Robust predictive modelling of water pollution using biomarker data

Marcin Budka[a,1], Bogdan Gabrys[b], Elisa Ravagnan[c]

[a] *mbudka@bournemouth.ac.uk,* [b] *bgabrys@bournemouth.ac.uk,* [c] *elisa.ravagnan@iris.no*
[a,b] *Computational Intelligence Research Group, School of DEC, Bournemouth University, Poole House, Talbot Campus, Fern Barrow, Poole BH12 5BB, United Kingdom*
[c] *International Research Institute of Stavanger, Mekjarvik 12, 4070 Randaberg, Norway*

Abstract

This paper describes the methodology of building a predictive model for the purpose of marine pollution monitoring, based on low quality biomarker data. A step–by–step, systematic data analysis approach is presented, resulting in design of a purely data–driven model, able to accurately discriminate between various coastal water pollution levels.

The environmental scientists often try to blindly apply various machine learning techniques to their data without much success, mostly because of the lack of experience with different methods and required 'under the hood' knowledge. Thus this paper is a result of a collaboration between the machine learning and environmental science communities, not only presenting a predictive model development workflow, but also discussing and addressing potential pitfalls and difficulties.

The novelty of the modelling approach presented in this paper lays in successful application of machine learning techniques to high dimensional, incomplete biomarker data, which to our knowledge has not been done before and is the result of close collaboration between the machine learning and environmental science communities.

Keywords: biomarkers, water quality monitoring, marine pollution, ensemble classification, missing data, predictive modelling

[1] *Corresponding author.* Tel.: +44 1202 524111 ext. 61463, fax: +44 1202 962736.

Robust predictive modelling of water pollution using biomarker data

Marcin Budka[a,1], Bogdan Gabrys[b], Elisa Ravagnan[c]

[a] *mbudka@bournemouth.ac.uk,* [b] *bgabrys@bournemouth.ac.uk,* [c] *elisa.ravagnan@iris.no*
[a,b] *Computational Intelligence Research Group, School of DEC, Bournemouth University, Poole House, Talbot Campus, Fern Barrow, Poole BH12 5BB, United Kingdom*
[c] *International Research Institute of Stavanger, Mekjarvik 12, 4070 Randaberg, Norway*

1. Introduction

Water pollution monitoring becomes a crucial problem as more and more contaminants enter the marine environment every year (Livingstone et al., 2000). The current trend is prediction of the toxicity level using various measurable attributes of the aquatic environment (Pace, 2001). This can be observed by a worldwide increase in the number of water quality research funding opportunities, e.g. by the European Commission[2], the National Research Council in Canada and the USA[3,4] and various local Councils. The data used in this research has been collected as a part of the 'Marine Environment IQ' project[5], which run between 2006 and 2008 and has been funded by the Research Council of Norway[6].

The condition of a marine environment not always can be diagnosed by chemical analysis of the water, as it does not provide any information regarding the health of the organisms. Moreover it may also fail to detect any pollution at all due to its low, yet biologically significant degree or very slow increase of contamination level. The solution to this problem is the use of biomarkers.

For many years biomarkers have been successfully used as a tool of exposure analysis. Their importance results from the fact, that they enable detection of pollutants not possible to achieve by other, commonly used methods like chemical or physical analysis (Ott et al., 2006; Peakall, 1994). Biomarkers are generally classified in two groups: biomarkers of exposure and biomarkers of

[1] *Corresponding author.* Tel.: +44 1202 524111 ext. 61463, fax: +44 1202 962736.

[2] European Commission Research, http://ec.europa.eu/research/index.cfm

[3] National Research Council Canada, http://www.nrc-cnrc.gc.ca/eng/index.html

[4] National Research Council, http://sites.nationalacademies.org/NRC/index.htm

[5] Developing an Index of the Quality of the Marine Environment (Marine Environment IQ) based on biomarkers: Integration of pollutant effects on marine organisms, http://www.iris.no/Internet/NFR-feb2009.nsf/

[6] Research Council of Norway, http://www.forskningsradet.no/

effect. Following (Lam and Gray, 2003), “exposure biomarkers detect biological changes that are indicative of exposure to a specific agent, even if these changes may not be directly linked to harmful (toxic) effects in the target organism, while effect biomarkers reveal biological changes occurring in organisms and caused by contaminants”.

Mussels have been used as sentinel organism from the 70s (Goldberg, 1986; Goldberg and Bertine, 2000). There are multiple advantages using bivalves in environmental monitoring: they are widely distributed and sedentary, they are easy to sample, they tolerate a wide range of environmental conditions, and, most important, bivalves bioconcentrate environmental toxicants because of their high filtration activity.

Over the years, a large number of biomarkers has been developed, related to their potential effect on organisms (Depledge and Fossi, 1994; Depledge et al., 1995; Harvey and Parry, 1997; Regoli et al., 1998; Bresler et al., 2003; Hellou and Law, 2003; Rank and Jensen, 2003; Barsiene et al., 2004; Dahlhoff, 2004; Moore et al., 2004; Yang et al., 2004; Amiard et al., 2006; Bocchetti and Regoli, 2006; Lesser, 2006; Magni et al., 2006; Widdows and Staff, 2006). Although biomarkers play a great role in ecotoxicology and environmental risk assessment, they are sometime difficult to interpret. To determine whether a biomarker response is an indicator of impairment or is a part of the homeostatic response, indicating that an organism is successfully dealing with the exposure, is extremely complex (Forbes et al., 2006). When dealing with mixtures of pollutants, the use of a group of biomarkers (‘battery’) is suggested (Eason and O’Halloran, 2002; Chèvre et al., 2003), combining effect and exposure tests. One of the objectives of this study was to validate the choice of biomarkers made during the ‘Marine Environment IQ’ project.

The collection of biomarker data is a rather involved process, which requires performing a set of usually destructive tests on biological material. Unfortunately, in the majority of the studies it is impossible to use the same animal for the whole battery of tests, because of the quantity of biological material required to perform chemical analyses (especially when using small animals like mussels). This dramatically reduces the quality of data by introducing missing attribute values and can have even more serious consequences. It is a common practice to pair the samples in order to have enough material to perform the chemical tests. This can however change the statistical properties of the data and as a result, lead to unexpected behavior of developed models, including false, highly positively biased accuracy estimates, which in consequence renders the models useless.

After the data has been collected it can finally be processed, which is the main focus of this paper. Although there have been several approaches to water quality prediction in the literature using neural networks (Maier and Dandy, Maier and Dandy), self organizing maps (Aguilera et al., 2001), Bayes networks (Reckhow, 1999) and other methods (Hamilton and Schladow, 1997), to our knowledge none of them was using biomarker data. From the point of view of data modelling, the biomarker data usually has low quality due to the missing values, high dimensionality and small size of the dataset, which can cause various problems (Bishop, 1995; Duda et al., 2000). Perhaps the most important of them is to define what does one expect the data to reveal and is the data adequate for this purpose. Apart from that issue, this paper addresses the choice of appropriate modelling technique from a large number of available methods, reliable estimation of future performance of the obtained model and various ways of dealing with low quality of data.

On many occasions researchers from outside the machine learning community try to apply various machine learning techniques to their data without much success. This frequently is a result of treating the machine learning methods as 'black boxes', while unfortunately, in most cases, successful and efficient use of these tools requires appropriate technical knowledge and experience. Thus this paper is a result of collaboration between the machine learning and environmental science communities, which shows and discusses how to design a purely data–driven, usable solution, making use of limited and deficient input data.

The rest of this paper is organized as follows. Section 2 describes the basic properties of the dataset, including some of its statistical characteristics and anticipated problems caused by the limited amount and low quality of the data.

In Section 3 we propose a model development workflow consisting of a number of clearly defined steps and allowing for systematic data analysis and predictive model building.

In Section 4, first individual models are built and their performance is measured for a number of data usage scenarios. It is also discussed in more detail how the data can be used and what one can expect of it.

Section 5 deals with the feature selection problem, investigating which biomarkers to use and which are not relevant for the problem at hand.

In Section 6, an ensemble model is described, building on the conclusions drawn from the previous sections and addressing in detail each of the difficulties caused by the quality of the dataset.

The experimental results for the ensemble model are given in Section 7. We show how the results have been improving by building a multistage combination of models and how various types of ensemble errors are correlated, to prove the reliability of estimation of future performance of our model. The usage of various features (biomarkers) by the final model and the source of errors (objects difficult to classify) are also presented and discussed.

Finally, the conclusions can be found in Section 8.

2. Dataset properties

2.1. Overview

The dataset contains a collection of biomarker data measured on mussels at 4 different marine stations located in South–West Norway (Rogaland County), in the course of a 4–week experiment. The stations have been chosen according to known water pollution levels (Grøsvik et al., 1999; Eriksen and Tvedten, 2002; Tvedten et al., 2002; Tvedten, 2003; Zorita et al., 2006) and the goal of the study was to provide field data to investigate the possible biomarker combinations to discriminate between various pollution levels. There are 50 objects[7] in the dataset, each having 12 attributes[8]. There are also 5 different classes, denoting the 5 stations, and 4% of attributes are missing. The locations of the sites can be seen in Figure 1, while the details of the classes have been given in Table 1 and the list of attributes with descriptions can be found in Table 2. From the point of view of building a usable classification model, a number of difficulties can already be expected even before examining the data in detail. The difficulties and their potential consequences are:

1. Small dataset size. This results in the lack of ability to use more advanced models with many degrees of freedom/parameters (e.g. all but the smallest neural networks) (Principe et al., 1999) and negatively influences the reliability of estimate of the model generalization ability.
2. Relatively high dimensionality of data. The number of attributes is higher than the number of objects per class, which can pose a whole number of difficulties known as the curse of dimensionality (Bishop, 1995), including the distance concentration phenomenon (Aggarwal et al., 2001; Francois et al., 2005).
3. Missing attributes. Although the level of missingness is not high, a number of problems arise here. First, most machine learning techniques do not natively support incomplete data, so some form of imputation is required. Secondly, from the statistical standpoint, the mechanism behind the missingness is not known. The only information is that the data is missing due to the fact that some biological tests have gone wrong in one way or another, but it is not known if there exists any relation between the test going wrong and the values of measured parameters. As a result, a common simplifying missing at random (Rubin, 1976) assumption may not hold and some form of a missingness model (Outhwaite and Stephen P Turner, 2007) may be required.

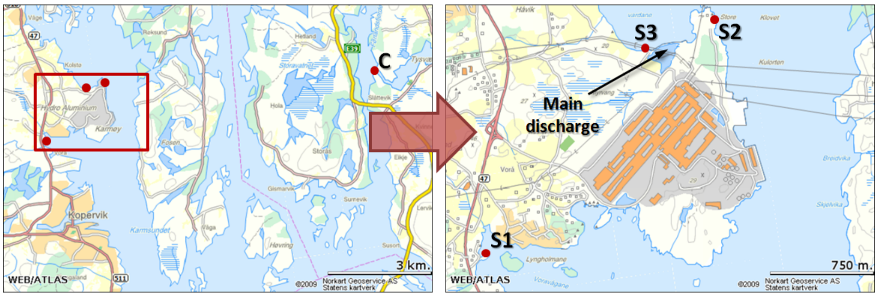
C
Kopervik
3 km.
WEB/ATLAS
©2009 Norkart Geoservice AS Statens kartverk
S3
S2
Main discharge
S1
750 m.
WEB/ATLAS
©2009 Norkart Geoservice AS Statens kartverk

Figure 1: Site locations

[7] The the words ‘object’, ‘instance’ or ‘sample’ are used interchangeably
[8] The words ‘attribute’, ‘feature’ or ‘biomarker’ are used interchangeably

Table 1: Class details

class	**objects**	**description**	**missing values**
T0-C	10	Control (clean) site at experiment start	10.0% (12 of 120)
T4-C	10	Control (clean) site after 4 weeks	0.0% (0 of 120)
T4-S1	10	Lightly polluted site after 4 weeks	2.5% (3 of 120)
T4-S2	10	Moderately polluted site after 4 weeks	2.5% (3 of 120)
T4-S3	10	Heavily polluted site after 4 weeks	5.0% (6 of 120)

2.2. Basic statistical analysis

Basic statistical analysis has been conducted in order to gain some insight into the structure of the dataset. For the estimation of statistical properties of the data, the missing values have been temporarily ignored and the dataset has been scaled to fit into the $0 \div 1$ range.

2.2.1. Mean and standard deviation

The mean and standard deviations for all attributes have been depicted in Figure 2, with the leftmost bar representing the whole dataset and remaining bars representing classes T0-C to T4-S3, left to right. It appears that the means differ between the classes, so the attributes should have some discriminative power. For example feature 9 alone may facilitate distinction between the most heavily polluted site and all the others. Note also, that features 1, 3 and especially 5[9] might as well be used to discriminate between classes T0-C and T4-C – the control site at the beginning and end of the experiment. This suggests some additional dependency in the system, as the feature values at the control site change over time although the pollution level does not. This phenomenon, known as concept drift (Tsymbal, Tsymbal), may render the predictions of the model less accurate as the time passes by.

2.2.2. Probability density functions

Class conditional probability density functions for each of the features can be seen in Figure 3. In almost every case the distributions overlap, thus none of the features alone is sufficient to discriminate between the classes. The exception is feature 9 – one of the classes (the heavily polluted site) is well separated. Also the peaks of the class conditional distributions of feature 11 form two, at least partially separated groups.

[9] An unexpected behavior of some of the models has been observed during the feature selection experiments. A simple Nearest Neighbour classifier trained on feature 5 alone produced a 0% 10–fold and 0% leave–one–out cross–validation error and the leave–one–out error of qdc in scenario 1 was also suspiciously low – 6%. It has turned out to be a result of pairing the samples to have enough material to perform the chemical tests. For this reason, feature 5 has been removed from the dataset before further analysis.

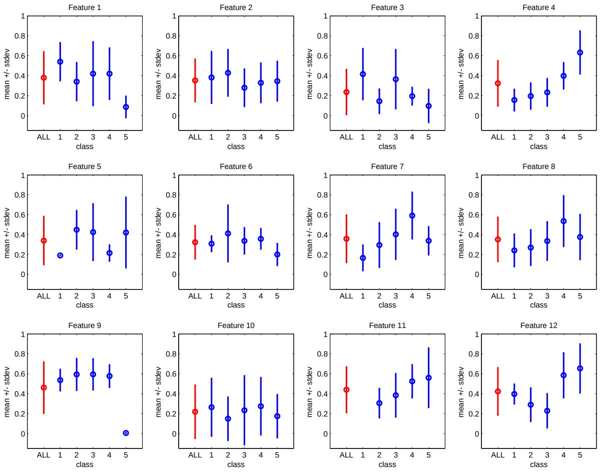

Figure 2: Mean values and standard deviations of all features for each class

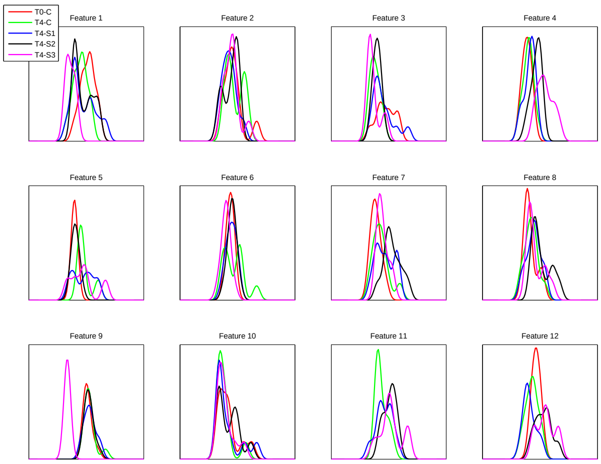

Figure 3: Class conditional probability density functions (colors denote classes)

Table 2: Attribute details

	Biomarker name	Method	Type of answer	Biological function/ meaning	Unit	Influenced by	Missing values
1	Lysosomal membrane stability (LMS)	Neutral Red Retention test, Histochemical assay	General toxicity of various classes of contaminants. General stress	Comprehensive information on progression of DG/haemocytes pathology and related disfunction	Minutes	Season, temperature, salinity	6.0% (3 of 50)
2	Haemaocrit value	Burker camera or haematocitometer	General stress	Haemocytes are the first defence system	Number of cells/ ml of haemolymph		0.0% (0 of 50)
3	Phagocytosis	Spectrophotometric on microplate, cell counting	Immunocompetence	Primary mechanism of immuno defence	Number of cells with zymosan/ mg proteins, of cells with zymosan/ ml haemolymph	Saliniy, temperature, food	4.0% (2 of 50)
4	Micronuclei frequency (MN)	cells stain on slide	Index of cytogenetic damage caused by genotoxic compound	Increased frequency	%		0.0% (0 of 50)
5	Metallothionein	Differential pulse polarography, radioimmunoassay, spectrophotometry ELISA, gene expression (molecular analysis)	Heavy metals (especially Cd, Zn, Cu, Hg), oxidative stress	Binding metals to limit availability, but also other protective functions (oxyradical scavenging)	μg/ g wet weight	Season, salinity	0.0% (0 of 50)
6	Glutathione-S-transferase (GST) activity	Spectrophotometric in cuvette or microplate	Exposure to PAHs, PCBs, Oxidative stress in general	Phase II enzime involved in detoxification o organic xenobiotics	nmol/min/mg proteins	Season (small), space	0.0% (0 of 50)
7	Catalase (CAT)	Spectrophotometric in cuvette or microplate	Oxidative stress	Antioxidant enzyme for the breakdown of hydrogen peroxid	μmol/min/mg proteins	Season, space	0.0% (0 of 50)
8	Superoxide Dismutase SOD)	Spectrophotometric	Oxidative stress	Catalyze the dismutation of superoxide anion into molecular oxygen and hydrogen peroxide, it is a metalloenzyme			0.0% (0 of 50)
9	TOSC-ROO	Reaction between ROO and substrate (KMBA) which is oxidase to ethylene	Oxidative stress	Total oxiradical scavenging capacity	TOSC unit/mg proteins		0.0% (0 of 50)
10	Malondialdehyde (MDA)	Spectrophotometric	Oxidative stress, damage of the lipid membranes, contamination of PCBs, PAHs, metals	Lipid peroxidation	nmol/ g wet weight	Season	0.0% (0 of 50)
11	Neutral lipid accumulation	Oil Red O technique	General stress, exposure to organic compouds	Accumulation of unsaturated neutral lipids, toxically induced disturbance of fat metabolism	Optical density (mean absorbance)		26.0% (13 of 50)
12	Lipofuscin	Schmol reaction	Oxidative	Accumulation of lipofuscines reflect degradation of cell membrane caused by oxidative damage following exposure to toxicants	Optical density (mean absorbance)		12.0% (6 of 50)

3. Model development workflow

Building upon the findings from the previous section, we now propose a predictive model development workflow, which has been depicted in Figure 4. The workflow consists of major steps of the model development process. Note, that the first step – Statistical Data Analysis has already been executed. The following steps have been described in detail in the next sections.

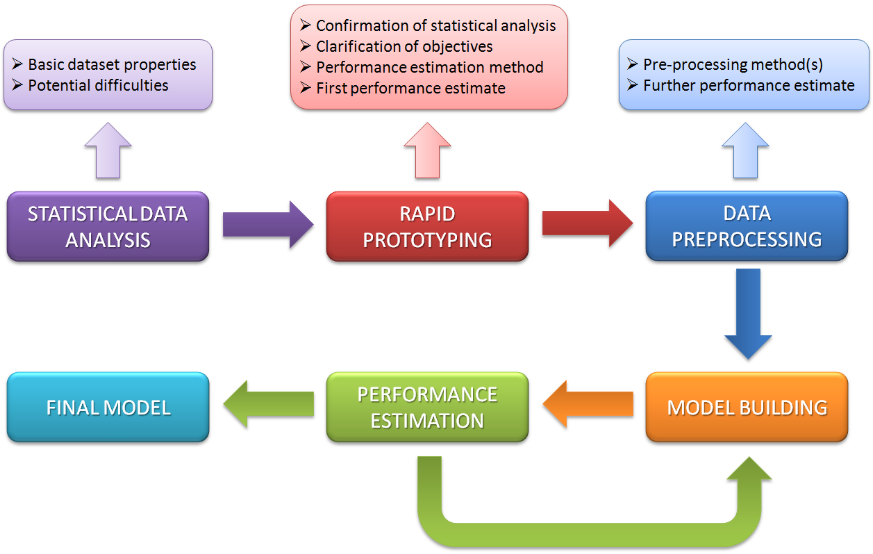
➢ Basic dataset properties
➢ Potential difficulties
➢ Confirmation of statistical analysis
➢ Clarification of objectives
➢ Performance estimation method
➢ First performance estimate
➢ Pre-processing method(s)
➢ Further performance estimate
STATISTICAL DATA ANALYSIS
RAPID PROTOTYPING
DATA PREPROCESSING
FINAL MODEL
PERFORMANCE ESTIMATION
MODEL BUILDING

Figure 4: Model development workflow

4. Classification with a single model

In order to quickly obtain a number of working prototype models, a simple classification experiment using a set of standard classifiers has been designed. It not only allows to get more detailed insight into the dataset and confirm the difficulties listed in Sections 2.1 and 2.2.1, but will also provide first performance estimates.

The classifiers used are a part of the PRTools (Duin et al., 2007) toolbox and their list is given in Table 3. The experiments have been primarily performed within a repeated 10–fold cross–validation scheme. Due to small size of the dataset and in order to obtain a better picture of possible performance, some experiments have been rerun using leave–one–out cross–validation.

We would like to stress here the importance of proper estimation of future model performance. It is not difficult to obtain a model with 0% classification error computed on the training dataset. The challenge however is to build a model which not only demonstrates low training data error, but will also perform well on new data not seen before (and thus not used for training). The

Table 3: Classifier details

acronym	**description**
fisherc	Fisher’s Linear Classifier using MSE minimization
ldc	Linear Bayes Normal Classifier / normal densities with common covariance
loglc	Logistic Linear Classifier / likelihood criterion and sigmoid function
nmc	Nearest Mean Classifier
nmsc	Nearest Mean Scaled Classifier / zero covariances and equal class variances
quadrc	Quadratic Discriminant Classifier / normal densities
qdc	Quadratic Bayes Normal Classifier / normal densities
udc	Uncorrelated Quadratic Bayes Normal Classifier / uncorrelated features
klldc	Linear Classifier using KL expansion of the common covariance matrix
pcldc	Linear Classifier using PC expansion on the joint data
knnc	K-Nearest Neighbor Classifier
parzenc	Parzen density based classifier
treec	Decision Tree Classifier
naivebc	Naive Bayes classifier / independency of features
svc	Support Vector Classifier (C–SVM)
nusvc	Support Vector Classifier (ν–SVM)

Table 4: Experiment scenarios

scenario	**class count**	**class details**
1	5	T0-C — T4-C — T4-S1 — T4-S2 — T4-S3
2	4	T0-C+T4-C — T4-S1 — T4-S2 — T4-S3
3	2	T0-C+T4-C — T4-S1+T4-S2+T4-S3
4	2	T0-C — T4-S1+T4-S2+T4-S3
5	2	T4-C — T4-S1+T4-S2+T4-S3
6	4	T0-C — T4-S1 — T4-S2 — T4-S3
7	4	T4-C — T4-S1 — T4-S2 — T4-S3
8	2	T0-C — T4-C

Table 5: 10–fold cross–validation experimental results

scenario	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
fisherc	35.6	37.9	22.5	21.5	19.7	29.2	26.0	41.5
ldc	**31.0**	32.9	20.8	21.7	22.7	27.3	25.8	41.5
loglc	43.2	35.8	20.2	28.7	23.5	40.0	32.5	34.0
nmc	36.0	36.1	**14.7**	17.8	**16.8**	22.8	28.3	30.0
nmsc	35.6	**29.1**	15.7	16.2	20.0	**22.5**	**24.0**	40.5
quadrc	56.0	54.0	33.8	22.2	36.7	48.8	48.0	40.5
qdc	60.0	53.7	33.8	38.7	37.5	50.5	51.7	31.5
udc	56.4	39.0	17.2	31.8	25.2	47.8	32.0	44.5
klldc	**31.0**	32.9	20.8	21.7	22.7	27.3	25.8	41.5
pcldc	**31.0**	32.9	20.8	21.7	22.7	27.3	25.8	41.5
knnc	44.8	45.0	21.2	19.8	44.0	36.0	42.0	37.5
parzenc	42.2	39.4	21.4	**14.8**	28.2	32.0	39.8	33.5
treec	66.6	67.1	32.7	25.7	49.5	55.8	57.7	40.5
naivebc	44.4	44.0	29.8	16.7	33.5	38.7	45.3	**11.5**
svc	38.8	34.3	**14.7**	36.2	50.5	27.3	29.0	31.5
nusvc	37.4	30.9	23.4	26.5	26.2	27.0	24.8	31.0
mean	43.1	40.3	22.7	23.8	29.9	35.0	34.9	35.8

Table 6: Leave–one–out cross–validation experimental results

scenario	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
fisherc	40.0	40.0	26.7	23.3	18.3	30.0	30.0	35.0
ldc	30.0	33.8	18.3	20.0	23.3	22.5	25.0	35.0
loglc	46.0	35.0	19.2	25.0	23.3	40.0	30.0	25.0
nmc	30.0	33.8	14.2	16.7	**16.7**	22.5	25.0	25.0
nmsc	36.0	**31.3**	15.8	16.7	**16.7**	22.5	25.0	40.0
quadrc	38.0	36.3	38.3	23.3	23.3	30.0	37.5	35.0
qdc	**12.0**	35.0	38.3	21.7	18.3	**10.0**	**10.0**	**10.0**
udc	58.0	40.0	15.8	31.7	25.0	50.0	35.0	45.0
klldc	30.0	33.8	18.3	20.0	23.3	22.5	25.0	35.0
pcldc	30.0	33.8	18.3	20.0	23.3	22.5	25.0	35.0
knnc	46.0	47.5	20.8	**15.0**	41.7	40.0	42.5	40.0
parzenc	40.0	40.0	22.5	**15.0**	31.7	32.5	40.0	35.0
treec	66.0	62.5	44.2	23.3	51.7	55.0	50.0	25.0
naivebc	42.0	43.8	29.2	**15.0**	33.3	35.0	50.0	**10.0**
svc	48.0	36.3	**13.3**	40.0	51.7	30.0	32.5	40.0
nusvc	38.0	35.0	21.7	26.7	28.3	27.5	25.0	35.0
mean	39.4	38.6	23.4	22.1	28.1	30.8	31.7	31.6

generalization error estimation becomes even more difficult when the amount of available data is severely limited, as in our case. The problem has been widely addressed in the literature (Duda et al., 2000; Weiss and Kulikowski, 1991) and a number of solutions has been devised, with cross–validation being the one used most commonly and successfully.

Since the percentage of missing features is rather small, at this stage a simple class–conditional mean imputation approach has been used to fill in the blanks. As in the case of class T0-C all values of feature 11 were missing, they have been replaced with a global mean value for the whole dataset. The dataset has been scaled to fit within the $0 \div 1$ interval, as the ranges of the original features vary greatly and feature number 5 has been removed.

A total of 8 different experiment scenarios, summarized in Table 4, has been devised. The goal was to see if any of the scenarios can be ruled-out at the early stage of experiments due to lack of discriminative power of the feature set. Additionally, some of the scenarios were chosen on purpose in order to verify the anticipated difficulties.

The results of preliminary experiments can be found in Tables 5 and 6.

4.1. Scenario 1 – all 5 classes

The first experiment involved classification of objects into one of 5 classes given in Table 1. The mean 10–fold cross–validation error of all classifiers (43.1%) is higher than the mean leave–one-out error (39.4%) mostly due to suspiciously good performance of qdc in the latter case (12.0%). Note, that before removing feature 5 from the dataset, leave–one–out qdc error was equal to 6.0%, while performances of other classifiers were more or less the same. As a result, all experiments with ensembles of classifiers were conducted only within the 10–fold cross–validation scheme, as the leave–one–out approach appears unreliable.

4.2. Scenario 2 – control site and various pollution degrees

For this experiment classes T0-C and T4-C have been combined together to form a single, control class. The results of both cross–validation approaches are once again consistent but not remarkable, although the 10–fold cross–validation mean error of all classifiers has been slightly reduced from roughly 43% to about 40%. Combination of the two control classes thus seems to have positive influence on the classification error. Moreover, this approach is the only way to address the concept drift issue with this limited amount of data. As a result we have decided to focus on scenario 2 in further experiments.

4.3. Scenario 3 – clean and polluted environment

In this scenario classes T0-C and T4-C have been combined together to form a single, control class. Classes T4-S1, T4-S2 and T4-S3 have also been combined to form a single class representing polluted sites. This resulted in a dramatic improvement of the classification accuracy (roughly 23% mean error in both cross–validation scenarios and 14.7% 10–fold CV error of best classifiers).

4.4. Scenario 4 and 5 – clean (T0-C / T4-C) and polluted environment

In these two scenarios, classes T4-S1, T4-S2 and T4-S3 have been combined to form a single class representing polluted environment. One of the clean environment classes has then been dropped (T0-C for scenario 4 and T4-C for scenario 5 respectively), effectively reducing the number of classes to two.

The most important thing to notice is the performance gap between those two scenarios (mean errors), reaching 6pp[10] in favor of scenario 4. This confirms the presence of concept drift in the data as the discrimination between control site at the beginning of the 4–week experiment (first data collection process) and polluted sites appears much easier.

4.5. Scenario 6 and 7 – control site (T0-C / T4-C) and various pollution degrees

Similarly to scenarios 4 and 5, classes T0-C (scenario 6) and T4-C (scenario 7) have been dropped respectively, while the classes representing various degrees of pollution have remained unchanged. The mean leave–one–out errors for both scenarios are lower than the 10–fold CV errors due to surprisingly good performance of the qdc – this issue has been already discussed, so only the latter errors seem to be meaningful in this case. The mean errors for both scenarios do not differ, although the difference in errors of individual classifiers reaches 8pp in some cases. This seems to contradict the results of experiments in scenarios 4 and 5. Note however, that in case of these previous experiments, the polluted class consisted of 30 objects, so the classifiers could be trained better. For this reason the results of scenarios 4 and 5 should be treated as more reliable.

[10] pp stands for percentage point, a concept causing a lot of confusion in the literature; a change from 10% to 20% is an increase by 10pp or 100%

4.6. Scenario 8 – control site at time T0 and T4

This experiment scenario has been designed to check if the classes T0-C and T4-C are indeed different. A dataset consisting of objects from only those two classes has been used to test the classification performance. Although both mean errors are quite high (over 30%), the best performing classifier naivebc has produced only 11.5% 10–fold CV error. Notice, that no anomalies similar to qdc have ever occurred in case of this particular classifier, so we have no reason to treat its error estimate as unreliable. This confirms that the objects collected at the same site in two different moments have distinct properties, influenced by factors other than the pollution level.

5. Feature selection

As mentioned in section 2.1, the dimensionality of the dataset is relatively high. This fact can often be very problematic for various machine learning techniques, since they are forced to operate in a sparse space (the number of data objects required to fill a d–dimensional space grows exponentially with d) and thus cannot be trained properly. As a result, reduction of the number of attributes usually has a significant, positive influence on the classification performance.

There is also another practical reason for using as few attributes as possible – the data acquisition cost. By identifying attributes which are correlated or otherwise irrelevant, one can reduce the number of tests needed to be performed during the data collection process. This not only saves money but is also especially important for the biomarker data collection, where some biological tests are mutually exclusive or destructive and the amount of biological material is usually limited.

As a result, we have decided to reduce the number of attributes by applying some preprocessing technique. Experiments described in this section aim to investigate which of the features have the lowest discriminative power and how their removal might affect the classification performance. We also check if some form of feature transformation might be beneficial for model performance. The experiments have been conducted only for scenario 2 from Table 4.

5.1. Removal of one feature at a time

The classification results for removal of one feature at the time (thus using 10 remaining features) have been given in Table 7. A modest improvement of mean classification error over previous experiments has been observed for removal of features 1, 6, 10 and 11. Further removal of features could possibly improve the results even more, but enumeration of all feature pairs, triplets etc. is a problem of exponential complexity, thus we do not run this experiments here and conclude that some form of feature selection should improve the classification performance.

Table 7: Single feature removal

feature	**1**	**2**	**3**	**4**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
fisherc	36.0	36.9	40.6	38.1	36.6	40.0	40.4	44.0	32.4	34.8	**35.6**
ldc	30.5	37.6	32.9	38.8	32.3	**33.5**	**29.6**	43.6	28.1	31.8	39.0
loglc	45.4	37.3	44.3	**29.1**	30.9	35.1	40.1	43.5	36.0	36.8	39.3
nmc	34.0	35.0	34.5	41.6	34.8	36.0	34.9	44.6	29.8	31.3	38.0
nmsc	28.4	**30.0**	**29.9**	38.6	**26.6**	35.0	32.4	**39.5**	28.7	**26.1**	42.8
quadrc	49.4	50.2	47.8	52.9	48.1	54.5	49.1	55.9	48.8	50.3	51.1
qdc	54.8	50.1	60.5	57.9	55.9	48.9	59.8	60.3	61.3	58.3	57.4
udc	37.8	38.5	40.0	43.6	32.8	43.6	36.6	46.3	34.5	39.2	46.1
klldc	30.5	37.6	32.9	38.8	32.3	**33.5**	**29.6**	43.6	28.1	31.8	39.0
pcldc	30.5	37.6	32.9	38.8	32.3	**33.5**	**29.6**	43.6	28.1	31.8	39.0
knnc	41.1	38.6	43.9	42.5	38.5	47.6	43.3	50.6	38.1	38.9	41.3
parzenc	34.9	36.1	40.8	38.0	38.8	44.4	41.9	49.7	39.5	38.6	40.1
treec	61.6	62.8	66.3	67.9	65.8	66.3	63.5	67.3	65.9	66.3	65.5
naivebc	48.0	45.7	44.8	52.1	45.9	44.3	45.6	54.6	45.3	44.3	44.7
svc	36.4	35.3	35.0	35.5	35.2	40.4	38.0	43.3	32.6	35.7	36.0
nusvc	**28.1**	31.6	36.8	36.4	30.4	35.1	35.1	44.3	**28.0**	32.6	36.1
mean	39.2	40.1	41.5	43.2	38.6	42.0	40.6	48.4	37.8	39.3	43.2

5.2. Classification using a single feature

In this experiment performance of a classifier built on a single feature has been tested. The results have been given in Table 8. Clearly, none of the features alone facilitates acceptable classification performance but in majority of cases it is still better than random guessing (75%). Also, there are two features which demonstrate the lowest error – feature 4 and 9. The latter is especially interesting since the probability density plot (Figure 3) suggested possible discriminative power to separate class T4-S3 from all other classes. A quick experiment using only those 2 features has revealed 37.8% mean classification error, which already is an improvement over the results obtained using all 11 features. This experiment has also uncovered suspicious properties of feature 5 mentioned earlier – 0 error of the nearest neighbor classifier (not given in Table 8).

5.3. Principal Component Analysis (PCA)

Principal Component Analysis is a procedure for transformation of a number of possibly correlated variables into a smaller number of uncorrelated variables called principal components. The first principal component accounts for as much of the variability in the data as possible, and each consecutive component accounts for as much of the remaining variability as possible (Duda et al., 2000). PCA is thus a procedure for reduction of dataset dimensionality preserving the maximum level of variance. Note, that PCA does not take advantage of class information given with the data and as a result can be considered as an unsupervised procedure.

The percentages of explained cumulated variance and the classification performance for all numbers of principal components (scenario 2) have been given in Table 9. The best results have been obtained for just 2 principal components, both in terms of mean error (35.5%) and error of the best individual classifier

Table 8: Single feature classification performance

feature	**1**	**2**	**3**	**4**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
fisherc	66.3	73.5	65.4	57.5	69.5	61.2	68.8	50.0	78.6	71.3	63.7
ldc	73.4	67.9	57.3	47.1	67.8	60.0	66.6	53.7	80.0	79.2	57.4
loglc	71.1	**66.5**	61.3	54.5	65.0	61.1	**64.3**	54.8	80.7	76.8	**53.1**
nmc	73.4	67.9	57.3	47.1	67.8	60.0	66.6	53.7	80.0	79.2	57.4
nmsc	73.4	67.9	57.3	47.1	67.8	60.0	66.6	53.7	80.0	79.2	57.4
quadrc	64.5	73.9	57.0	53.0	59.5	**55.3**	68.2	52.4	81.2	71.1	61.0
qdc	**64.8**	75.8	58.2	**45.6**	**59.0**	56.0	67.5	54.1	77.3	74.5	65.9
udc	**64.8**	75.8	58.2	**45.6**	**59.0**	56.0	67.5	54.1	77.3	74.5	65.9
klldc	73.0	67.9	57.3	47.1	67.8	60.0	66.6	53.7	80.0	79.2	57.4
pcldc	73.0	67.9	57.3	47.1	67.8	60.0	66.6	53.7	80.0	79.2	57.4
knnc	67.6	77.9	68.8	52.1	79.3	65.1	70.9	53.3	**73.8**	69.0	69.4
parzenc	67.6	71.3	61.4	50.4	62.2	57.1	77.8	54.9	80.5	70.0	67.9
treec	69.1	74.5	71.9	51.9	77.6	65.9	65.6	51.1	75.1	76.4	69.0
naivebc	69.3	75.4	71.0	48.2	59.6	58.1	73.4	**44.0**	83.2	76.4	66.6
svc	74.9	75.0	75.0	57.3	75.0	67.6	74.8	50.0	75.0	72.9	75.0
nusvc	65.9	83.8	**56.0**	55.3	76.0	64.1	75.9	50.6	77.1	**63.6**	63.9
mean	69.5	72.7	61.9	50.4	67.5	60.5	69.2	52.4	78.7	74.5	63.0

(27.3%, udc). This is surprising as the first two components account for only 47.1% of the variance.

Examination of PCA rotation matrix reveals that all original features are relevant as no weights are driven to zero.

Table 9: Classification performance on PCA–transformed dataset

PCs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**
variance	29.7	47.1	60.2	70.3	77.7	84.2	89.0	93.0	96.2	98.7	100.0
fisherc	50.0	39.0	37.8	34.3	35.4	37.1	39.2	38.6	34.5	36.0	35.0
ldc	**39.4**	33.9	40.3	33.4	34.1	30.3	**29.5**	35.4	**31.6**	30.0	**30.3**
loglc	46.0	31.8	42.6	36.5	34.6	33.5	34.1	**31.6**	32.7	37.6	42.0
nmc	**39.4**	31.4	36.1	35.5	37.1	35.9	36.5	38.8	33.8	34.3	33.6
nmsc	**39.4**	27.8	33.6	**31.3**	**33.6**	**30.0**	31.5	34.6	34.6	**28.9**	30.9
quadrc	40.4	29.0	39.8	41.8	42.4	44.4	51.4	61.7	56.0	52.5	52.4
qdc	43.3	31.3	40.9	41.4	41.6	42.4	47.1	54.6	53.3	59.1	54.0
udc	43.3	**27.3**	**31.4**	38.6	41.2	39.8	41.4	40.6	42.3	41.1	35.4
klldc	**39.4**	33.9	40.3	33.4	34.1	30.3	**29.5**	35.4	31.6	30.0	**30.3**
pcldc	**39.4**	33.9	40.3	33.4	34.1	30.3	**29.5**	35.4	31.6	30.0	**30.3**
knnc	47.5	34.0	37.0	36.1	40.9	45.5	41.5	46.8	48.5	45.5	45.1
parzenc	51.2	30.6	35.3	39.3	40.1	38.5	37.5	35.9	36.8	38.6	39.0
treec	46.4	52.6	55.7	56.0	55.5	56.4	58.0	57.1	57.3	59.3	59.7
naivebc	63.8	40.6	44.4	52.1	53.3	60.5	62.1	68.9	66.9	63.6	63.8
svc	52.3	48.5	45.5	41.8	42.0	41.8	38.0	38.8	39.1	36.5	36.0
nusvc	44.1	43.3	42.8	40.0	40.3	39.4	41.0	36.6	36.4	36.2	35.9
mean	45.3	35.5	40.2	39.0	40.0	39.7	40.5	43.2	41.7	41.2	40.8

5.4. Linear Discriminant Analysis (LDA)

Linear Discriminant Analysis is a method which tries to find a linear projection of the data which best separates the classes and is thus useful for discrimination purposes. The shortcoming of LDA is that the maximum dimensionality of the projection is limited to $C - 1$, where C is the number of classes (Duda et al., 2000). Unlike PCA though, LDA tries to takes advantage of the class information, so better classification performance can be expected.

Unfortunately, in our case there is no performance gain when compared to PCA (32.5% error of the best classifier). Moreover, examination of the transformation matrix also does not indicate irrelevance of any of the original features, likely due to the high dimensionality reduction level.

6. Ensemble of classifiers

Having insight into the structure of the dataset and potential performance of individual classifiers, we have decided to construct an ensemble model, which constitutes the 'Model Building' step of the proposed workflow depicted in Figure 4.

The rationale behind using a combination of classifiers rather than a single best model is quite intuitive. Various classifiers tend to differ for reasons ranging from different underlying mathematical models to different data or attributes used during the training process. This usually leads to another tendency of making classification error on different objects. By taking advantage of this fact, it is possible to exploit this complementarity of various models and construct an ensemble able to outperform any individual classifier. There are of course many ways to do this and a comprehensive review of various methods can be found in (Kuncheva, 2004).

For training of individual models, we have decided to go for feature selection rather than transformation, for a number of reasons. First of all, the latter approach did not demonstrate a considerable performance improvement at the same time bringing in the loss of interpretability of the results. Moreover, feature selection may facilitate reduction of data acquisition costs – if some attributes are never used, there is no need to measure them by performing expensive biological tests.

In the following sections three different terms are being used in order to refer to an individual classifier:

- Base classifier – one of the classifiers from Table 3,
- Candidate classifier – base classifier using a subset of features and being considered for inclusion into some combination. Candidate classifier can differ by the choice of the base classifier, choice of feature subset or both,
- Component classifier – candidate classifier included in a combination.

6.1. Architecture of the ensemble

The ensemble has been built using a simple majority vote rule (Ruta and Gabrys, 2001) to obtain a multi–level combination of classifiers (Ruta and Gabrys, 2002, 2005). Special measures have been taken in order to enforce diversity in the pool of classifiers and to improve handling of missing data. The most important assumptions of the ensemble approach are:

1. Variety of base classifiers, covering all classifiers specified in Table 3, except for svc which was very slow to train and usually inferior to nusvc in terms of classification accuracy.
2. Non-exhaustive feature subset search performed using three different greedy algorithms: forward, backward and plus-l-takeaway-r feature selection, all executed with default parameters in a 10–fold cross–validation scheme on the whole dataset. Error rate of each of the base classifiers has been in turn used as a criterion for feature selection, the procedure has been repeated 10 times and candidate classifiers have been created using all obtained unique feature subset/base classifier pairs.
3. Maximum likelihood imputation from univariate class conditional distributions rather than mean imputation. The procedure involved estimation of the probability density function for each class/feature pair using the Parzen window method (Duda et al., 2000) and imputation of the most likely value from this distribution. Figure 5 depicts an example of how the value imputed using this approach may vary from the class conditional mean. In our experiments this imputation method allows to achieve on average about 2.5pp 10–fold cross–validation error improvement using all 11 features, with the most improved classifier (loglc) better by 7.4pp.
4. Incorporation of a missingness model. The model creates a binary missingness map for the training dataset (denoting a missing value by 1). The columns with all 0's are then dropped (they correspond to the features of the original dataset which are never missing) and the training dataset is augmented with the remaining part of the missingness map by treating each column of the map as a new feature. The missingness model is used only if it is beneficial for each particular candidate classifier. This is achieved by creating two versions of each candidate classifier, with and without the missingness model incorporated. The latter versions are then given priority when shortlisting the classifiers for combinations (if both versions produce the same validation error). We have observed an average 10–fold cross–validation error improvement of about 5pp due to using the described missingness model, with the most improved classifier (quadr) better by as much as 28.8pp.

6.2. Experiment scenario

The experiment with combined classifiers and feature selection has been designed using a nested cross–validation scheme, presented in Figure 6. First, the whole dataset for scenario 2 from Table 4 has been randomly divided into 10 test folds, each consisting of 5 objects (1 object per class). Then each of those 10 test folds in turn has been put aside as test data and a number of combined models has been constructed using the remaining 9 folds (now called validation folds) in a similar, iterative manner: each of the 9 validation folds in turn has been put aside, all candidate classifiers have been trained on the remaining 8 folds and tested on the validation fold. After iterating over all 9 validation folds, a binary validation error map has been constructed for each of the candidate

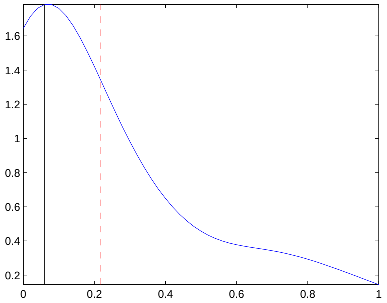

Figure 5: Mean imputation (dashed line) vs. imputation from univariate distribution (solid line) for the 10^{th} feature

classifiers and the validation set errors have been calculated. After iterating over all 10 test folds a total of 21 (due to limited resources) best classifiers in terms of the validation error have been selected, retrained on all 9 folds and used to create an exhaustive set of combinations, with a constraint that each combination can consist only of an odd number of component classifiers. This has resulted in a grand total of over 1 million combinations, which could then been combined again in a similar way to obtain a multi–level model. Note, that the experiment has been designed in such a way, that the test data has not been used for training or selection of classifiers/combinations at any stage, which is expected to result in more accurate estimates of the future performance.

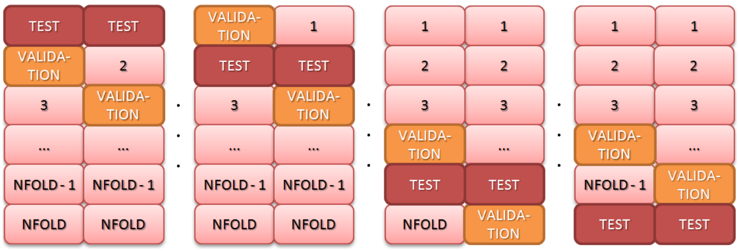

Figure 6: Nested cross–validation scheme. NFOLD denotes the total number of folds.

7. Classification performance

The generalization performance of built models estimated by the classification errors on the test set have been given in Table 10. There were 569 candidate classifiers in total with mean error of 31.1%. The test error of the best component classifier (parzenc built on features 1, 3, 4, 7, 9 and 12) is 18.0%, which compared to the previous results from Table 5 (29.1%, all features used) and Table 9 (27.3%, PCA) is a considerable improvement achieved due to the feature selection mechanism, maximum likelihood imputation and incorporation of the missingness model. The same classifier has produced the lowest validation error equal to 20.31% and although this does not necessarily imply the lowest test error, the two types of errors are well correlated, as it will be discussed later. Note, that at this stage the majority vote (MV) error (21.8%) did not improve over the error of the best classifier.

Table 10: Test errors of combinations, component and candidate classifiers

test error	**min**	**mean**	**max**	**MV**	**count**
candidate classifiers	18.0	31.1	57.3	**21.8**	569
component classifiers	18.0	22.3	26.4	**16.4**	21
level 1 combinations (all)	13.2	16.9	22.6	**16.6**	1 048 576
level 1 combinations (better than mean)	13.2	16.5	19.2	**16.4**	558 072
level 1 combinations (better than min+std)	13.6	15.4	17.2	**14.4**	1 381
level 2 combinations (all)	10.2	11.3	12.8	**12.0**	524 288
level 2 combinations (better than mean)	10.2	11.2	12.6	**11.0**	215 222
level 2 combinations (better than min+std)	10.4	10.9	11.6	**10.6**	95

As mentioned earlier, due to limited resources only 21 best candidate classifiers in terms of the validation error have been selected for combining. Their MV test error is equal to 16.4%, which for the first time is less than the error of the best candidate classifier. Exhaustive search for the best combination of 21 component classifiers (level 1 combinations) brings further improvements. Test error of the best level 1 combination is 13.2%. This time however the best test set model is only 6286^{th} in the validation data performance ranking of combinations and thus there is no reason to prefer this particular model over the rest. The test error of the highest ranked combination is 14.6% and the MV error of a subset of best level 1 combinations (with validation error within one standard deviation from the minimal validation error) is 14.4%.

For the level 2 combinations, 20 best level 1 models have been chosen and once again combined exhaustively. This time the best combination produced 10.2% test error (13.56% validation error, 96^{th} in the ranking) and the lowest MV error achieved was 10.6%. An attempt for another combination level did not produce further improvements.

Note, that the multilevel ensemble structure has been obtained by iterating over the two steps of the model development workflow from Figure 4 (i.e. 'Model Building' and 'Performance Estimation') until no improvement could be achieved.

7.1. Correlation between test and validation error

During the selection of models to be combined, the validation error has been used as the criterion. To confirm that it was indeed the right choice, we have calculated the correlation between those two types of errors. At the level of individual candidate classifiers the correlation is very high (0.9662), but it drops considerably for level 1 combinations (0.6076) just to decrease even further for level 2 combinations (0.4889). This can also be seen on the plots of test vs. validation errors given in Figure 7. First thing to notice is that by combining individual models both errors have been dragged towards the origin and are much more concentrated – the variability of error in the pool of models is smaller. Also, the test error is on average smaller than the validation error which is due to the amount of data used for training (45 objects in the case of test set and 40 in the case of the validation set). Note that high error correlation in the case of individual classifiers is partly caused by much wider range of validation errors than for combinations on both levels, while the ranges of test errors are similar.

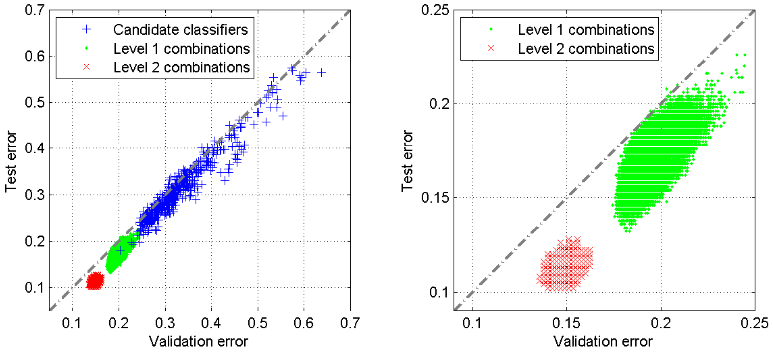

Figure 7: Test versus validation errors (mean values of repeated cross–validation)

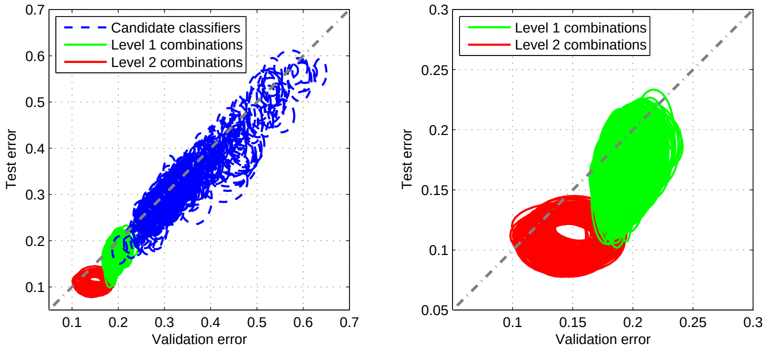

Figure 8: Test versus validation errors (mean values + stdev of repeated cross–validation)

Figure 8 depicts test vs. validation errors again, but this time the errors of each model are represented by ellipses with semiaxes equal to standard deviations of the respective errors. It can be well seen especially on the close–up plot, that apart from reducing the mean error value, the variance of test error has also diminished.

7.2. Difficult objects

To understand where do the errors come from a plot of misclassification rates of the 95 best level 2 combinations has been given in Figure 9. Each bar represents the percentage of level 2 models which have misclassified a particular object (objects 1 to 20 represent the control site, 21 to 30 class T4-S1 and so on). Due to the combination method used, any object with misclassification rate ≥ 0.5 will be misclassified by the combined model as well. As it can be seen, there are 5 such objects in the dataset which corresponds to 10% error (the actual error is 10.6% as it has been averaged over 10 runs and so were the misclassification rates).

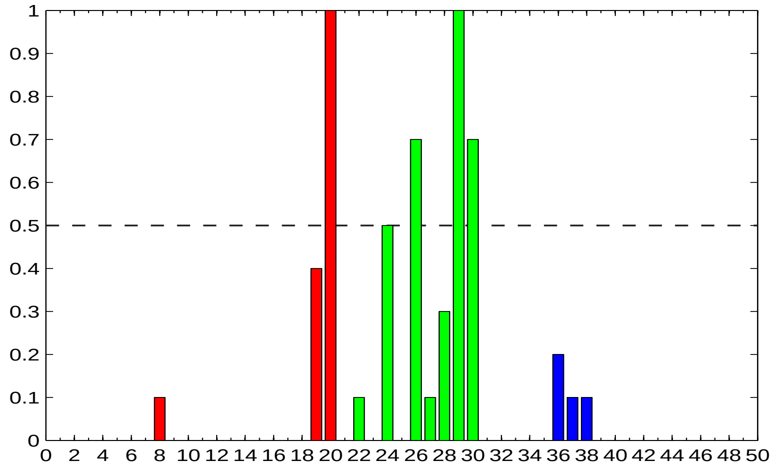

Figure 9: Misclassification rates of the best level 2 models for each test object

There is a number of interesting observation to be made here. First, the classification of objects belonging to the heavily polluted site (objects 41 through 50) does not pose a problem, as none of the models makes an error there. The same applies to most of the objects belonging to the moderately polluted site as only in the case of three of them the models slightly disagree regarding the class label. Also most of the objects belonging to the control site are classified correctly without any difficulties. There are two exceptions however – object 20 is never classified correctly and object 19 is classified correctly only by a small margin. The class causing difficulties is the lightly polluted site with 4 (that is 40%) of objects being misclassified, 3 of which by a considerable margin. This allows to presume, that although the biomarkers used in this study facilitate rather good discrimination between the classes in most cases, they are not discriminative enough to separate the objects coming from the lightly polluted site (T4-S1) and the control site (T0-C + T4-C). It might be the case that the pollution level at the site T4-S1 is so low, that these particular biomarkers simply fail to detect it. Thus a possible future research direction is to focus on the evaluation of usability of various biomarkers for detection of very low pollution levels.

7.3. Feature usage

The usage of particular features by all component classifiers included in the top 95 level 2 combinations has been depicted in Figure 10. As it can be seen, apart from feature 5, which has been dropped deliberately, there is another completely irrelevant feature – 10. Also the attribute number 1 is used by less than 20% component classifiers, so it might be considered as the second candidate for removal. The rest of the features appears important, with attribute 9 absolutely crucial (used by all component classifiers).

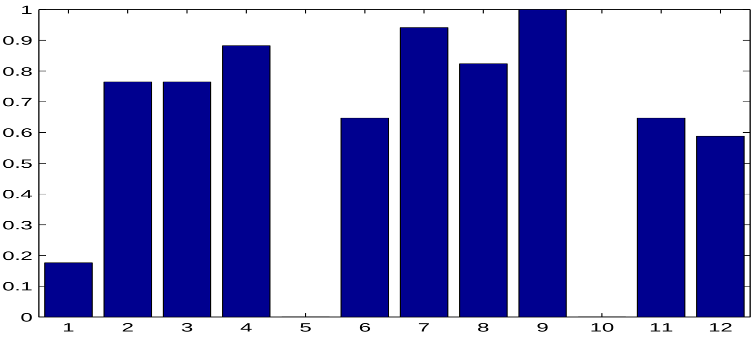

Figure 10: Feature usage by component classifiers

The above confirms that the biomarkers selected to form the input of the predictive system were suitable for the task, as there is no doubt that 9 out of 12 attributes are relevant. Although the choice was good, it was not necessarily optimal – it is possible that some other set of biomarkers would even facilitate error–free classification. The point we are trying to make here is that it's almost impossible to tell which biomarkers should be used for a particular task before the actual model is built. Thus in a perfect–world scenario one would run as many biological test as possible and then select an optimal subset of attributes during the model building process, in a way similar to what we have done. There are two limiting factors here however.Each biological test has an associated monetary cost and a total cost of performing a batch of tests is limited. Moreover, due to ethical reasons the amount of biological material used should be as small as possible, to leave the environment in its original state. This leads to an interesting problem of balancing the classification performance with the total cost (both monetary and ethical) of performing a set of biological tests. In other words, the ability to estimate how much would it cost to achieve a given performance level or how accurate can the predictive model be given a known cost limit, might be very desirable. Such analysis could even reveal that the best performance might be achieved using only a small set of relatively inexpensive tests, which would minimize both the total cost and classification error at the same time. Although with the limited amount of data we were not able to address this problem here, it is a promising research direction, definitely worth considering in future studies.

8. Conclusions

Coastal water pollution monitoring using the biomarker data is very appealing as the biomarkers are able to detect even a very low concentration of pollutants, unobservable using different methods. Due to the specific data collection process however, the biomarker data is quite difficult to process in order to obtain meaningful results. Blindly applying one of the many available machine learning techniques is seldom a successful approach, thus in this paper we have described a whole predictive model building methodology, which can be used for similar problems as well.

The most important conclusions of this work are:

- biomarkers can be successfully used for discrimination between various aquatic toxicity levels even when small amount of data is available,
- in order to deal with imperfections of the biomarker data, a sophisticated multi–stage ensemble model had to be built, addressing not only the limited amount of data but also its low quality,
- the choice of biomarkers for a specific predictive task is an important issue as it can dramatically influence performance of the built models and is also strongly connected with the monetary cost and ethical issues of data acquisition,
- due to the fact that the environment is evolving regardless of the changes in pollution level, the results could be further improved if some environmental features were also measured (e.g. water temperature).

Although the predictive model obtained by following the model development workflow presented in this paper performs relatively well, some open problems remain. First, it is still not known how to select the biological tests to be performed, before building the prototype of the model (i.e. before actually performing the tests). Since the literature on using biomarkers for predictive modelling is discordant, it seems that at least for some time this choice will need to remain more or less random.

The above issue leads to another interesting problem of balancing the model performance with data acquisition cost. It might be the case that for some applications, models cheap to develop, yet not very accurate would be sufficient, while for some other purpose the precision of the model will be the most important factor, regardless of the cost.

Acknowledgements

The data used in this paper have been collected during the project "Developing an Index of the Quality of the Marine Environment (Marine Environment I.Q.) based on biomarkers: integration of pollutant effects on marine organisms", Project 173343/S40, funded by the Research council of Norway. Thanks are due to the staff at IRIS-Biomiljø for collaboration in the sampling and analysis phases.

References

Aggarwal, C., A. Hinneburg, and D. Keim (2001). On the surprising behavior of distance metrics in high dimensional space. *Lecture Notes in Computer Science 2001*, 420–435.

Aguilera, P., A. Frenich, J. Torres, H. Castro, J. Vidal, and M. Canton (2001). Application of the Kohonen neural network in coastal water management: methodological development for the assessment and prediction of water quality. *Water Research 35*(17), 4053–4062.

Amiard, J., C. Amiard-Triquet, S. Barka, J. Pellerin, and P. Rainbow (2006). Metallothioneins in aquatic invertebrates: their role in metal detoxification and their use as biomarkers. *Aquatic Toxicology 76*(2), 160–202.

Barsiene, J., J. Lazutka, J. Syvokiene, V. Dedonyte, A. Rybakovas, E. Bagdonas, A. Bjornstad, and O. Andersen (2004). Analysis of micronuclei in blue mussels and fish from the Baltic and North Seas. *Environmental toxicology 19*(4), 365.

Bishop, C. (1995). *Neural Networks for Pattern Recognition.* Oxford University Press, USA.

Bocchetti, R. and F. Regoli (2006). Seasonal variability of oxidative biomarkers, lysosomal parameters, metallothioneins and peroxisomal enzymes in the Mediterranean mussel Mytilus galloprovincialis from Adriatic Sea. *Chemosphere 65*(6), 913–921.

Bresler, V., A. Abelson, L. Fishelson, T. Feldstein, M. Rosenfeld, and O. Mokady (2003). Marine molluscs in environmental monitoring. *Helgoland Marine Research 57*(3), 157–165.

Chèvre, N., F. Gagné, P. Gagnon, and C. Blaise (2003). Application of rough sets analysis to identify polluted aquatic sites based on a battery of biomarkers: a comparison with classical methods. *Chemosphere 51*(1), 13–23.

Dahlhoff, E. (2004). Biochemical indicators of stress and metabolism: applications for marine ecological studies.

Depledge, M., A. Aagaard, and P. Gyørkøs (1995). Assessment of trace metal toxicity using molecular, physiological and behavioural biomarkers. *Marine Pollution Bulletin 31*(1-3), 19–27.

Depledge, M. and M. Fossi (1994). The role of biomarkers in environmental assessment (2). Invertebrates. *Ecotoxicology 3*(3), 161–172.

Duda, R., P. Hart, and D. Stork (2000). *Pattern Classification 2nd ed.* Wiley-Interscience.

Duin, R., P. Juszczak, P. Paclik, E. Pekalska, D. de Ridder, D. Tax, and S. Verzakov (2007). Pr-tools 4.1, a matlab toolbox for pattern recognition.

Eason, C. and K. O'Halloran (2002). Biomarkers in toxicology versus ecological risk assessment. *Toxicology 181*, 517–521.

Eriksen, V. and Ø. Tvedten (2002). Resipientundersøkelse i karmsundet for fmc biopolymer. Technical report, RF-Rogalandsforskning.

Forbes, V., A. Palmqvist, and L. Bach (2006). The use and misuse of biomarkers in ecotoxicology. *Environmental Toxicology and Chemistry 25*(1), 272–280.

Francois, D., V. Wertz, and M. Verleysen (2005). Non-Euclidean metrics for similarity search in noisy datasets. *Proceedings of the European Symposium on Artificial Neural Networks*, 339–334.

Goldberg, E. (1986). The mussel watch concept. *Environmental Monitoring and Assessment 7*(1), 91–103.

Goldberg, E. and K. Bertine (2000). Beyond the Mussel Watchnew directions for monitoring marine pollution. *Science of the Total Environment, The 247*(2-3), 165–174.

Grøsvik, B., E. Aas, and J. Brseth (1999). Overvking av miljeffekter ifm. legging av gassrrledning over karmsundet. Technical report, Akvamilj AS.

Hamilton, D. and S. Schladow (1997). Prediction of water quality in lakes and reservoirs. Part Imodel description. *Ecological Modelling 96*(1-3), 91–110.

Harvey, J. and J. Parry (1997). The detectioon of genotoxin-induced DNA adducts in the common mussel Mytilus edulis. *Mutagenesis 12*(3), 153.

Hellou, J. and R. Law (2003). Stress on stress response of wild mussels, Mytilus edulis and Mytilus trossulus, as an indicator of ecosystem health. *Environmental Pollution 126*(3), 407–416.

Kuncheva, L. (2004). *Combining pattern classifiers: methods and algorithms.* Wiley-Interscience.

Lam, P. and J. Gray (2003). The use of biomarkers in environmental monitoring programmes. *Marine pollution bulletin 46*(2), 182–186.

Lesser, M. (2006). Oxidative stress in marine environments: biochemistry and physiological ecology.

Livingstone, D., J. Chipman, D. Lowe, C. Minier, and R. Pipe (2000). Development of biomarkers to detect the effects of organic pollution on aquatic invertebrates: recent molecular, genotoxic, cellular and immunological studies on the common mussel (Mytilus edulis L.) and other mytilids. *International Journal of Environment and Pollution 13*(1), 56–91.

Magni, P., G. De Falco, C. Falugi, M. Franzoni, M. Monteverde, E. Perrone, M. Sgro, and C. Bolognesi (2006). Genotoxicity biomarkers and acetylcholinesterase activity in natural populations of Mytilus galloprovincialis along a pollution gradient in the Gulf of Oristano (Sardinia, western Mediterranean). *Environmental Pollution 142*(1), 65–72.

Maier, H. and G. Dandy. The use of artificial neural networks for the prediction of water quality parameters. *Water Resources Research 32*(4).

Moore, M., D. Lowe, and A. Koehler (2004). Biological effects of contaminants: measurments of lysosomal membrane stability. *ICES techniques in marine environmental sciences 36*, 31.

Ott, W., A. Steinemann, and L. Wallace (2006). *Exposure analysis.* CRC.

Outhwaite, W. and S. Stephen P Turner (2007). *Handbook of Social Science Methodology.* SAGE Publications Ltd.

Pace, M. (2001). Prediction and the aquatic sciences. *Canadian Journal of Fisheries and Aquatic Sciences 58*(1), 63–72.

Peakall, D. (1994). The role of biomarkers in environmental assessment (1). Introduction. *Ecotoxicology 3*(3), 157–160.

Principe, J., N. Euliano, and W. Lefebvre (1999). *neural and adaptive systems: Fundamentals through simulations with CD-ROM.* John Wiley & Sons, Inc. New York, NY, USA.

Rank, J. and K. Jensen (2003). Comet assay on gill cells and hemocytes from the blue mussel Mytilus edulis. *Ecotoxicology and Environmental Safety 54*(3), 323–329.

Reckhow, K. (1999). Water quality prediction and probability network models. *Canadian Journal of Fisheries and Aquatic Sciences 56*(7), 1150–1158.

Regoli, F., G. Winston, V. Mastrangelo, G. Principato, and S. Bompadre (1998). Total oxyradical scavenging capacity in mussel Mytilus sp. as a new index of biological resistance to oxidative stress. *Chemosphere 37*(14-15), 2773–2783.

Rubin, D. (1976). Inference and missing data. *Biometrika 63*(3), 581–592.

Ruta, D. and B. Gabrys (2001). Analysis of the correlation between majority voting error and the diversity measures in multiple classifier systems. *Proceedings of the 4th International Symposium on Soft Computing*, 1824–025.

Ruta, D. and B. Gabrys (2002). A theoretical analysis of the limits of majority voting errors for multiple classifier systems. *Pattern Analysis & Applications 5*(4), 333–350.

Ruta, D. and B. Gabrys (2005). Classifier selection for majority voting. *Information fusion 6*(1), 63–81.

Tsymbal, A. The problem of concept drift: definitions and related work. *Informe técnico: TCD-CS-2004-15, Departament of Computer Science Trinity College, Dublin, https://www. cs. tcd. ie/publications/techreports/reports 4*, 2004–15.

Tvedten, Ø. (2003). Pah- og metallinnhold i blåskjell, torsk og krabbe fra karmsundet. Technical report, RF-Rogalandsforskning.

Tvedten, Ø., V. Eriksen, and Å. Molværsmyr (2002). Miljøtilstand og tilførsler til karmsundet. Technical report, RF-Rogalandsforskning.

Weiss, S. and C. Kulikowski (1991). *Computer systems that learn: classification and prediction methods from statistics, neural nets, machine learning, and expert systems.* Morgan Kaufmann Publishers Inc. San Francisco, CA, USA.

Widdows, J. and F. Staff (2006). Biological effects of contaminants: Measurement of scope for growth in mussels. *ICES Techniques in Marine Environmental Sciences* (40), 30.

Yang, H., Q. Zeng, E. Li, S. Zhu, and X. Zhou (2004). Molecular cloning, expression and characterization of glutathione S-transferase from Mytilus edulis. *Comparative Biochemistry and Physiology, Part B 139*(2), 175–182.

Zorita, I., M. Ortiz-Zarragoitia, M. Soto, and M. Cajaraville (2006). Biomarkers in mussels from a copper site gradient (Visnes, Norway): An integrated biochemical, histochemical and histological study. *Aquatic Toxicology 78*, 109–116.