of using LOO, which can be computationally expensive in such situations, a CV by sample can be applied. The number of folds selected should be as large as possible depending on the computational resources available in order to obtain a more precise estimate. Another important aspect is that, in order to compute the performance measures, the actual samples of the dataset must be considered, without aggregating those that belong to the same fold. Testing samples must not be aggregated

because this procedure will create new artificial samples. The error should be measured for each sample separately and then averaged. For instance, if we have a dataset with 1000 samples, carrying out a LOO by sample will be computationally expensive, as 1000 training and testing operations will be required, each one with a large number of individual examples. This can be reduced to our liking, selecting a number of folds and performing a CV by sample. For instance, with 10-folds, 900 samples will be used for each training process, using the other 100 samples for testing. This process will be carried out 10 times (vs. 1000 iterations of LOO), thus saving in training time. Note that we cannot evaluate error using a test set of 100 samples together, as it would be an artificial sample, suffering the same problems as standard cross validation. Instead, the error should be measured for each test sample separately and then averaged.

Case under study

A relatively large dataset of samples was collected to study the behavior of plankton recognition systems and model assessment methods. Specifically, the images obtained correspond to 60 different samples obtained at different places and different times. This dataset was captured using a FlowCAM (Sieracki et al. 1998) in the Bay of Biscay and off the northern coast of Spain and Portugal between August 2008 and April 2010 (Álvarez et al. 2012). Images were captured using 100X magnification with the aim of analyzing organisms with an equivalent spherical diameter (ESD) between 20 μm and 100 μm. Each of the captured images was segmented using the intensity-based method proposed by Tang et al. (1998). Once segmented, the images were classified by an expert taxonomist into eight categories (Artefacts, Diatoms, Detritus, Sillicoflagellates, Ciliates, Dinoflagellates, Crustaceans and the category Others, for other living objects which could not be classified among the previous categories). A crucial aspect is that all the organisms within each sample were analyzed and labeled without exception to avoid sampling bias.

The sample stations are located at different geographical points and at different depths, as shown in Fig. 2. This results in high variation in the concentration of species, can be seen in Fig. 3, since large regions have been covered in both temporal and spatial terms. For example, the concentration of diatoms in Sample 57 is large (over 75%) compared to Sample 54, in which there are almost no diatoms (less than 1%). More examples like this one can be found in the dataset.

The features vector for each image, x, for each image was calculated using the EBImage R package (Pau et al. 2010). Standard descriptors, including shape and texture features (Haralick et al. 1973), were computed with this package. Furthermore, features computed by the FlowCAM software, such as particle diameter and elongation, were also included

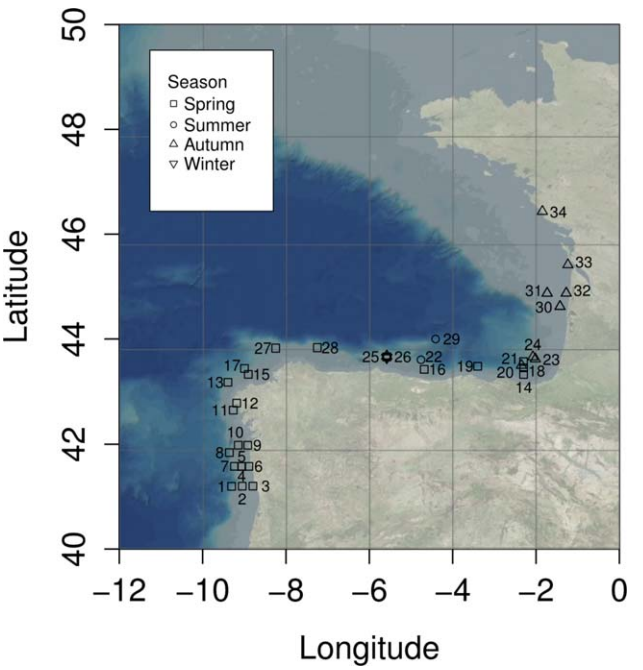

Fig. 2. Geographical location and sampling season. The numbers shown in the figure correspond to the station (see Fig. 3) to allow the identification of the place where each sample was collected).

in the feature vector. In all, a vector with 64 characteristics was computed for each image.

To summarize, a total of $m = 60$ samples were captured and processed, resulting in a total of $n = 39$, 613 images manually labeled in eight different classes.

All experiments were performed using the caret R package (Kuhn 2008). Results were extracted using two different learning algorithms, Support Vector Machines (SVM) (Vapnik and Vapnik 1998) and Random Forest (Breiman 2001), to confirm that the results do not depend on a particular classifier. These classifiers are the most popular in plankton recognition papers. A Gaussian kernel was used to train the SVM models, using a grid search in order to find the best parameters for just the training dataset of each run (regularization parameter C values from 1 to 1×10^3, and sigma values from 1×10^{-6} to 1×10^{-1}). In the case of Random Forest, each model is composed of 500 trees. A grid search was also used to estimate the number of random features selected (values 4,8,16). Tuning parameters is essential in order to avoid overfitting and to obtain better results. This process must be carried out using only the training data in each run of the learning algorithm.

Results

The goal of the experiments was not to build the best classifier or analyze various learning approaches. The experiment was simply designed to compare model assessment

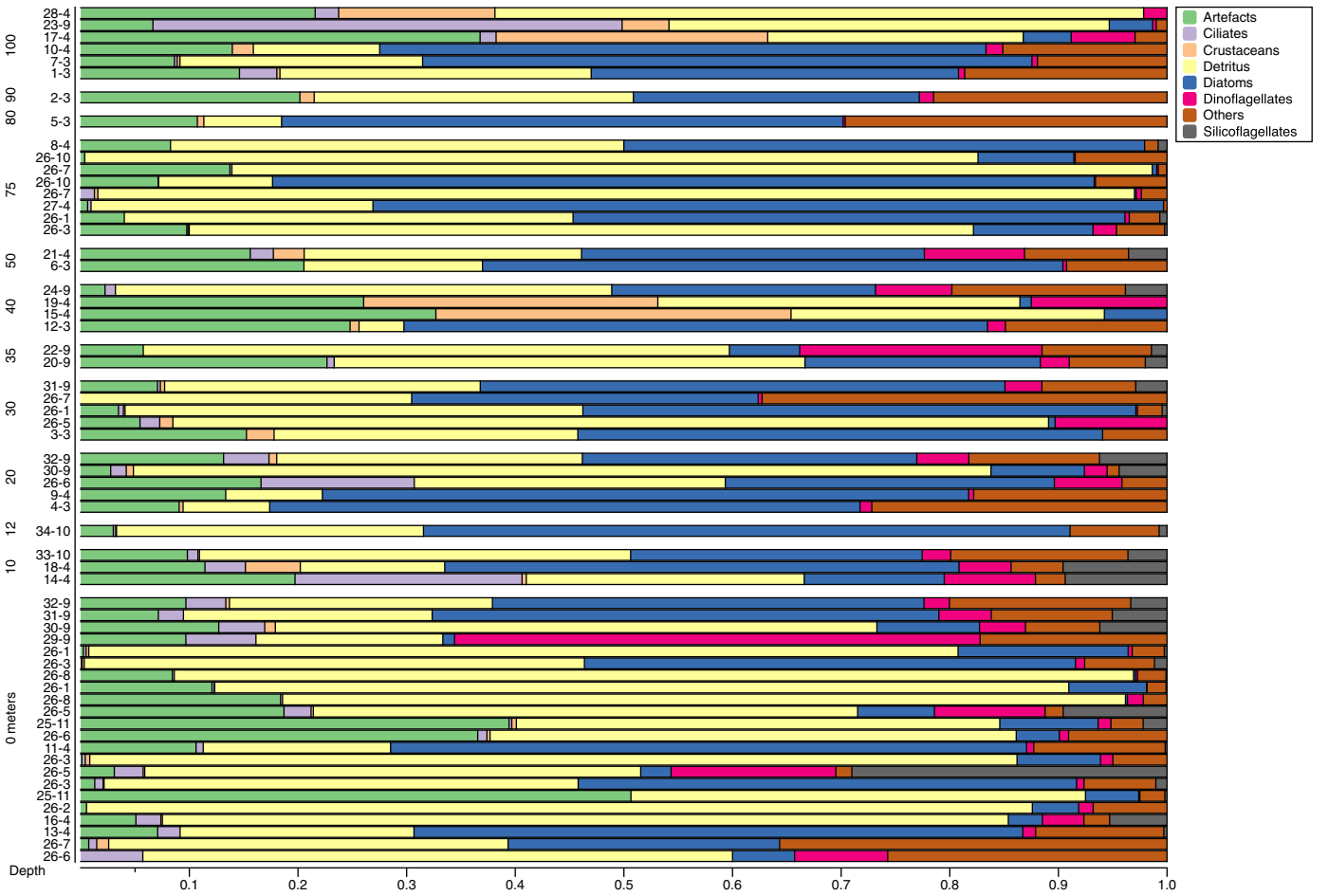

Fig. 3. Distribution of samples by classes. Samples are grouped by depth (in meters) and labeled using the station number and the month in which they were taken.

methods, focusing on the differences between those based on the performance at an individual level and those based on samples. We thus compared standard cross-validation (CV), which works at an individual level, with the proposed leave-one-out (LOO) by sample. Note that in the former case the whole dataset is merged and the samples are not taken into account to obtain the folds. Thus, individual examples from the same sample may belong to different folds. Specifically, we compared three methods: 10-fold CV, 60-fold CV (both working at an individual level) and LOO by sample. We selected 10-fold CV because it is a quite common experimental procedure in many studies (see Table 1) and 60-fold CV to match up the number of samples in the dataset, providing a fair comparison to the experiment using the LOO by sample method.

Table 2 presents the results for both algorithms (SVM and RF) using the three different validation techniques discussed previously. The results for SVM are slightly better, although both algorithms show the same trend. SVM achieves a reasonable degree of accuracy of 82.88% using a standard 10-fold CV. There is no significant difference when the number of folds is increased to 60. Similar results are also obtained with RF using 10-fold CV and 60-fold CV. We can thus conclude that the number of folds has no influence over the obtained estimate. This is mainly due to the fact that the number of examples in the dataset is quite large. Therefore, the probability distributions represented by the training datasets used in each trial are similar because they are large (35,652 examples in a 10-fold CV vs. 38,953 in a 60-fold CV)

Table 2. Accuracy (in percentage) and standard error using different validation methods.

	10 CV	60 CV	LOO by sample	
	Acc	*Acc*	*Acc*	Acc_{sample}
SVM	82.88 ± 0.191	83.10 ± 0.179	77.74	71.78 ± 1.679
RF	82.06 ± 0.180	82.16 ± 0.183	77.05	70.36 ± 1.912

and cross-validation tends to produce similar folds, so the learned models should be approximately equal in each run. In fact, exploring results fold by fold, it was found that the variation in accuracy between folds was small (1.5%/1.4% for 10-fold CV and 6.1%/6.2% for 60-fold CV using SVM/RF, respectively).

However, the accuracy estimate is lower when LOO by sample is used. Notice that we can compute accuracy in two different ways here: (1) summing up the number of correct predictions on each sample and dividing by the total number of examples (this corresponds to the probability of correctly classifying a single unseen instance), and (2) averaging the accuracy per sample (the estimate is the average accuracy of a given unseen sample). Although the scores are different, they are computed from the same individual predictions, the difference arising from the way of averaging the predictions.

In the case of standard CV, both values, Acc_{byfold} and Acc are approximately equal. This is because all the folds have approximately the same size. Being n the number of examples in training set D, NF the number of folds and n_{F_j} the number of examples in fold F_j, we have that:

$$Acc_{byfold} = \frac{1}{NF}\sum_{j=1}^{NF}\frac{1}{n_{F_j}}\sum_{x_i \in F_j} I(h_j(x_i)=y_i) \approx \frac{1}{n}\sum_{j=1}^{NF}\sum_{x_i \in F_j} I(h_j(x_i)=y_i) = Acc, \tag{7}$$

because $\frac{n}{NF} \approx n_{F_j}$ for all j. In contrast, the samples have different sizes in a LOO by sample experiment and hence the two values differ.

Comparing the accuracy estimate at an individual level obtained by means of CV and LOO by sample, the question that has to be answered here is why they are different. First, in our opinion, standard CV is optimistic because, as stated previously, individual examples from the same sample are placed in different folds. Thus, the learner uses examples for training from the same samples as those in the testing set. The estimate is optimistic because these examples are correlated and tend to be similar. This will not occur when the model is deployed and it classifies a new unseen sample, which is in fact the conditions that LOO by sample simulates.

On the other hand, the estimate provide by LOO could be seen as pessimistic because one particular sample used as the test sample may be very different from the rest. This obviously will not occur the same number of times if the training set is composed of a larger collection of samples. Actually, when the number of samples tends to infinity, both methods will return the same estimate (which will be the true accuracy). However, bear in mind that we are estimating the accuracy for the model computed with a limited dataset, not with an infinite number of samples. Hence, in our case, if the test sample is not very well classified in one iteration of LOO using the model learned with the other 59 samples (nearly 40,000 examples), we may infer that the same will occur with other unseen samples when we train our model with the complete 60-sample training set. This is, in fact, the goal of the validation process, making estimates of the future performance. Moreover, these cases show us that the training dataset is possibly not large enough, and that more samples are required.

For all the above reasons, we do think that the estimate computed by means of LOO by sample is more realistic than the one computed by means of standard CV, even though the latter may be somewhat pessimistic, which is better than being optimistic, and it is probably more accurate for a finite collection of samples, which is our goal. Another interesting aspect is that the difference between both measures can serve as an estimator of the completeness of the training set.

On the other hand, the average accuracy at a sample level is especially useful once we have trained our model and we wish to apply it to classify new, unseen single samples. Recall that it measures the expected accuracy when the model only classifies one finite sample and hence it is different to the one previously discussed. First, it is logical for the accuracy in this case to be lower due to size of the samples; the accuracy tends to be lower for smaller samples because any mistake represents a higher percentage. Furthermore, for the same reason, it is always more variable than in the case of large samples. For very large samples, the accuracy at a sample level will tend to be the same as that estimated at an individual level. However, several ecological studies work with relatively small samples.

The second aspect to consider is that the variability in terms of samples can be huge with respect to several features, like size, difficulty and class distribution, among others. We can find small samples and large ones, samples that contain individuals that are particularly difficult to classify and other samples that are composed of easy examples, samples with a different class distribution, etc. Thus, the accuracy can dramatically differ in all of these situations. For instance, in the LOO experiment in Table 2, the accuracy for the worst sample is as low as 30.1% (93 examples); in the contrary case, this value rises to 96.4% for the best sample (2731 individuals). The standard deviation of the estimate is 13.01, showing the great variability in accuracy when different samples are considered. For the sake of comparison, the standard deviation in 10-fold CV is 0.60 and 1.39 in 60-fold CV. The standard deviation of LOO seems excessively high in this experiment, suggesting that we should probably add new samples to the training data to increase the stability of the model. Nonetheless, it is far more realistic than the one provided by standard CV.

In conclusion, we need as large a collection as possible of actual samples in order to estimate the accuracy, or any other magnitude, at a sample level. This is the reason for proposing LOO by sample. Note that these measures cannot be computed using traditional CV because the folds that CV generates: (1) are artificial samples that do not represent

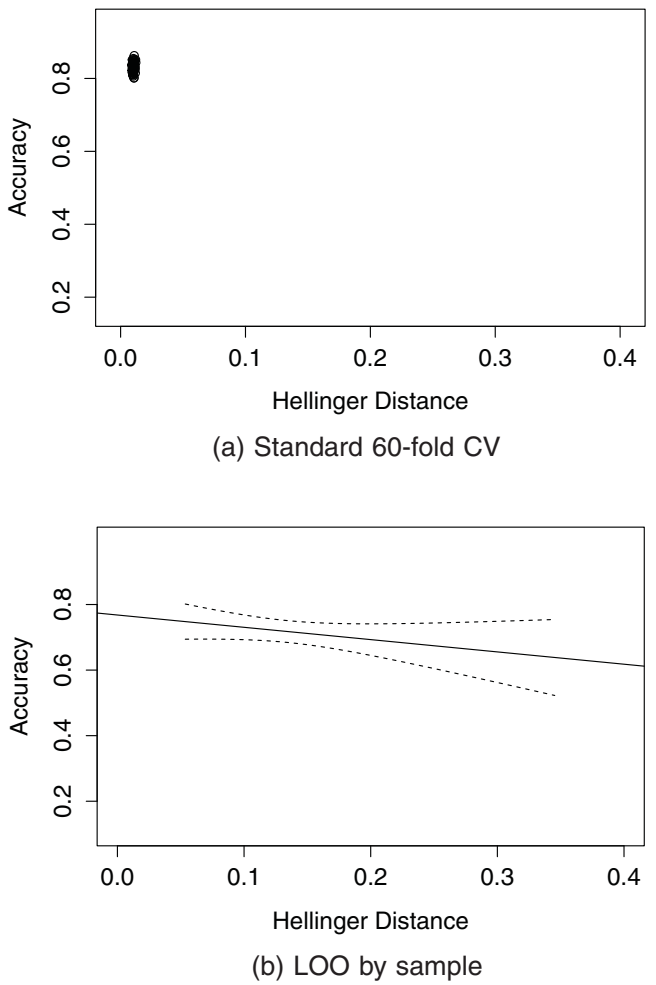

Fig. 4. Relationship between HD and accuracy for each fold/sample using SVM as the classifier ($R^2 = 0.0341$, p-value $= 0.1577$ and $S = 0.1289$ for the LOO experiment). Dotted lines: 95% confidence interval. A similar graph is obtained for the Random Forest classifier. (**a**) Standard 60-fold CV, (**b**) LOO by sample.

actual ones, and (2) they do not have the required variability. In fact, the folds of a CV are a sort of average of the underlying population. Figure 4 shows this feature, comparing the training and testing distributions of both experiments. The figure shows the Hellinger Distance (HD) of both sets, computed used 30 bins, and the corresponding accuracy estimate of the sample/fold. Both distributions are approximately equal in the case of CV (Fig. 4a). It is worth noting that the accuracy and HDs remain practically constant throughout each of the folds. In contrast, the distributions of the training and testing sets differ when LOO by sample is applied; the HDs are significantly different and the accuracy between samples also varies notably (see Fig. 4b). Notice that the minimum HD in the LOO experiment is greater than all the HDs for the CV experiment.

The box plot in Fig. 5 shows the accuracies of LOO by sample and 60-fold CV using SVM for the different classes across the iterations of the experiment. Samples or folds with less than 10 examples for a given class were omitted (three for Crustaceans). This filter was applied to exclude situations that do not produce representative results. For instance, if there is only a single example of a class in a given test set and the classifier fails to recognize it, it will yield an accuracy of 0% for that particular iteration and class, when in fact the classifier has only failed to classify one example, which is insignificant.

Analyzing the box plot in detail, major differences can be seen in classification stability for some classes, especially Detritus and Diatoms. When using standard CV (Fig. 5b), the success rate by class once again remains much more constant throughout all folds because their variability is small. The classes with more variability are those with a limited number of examples per fold (Ciliates, 10.3 examples; Crustaceans, 3.2; Dinoflagellates, 12.6; and Silicoflagellates, 12.3). Classes with a large number of examples (Detritus and Diatoms) do not show any variability. It is thus impossible to study the robustness and stability of the model for each class. In contrast, when LOO by sample is used (Fig. 5a), the classifier accuracy for each class in different iterations tends to be more variable, allowing researchers to analyze these cases in order to improve their models. Once more, this is because these measures are estimates obtained at a sample level, in which LOO by sample is a more appropriate validation method.

The second part of these experiments is devoted to analyzing the behavior of the proposed performance metrics to estimate the abundance. Our purpose is to show a comparison between two algorithms, in this case SVM and Random Forest. Two performance measures are considered: *SMAPE* and Bray–Curtis dissimilarity. The former is applied to study the precision for each class individually and the latter to obtain a global measure. In all cases, we use the predictions obtained in the LOO by sample experiments. Table 3 contains the results for both SVM and RF.

Analyzing the SMAPE results, the errors are excessively high, except for Detritus and Diatoms. Comparing SVM and RF, the scores obtained by SVM are better than those of RF for most classes, except for Others and Silicoflagellates. This is also confirmed by the Bray–Curtis dissimilarity value, which is lower in the case of SVM. These results seem to suggest that the better the accuracy of a model, the better the estimates at an aggregated level.

Figure 6 shows the relationship between accuracy and Bray–Curtis dissimilarity when SVM and RF are used in a LOO by sample experiment. Each point represents both scores for a sample. In both cases, the correlation between the two measures are lower than expected, confirming that better accuracy at an individual level does not mean better performance when an aggregated magnitude is predicted. For instance, in sample 31 the accuracy is just 0.62 but BC score is relatively low, 0.09, while sample 39 has a much

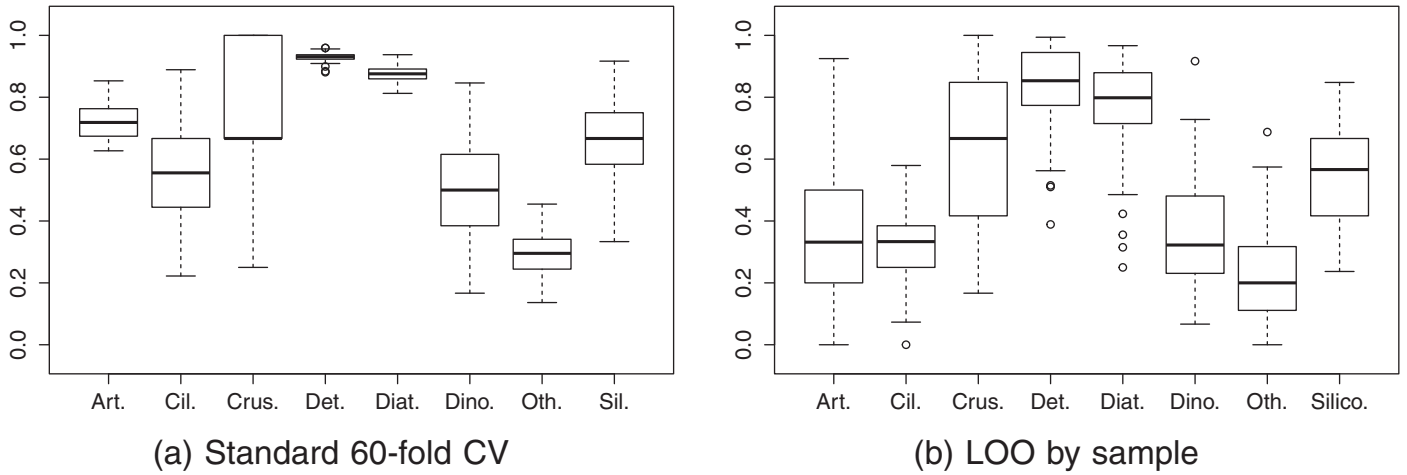

Fig. 5. Accuracy by class and iteration/fold using SVM as the classifier. Only classes with 10 (three for the Crustaceans class) or more examples in the iteration/fold are represented. The number of these cases for LOO are: Artefacts (46), Ciliates (13), Crustaceans (19), Detritus (59), Diatoms (49), Dinoflagellates (24), Others (49), and Silicoflagellates (18). For CV, there are always 60 values. (**a**) Standard 60-fold CV, (**b**) LOO by sample.

Table 3. SMAPE scores and Bray–Curtis dissimilarity in the LOO by sample experiment.

SMAPE scores	SVM	Random Forest
Artefacts	0.41 ± 0.03	0.53 ± 0.04
Ciliates	0.47 ± 0.05	0.63 ± 0.05
Crustaceans	0.32 ± 0.05	0.33 ± 0.05
Detritus	0.10 ± 0.01	0.12 ± 0.01
Diatoms	0.19 ± 0.03	0.25 ± 0.03
Dinoflagellates	0.41 ± 0.04	0.57 ± 0.05
Others	0.45 ± 0.04	0.42 ± 0.04
Silicoflagellates	0.38 ± 0.05	0.38 ± 0.05
	SVM	Random Forest
Bray–Curtis	0.15 ± 0.01	0.19 ± 0.01

better accuracy, 0.83, but a higher BC dissimilarity, 0.13. In fact, the two problems are different from a learning point of view, as discussed previously, and the optimal model for one of them is not optimal for the other, except in the trivial case of obtaining a perfect classifier, which is unrealistic.

Discussion

We can distinguish between two major groups of studies in which image classification tools are useful. Abundance is a primary ecological currency and many studies require automatic methods able to predict the abundance of the different planktonic groups for samples collected in plankton surveys. However, the aim of other studies is to understand properties of the plankton community in addition to abundance (for example, calculating the size structure composition of each classification category) and hence require precise classification of each individual image. Although both cases seem the same learning problem (both classify plankton images), these two types of applications likely require different learning algorithms and surely call for different model assessment and validation methods. The most important difference between both types of studies is that, in the former the aim is to minimize the error per sample, while in the latter, the learning algorithm should also seek to minimize the error for each individual image. This paper focuses mainly on the analysis of the validation techniques required for those studies that require predictions for complete samples.

It is important to stress the great variability found in the performance rates depending on the classified sample. A great disparity in results is also observed in intra-class accuracy. In this respect, the proposed methodology can show us aspects of the capabilities of the models that would remain hidden using other validation strategies. One of these aspects is the significant variability in the results found for certain classes (e.g., diatoms and ciliates). In difficult problems like the one addressed in this paper, it is important to try to look beyond the overall accuracy rate. Very useful information can thus be found which may be valuable in order to build better automatic recognition systems.

One of the most interesting features of performing sample-oriented experiments, like LOO by sample, is that it helps researchers to draw conclusions about the adequacy of the whole learning process. High variations in performance between samples, or for certain classes, may reflect a need to increase the size of the dataset, adding new labeled samples. Eventually, the system may face a new sample, that contains examples that seldom appear in the rest of the training dataset, causing a high error rate for that sample and class and hence high variability in the results. Adding more samples will make the results and the system more robust and more stable.

An illustrative example can be seen in Fig. 5a. There is at least one sample for which the hit rate is 0 for the class

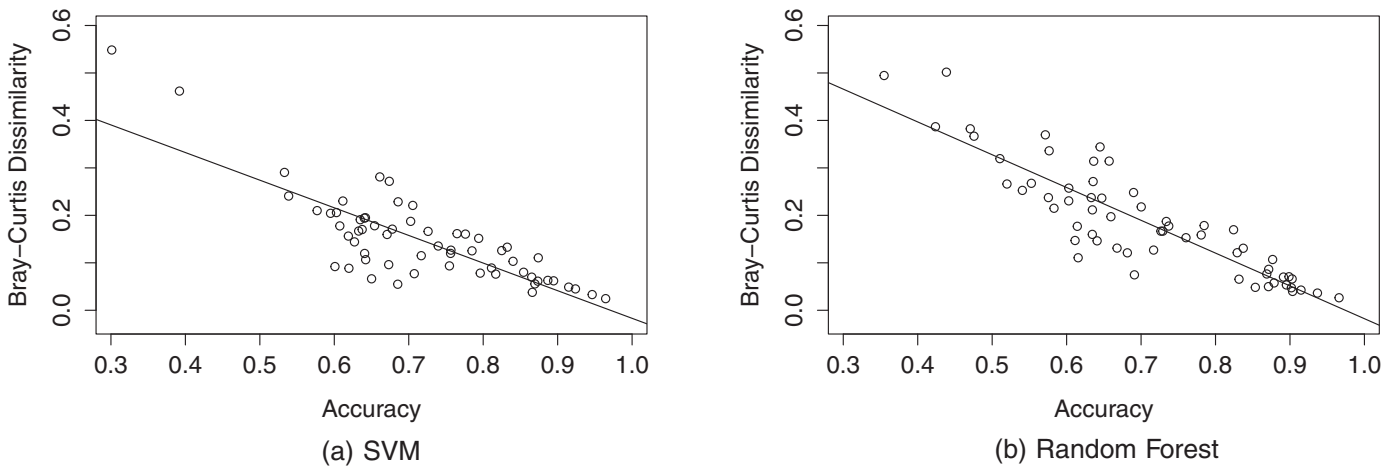

Fig. 6. Relationship between Accuracy and Bray–Curtis Dissimilarity in LOO experiments ($R^2 = 0.6510$, p-value $= 0$, $S = 0.0560$ for SVM and $R^2 = 0.7798$, p-value $= 0$, $S = 0.0548$ for RF). (**a**) SVM, (**b**) Random Forest.

Ciliates. The sample in question is Sample 13, which has 140 examples labeled as Ciliates. All of these examples are misclassified by the classifier when LOO is applied. Investigating more deeply, it turned out that this group of examples was actually a subspecies of ciliates, oligotrichs. There are only 142 examples of this subspecies in our dataset, 140 of which are in Sample 13. Obviously, when excluding Sample 13 from the training set, the classifier does not have enough information to learn how to classify this subtype. Such situations cannot be detected using standard CV during the experiments, but will occur once the model is deployed. This is another important reason why the validation strategy should cover such cases, in order to detect them and, if necessary, improve both the dataset and the classification algorithm. It may be considered that sufficient samples are taken when LOO results are good enough not only in terms of overall accuracy, but also with respect to other aspects, like the variability in inter-sample and intra-class performance. The ultimate goal is to obtain a more robust final model and its corresponding accurate performance estimate.

An interesting open question is whether we can somehow anticipate the reliability of the prediction for a new sample. In this respect, Fig. 4 seems to suggest that when a sample is far from the training set in terms of Hellinger distance, any prediction made is less reliable. This is partially true, although there are other factors that also exert an influence. The most important is the difficulty in classifying the instances of the sample: classifiers make most mistakes in those examples near the frontiers between classes. Actually, to detect whether the sample is strange, we could compute the minimum distance between the sample and all of those in the training set. A large distance implies that the new sample is so different to the samples in the training set, thus making the prediction less reliable.

The main drawback of the methodology proposed here is that it requires a large collection of samples to be absolutely precise. In some studies, this is impossible due to the cost of labeling individual examples. A possible alternative in these situations is to generate artificial, yet biologically plausible samples. This technique is used in quantification learning and is based on the fact that we are dealing with a $\mathcal{Y} \rightarrow \mathcal{X}$ problem and the class causally determines the values of the inputs. We know that $P(y)$ changes in abundance-related problems, and in some studies we can make the further assumption that $P(x|y)$ remains constant. Given an actual sample, we can generate a new artificial sample following these two steps: (1) varying the proportions of the classes of the original sample, generating random values for $P(y)$ possibly using predefined thresholds, and (2) performing a random sampling with replacement (to ensure that $P(x|y)$ does not change) in the original sample, until the number of examples required for each class is obtained. This process has to be carried out using knowledge about the actual study in order to generate plausible samples with the expected distribution of classes, $P(y)$. This allows us to study whether the model is able to correctly predict the abundance for a wide range of expected distributions. In some problems, the assumption that $P(x|y)$ remains constant is too strong; for instance, in the case under study, in which only top-level classes are considered. However, in problems with a large taxonomy, the assumption is probably true for the classes at the bottom of the taxonomy and the procedure could be applicable.

Conclusions

This paper analyzes different validation techniques used in plankton recognition problems, comparing the common

methods used in the field. Although studies apply different approaches, most present similar issues. Results can be very different when different validation strategies are employed, leading to results which are not directly comparable. Even more importantly, when they are used in "production," the models learned are likely to provide less satisfactory results than those estimated in the experimental phase applying traditional model assessment methods. The reason is that these techniques, such as standard cross-validation, are devised for other kinds of learning tasks.

After discussing the shortcomings of these validation strategies for those problems in which the goal is to predict a magnitude given new samples, we propose to change the basic unit of these studies, using the sample as the basic unit. In keeping with this idea, the present paper proposes an extension of the well-known leave-one-out method as a good alternative to obtain accurate estimates at a sample level. The method is able to estimate classifier performance more realistically, taking into account the variety of samples the classifier will face. Using this model assessment method and applying the Hellinger distance, it has been found that the difference between the training and test sets exerts a certain influence over model performance.

Another important conclusion is that it is necessary to focus efforts on designing new learning algorithms which are more robust to the differences between training and test sets. This does not mean to increasing the overall classifier accuracy (which already may be high enough), but making the methods more robust to the changes that occur under real world conditions. These algorithms could be applied in domains like the one studied here, in which, due to a variety factors, the data used to train the model does not accurately represent the final data it will predict. From the point of view of machine learning researchers, this validation strategy allows them to test whether their ideas and the algorithms they have developed to address plankton classification problems work well when there are changes in data distribution.

In the era of Big Data, in which large collections of data are obtained for different applications, plankton recognition also needs to build large datasets for different types of analytic studies. In this respect, using software and hardware tools that allow taxonomists to classify instances quickly can help to obtain these datasets, thereby reducing costs. Ultimately, machine learning requires data, particularly for difficult learning problems like plankton recognition. Lack of data leads to poor models, to poor model assessments or, often, to both.

References

Álvarez, E., Á. López-Urrutia, and E. Nogueira. 2012. Improvement of plankton biovolume estimates derived from image-based automatic sampling devices: Application to FlowCAM. J. Plankton Res. **34**: 454–469. doi:10.1093/plankt/fbs017

Armstrong, J. S. 1978. Long-range forecasting: From crystal ball to computer. Wiley.

Barranquero, J., P. González, J. Díez, and J. J. del Coz. 2013. On the study of nearest neighbour algorithms for prevalence estimation in binary problems. Pattern Recognit. **46**: 472–482. doi:10.1016/j.patcog.2012.07.022

Barranquero, J., J. Díez, and J. J. del Coz. 2015. Quantification-oriented learning based on reliable classifiers. Pattern Recognit. **48**: 591–604. doi:10.1016/j.patcog.2014.07.032

Bartlett, P., and J. Shawe-Taylor. 1999. Generalization performance of support vector machines and other pattern classifiers, p. 43–54. In B. Schölkopf, C. J. C. Burges and A. J. Smola [eds.], Advances in kernel methods—support vector learning. The MIT Press, Cambridge, Massachusetts.

Beaufort, L., and D. Dollfus. 2004. Automatic recognition of coccoliths by dynamical neural networks. Mar. Micropaleontol. **51**: 57–73. doi:10.1016/j.marmicro.2003.09.003

Bell, J. L., and R. R. Hopcroft. 2008. Assessment of ZooImage as a tool for the classification of zooplankton. J. Plankton Res. **30**: 1351–1367. doi:10.1093/plankt/fbn092

Benfield, M., and others. 2007. RAPID: Research on automated plankton identification. Oceanography **20**: 172–187. doi:10.5670/oceanog.2007.63

Blaschko, M., G. Holness, M. Mattar, D. Lisin, P. Utgoff, A. Hanson, H. Schultz, and E. Riseman. 2005. Automatic in situ identification of plankton, v. 1, p. 79–86. *In* IEEE Workshop on Application of Computer Vision, Breckenridge, Colorado.

Boddy, L., C. W. Morris, M. F. Wilkins, G. A. Tarran, and P. H. Burkill. 1994. Neural network analysis of flow cytometric data for 40 marine phytoplankton species. Cytometry **15**: 283–293. doi:10.1002/cyto.990150403

Boddy, L., C. W. Morris, M. F. Wilkins, L. AlHaddad, G. A. Tarran, R. R. Jonker, and P. H. Burkill. 2000. Identification of 72 phytoplankton species by radial basis function neural network analysis of flow cytometric data. Mar. Ecol. Prog. Ser. **195**: 47–59. doi:10.3354/meps195047

Bray, J. R., and J. T. Curtis. 1957. An ordination of the upland forest communities of southern Wisconsin. Ecol. Monogr. **27**: 325–349. doi:10.2307/1942268

Breiman, L. 2001. Random forests. Mach. Learn. **45**: 5–32. doi:10.1023/A:1010933404324

Chawla, N. V., N. Japkowicz, and A. Kotcz. 2004. Editorial: Special issue on learning from imbalanced data sets. SIGKDD Explor. Newsl. **6**: 1–6. doi:10.1145/1007730.1007733

Cieslak, D. A., and N. V. Chawla. 2009. A framework for monitoring classifiers' performance: When and why failure occurs? Knowl. Inf. Syst. **18**: 83–108. doi:10.1007/s10115-008-0139-1

Cristianini, N., and J. Shawe-Taylor. 2000. An introduction to support vector machines and other kernel-based learning methods. Cambridge Univ. Press.

Culverhouse, P. F., and others. 1996. Automatic classification of field-collected dinoflagellates by artificial neural

network. Mar. Ecol. Prog. Ser. **139**: 281–287. doi:10.3354/meps139281

Dai, J., R. Wang, H. Zheng, G. Ji, and X. Qiao. 2016. ZooplanktoNet: Deep convolutional network for zooplankton classification, p. 1–6. *In* OCEANS 2016-Shanghai. IEEE, Shanghai, China.

Davis, C. S., Q. Hu, S. M. Gallager, X. Tang, and C. J. Ashjian. 2004. Real-time observation of taxa-specific plankton distributions: An optical sampling method. Mar. Ecol. Prog. Ser. **284**: 77–96. doi:10.3354/meps284077

Duda, R. O., P. E. Hart, and D. G. Stork. 2012. Pattern classification. John Wiley & Sons.

Ellen, J., H. Li, and M. D. Ohman. 2015. Quantifying California current plankton samples with efficient machine learning techniques, p. 1–9. *In* OCEANS 2015-MTS/IEEE Washington. IEEE, Washington D.C., USA.

Embleton, K. V., C. E. Gibson, and S. I. Heaney. 2003. Automated counting of phytoplankton by pattern recognition: A comparison with a manual counting method. J. Plankton Res. **25**: 669–681. doi:10.1093/plankt/25.6.669

Faillettaz, R., M. Picheral, J. Y. Luo, C. Guigand, R. K. Cowen, and J. O. Irisson. 2016. Imperfect automatic image classification successfully describes plankton distribution patterns. Methods Oceanogr. **15–16**: 60–77. doi:10.1016/j.mio.2016.04.003

Fawcett, T., and P. Flach. 2005. A response to Webb and Ting's on the application of ROC analysis to predict classification performance under varying class distributions. Mach. Learn. **58**: 33–38. doi:10.1007/s10994-005-5256-4

Forman, G. 2008. Quantifying counts and costs via classification. Data Min. Knowl. Discov. **17**: 164–206. doi:10.1007/s10618-008-0097-y

Frankel, D. S., S. L. Frankel, B. J. Binder, and R. F. Vogt. 1996. Application of neural networks to flow cytometry data analysis and real-time cell classification. Cytometry **23**: 290–302. doi:10.1002/(SICI)1097-0320(19960401)23:4<290::AID-CYTO5>3.0.CO;2-L

Gislason, A., and T. Silva. 2009. Comparison between automated analysis of zooplankton using ZooImage and traditional methodology. J. Plankton Res. **31**: 1505–1516. doi:10.1093/plankt/fbp094

González, P., E. Alvarez, J. Barranquero, J. Díez, R. Gonzalez-Quiros, E. Nogueira, A. Lopez-Urrutia, and J. J. del Coz. 2013. Multiclass support vector machines with example-dependent costs applied to plankton biomass estimation. IEEE Trans. Neural Netw. Learn. Syst. **24**: 1901–1905. doi:10.1109/TNNLS.2013.2271535

Gorsky, G., P. Guilbert, and E. Valenta. 1989. The Autonomous Image Analyzer–enumeration, measurement and identification of marine phytoplankton. Mar. Ecol. Prog. Ser. **58**: 133–142. doi:10.3354/meps058133

Gorsky, G., and others. 2010. Digital zooplankton image analysis using the ZooScan integrated system. J. Plankton Res. **32**: 285–303. doi:10.1093/plankt/fbp124

Grosjean, P., M. Picheral, C. Warembourg, and G. Gorsky. 2004. Enumeration, measurement, and identification of net zooplankton samples using the ZOOSCAN digital imaging system. ICES J. Mar. Sci. **61**: 518–525. doi:10.1016/j.icesjms.2004.03.012

Haralick, R. M., K. Shanmugam, and I. H. Dinstein. 1973. Textural features for image classification. IEEE Trans. Syst. Man Cybern. **6**: 610–621. doi:10.1109/TSMC.1973.4309314

Haury, L., J. McGowan, and P. Wiebe. 1978. Patterns and processes in the time-space scales of plankton distributions, p. 277–327. In J. H. Steele [ed.], Spatial pattern in plankton communities. Woods Hole Oceanographic Institution, Woods Hole, Massachusetts.

Hu, Q., and C. Davis. 2005. Automatic plankton image recognition with co-occurrence matrices and support vector machine. Mar. Ecol. Prog. Ser. **295**: 21–31. doi:10.3354/meps295021

Hu, Q., and C. Davis. 2006. Accurate automatic quantification of taxa-specific plankton abundance using dual classification with correction. Mar. Ecol. Prog. Ser. **306**: 51–61. doi:10.3354/meps306051

Jeffries, H. P., M. S. Berman, A. D. Poularikas, C. Katsinis, I. Melas, K. Sherman, and L. Bivins. 1984. Automated sizing, counting and identification of zooplankton by pattern recognition. Mar. Biol. **78**: 329–334. doi:10.1007/BF00393019

Kuhn, M. 2008. Building predictive models in R using the caret package. J. Stat. Softw. **28**: 1–26. doi:10.18637/jss.v028.i05

Lindgren, J. F., I. M. Hassellöv, and I. Dahllöf. 2013. Analyzing changes in sediment meiofauna communities using the image analysis software ZooImage. J. Exp. Mar. Biol. Ecol. **440**: 74–80. doi:10.1016/j.jembe.2012.12.001

Lisin, D., M. Mattar, M. Blaschko, E. Learned-Miller, and M. Benfield. 2005. Combining local and global image features for object class recognition, p. 47–47. *In* CVPR Workshops, San Diego, CA, USA.

Luo, T., K. Kramer, D. Goldgof, L. Hall, S. Samson, A. Remsen, and T. Hopkins. 2003. Learning to recognize plankton, v. 1, p. 888–893. *In* IEEE International Conference on Systems, Man, and Cybernetics, Washington, D.C., USA.

Luo, T., K. Kramer, D. Goldgof, L. Hall, S. Samson, A. Remsen, and T. Hopkins. 2004. Recognizing plankton images from the shadow image particle profiling evaluation recorder. IEEE Trans. Syst. Man Cybern. Part B Cybern. **34**: 1753–1762. doi:10.1109/TSMCB.2004.830340

Luo, T., K. Kramer, D. B. Goldgof, L. O. Hall, S. Samson, A. Remsen, and T. Hopkins. 2005. Active learning to recognize multiple types of plankton. J. Mach. Learn. Res. **6**: 589–613. http://www.crossref.org/jmlr_DOI.html

Moreno-Torres, J. G., T. Raeder, R. Alaiz-RodríGuez, N. V. Chawla, and F. Herrera. 2012. A unifying view on dataset shift in classification. Pattern Recognit. **45**: 521–530. doi:10.1016/j.patcog.2011.06.019

Orenstein, E. C., O. Beijbom, E. E. Peacock, and H. M. Sosik. 2015. Whoi-plankton-a large scale fine grained visual recognition benchmark dataset for plankton classification. arXiv Preprint arXiv **1510**: 00745.

Pau, G., F. Fuchs, O. Sklyar, M. Boutros, and W. Huber. 2010. EBImage—an R package for image processing with applications to cellular phenotypes. Bioinformatics **26**: 979–981. doi:10.1093/bioinformatics/btq046

Schapire, R. E., and Y. Singer. 1999. Improved boosting algorithms using confidence-rated predictions. Mach. Learn. **37**: 297–336. doi:10.1023/A:1007614523901

Sieracki, C. K., M. E. Sieracki, and C. S. Yentsch. 1998. An imaging-in-flow system for automated analysis of marine microplankton. Mar. Ecol. Prog. Ser. **168**: 285–296. doi:10.3354/meps168285

Simpson, R., P. Culverhouse, R. Ellis, and B. Williams. 1991. Classification of euceratium Gran. in neural networks, p. 223–229. *In* IEEE Conference on Neural Networks for Ocean Engineering, Washington, D.C., USA.

Solow, A., C. Davis, and Q. Hu. 2001. Estimating the taxonomic composition of a sample when individuals are classified with error. Mar. Ecol. Prog. Ser. **216**: 309–311. doi:10.3354/meps216309

Sosik, H. M., and R. J. Olson. 2007. Automated taxonomic classification of phytoplankton sampled with imaging-in-flow cytometry. Limnol. Oceanogr.: Methods **5**: 204–216. doi:10.4319/lom.2007.5.204

Tang, X., W. K. Stewart, L. Vincent, H. Huang, M. Marra, S. M. Gallager, and C. S. Davis. 1998. Automatic plankton image recognition, p. 177–199. In S. Panigrahi [ed.], Artificial intelligence for biology and agriculture, North Dakota State University, Fargo, USA.

Tang, X., F. Lin, S. Samson, and A. Remsen. 2006. Binary plankton image classification. IEEE J. Oceanic Eng. **31**: 728–735. doi:10.1109/JOE.2004.836995

Tofallis, C. 2015. A better measure of relative prediction accuracy for model selection and model estimation. J. Oper. Res. Soc. **66**: 1352–1362. doi:10.1057/jors.2014.103

Vandromme, P., L. Stemmann, C. Garcìa-Comas, L. Berline, X. Sun, and G. Gorsky. 2012. Assessing biases in computing size spectra of automatically classified zooplankton from imaging systems: A case study with the ZooScan integrated system. Methods Oceanogr. **1**: 3–21. doi:10.1016/j.mio.2012.06.001

Vapnik, V. N., and V. Vapnik. 1998. Statistical learning theory, v. **1**. Wiley.

Vapnik, V., and O. Chapelle. 2000. Bounds on error expectation for support vector machines. Neural Comput. **12**: 2013–2036. doi:10.1162/089976600300015042

Ye, L., C. Y. Chang, and C. Hsieh. 2011. Bayesian model for semi-automated zooplankton classification with predictive confidence and rapid category aggregation. Mar. Ecol. Prog. Ser. **441**: 185–196. doi:10.3354/meps09387

Zhao, F., F. Lin, and H. S. Seah. 2010. Binary SIPPER plankton image classification using random subspace. Neurocomputing **73**: 1853–1860. doi:10.1016/j.neucom.2009.12.033

Acknowledgments

This research has been supported by MINECO (the Spanish Ministerio de Economía y Competitividad) and FEDER (Fondo Europeo de Desarrollo Regional), grant TIN2015-65069-C2-2-R. Juan José del Coz is also supported by the Fulbright Commission and the Salvador de Madariaga Program, grant PRX15/00607.

Conflict of Interest

None declared.

Submitted 23 June 2016
Revised 05 October 2016
Accepted 07 November 2016

Associate editor: Paul Kemp