



**International Doctoral School in  
Information and Communication Technology**

Department of Information Engineering and Computer Science  
University of Trento

**ACTIVE LEARNING METHODS FOR  
CLASSIFICATION AND REGRESSION PROBLEMS**

Edoardo Pasolli

Advisor: Prof. Farid Melgani, University of Trento



*A Chiara*



## Abstract

*In the pattern recognition community, one of the most critical problems in the design of supervised classification and regression systems is given by the quality and the quantity of the exploited training samples (ground-truth). This problem is particularly important in such applications in which the process of training sample collection is an expensive and time consuming task subject to different sources of errors. Active learning represents an interesting approach proposed in the literature to address the problem of ground-truth collection, in which training samples are selected in an iterative way in order to minimize the number of involved samples and the intervention of human users.*

*In this thesis, new methodologies of active learning for classification and regression problems are proposed and applied in three main application fields, which are the remote sensing, biomedical, and chemometrics fields. In particular, the proposed methodological contributions include: i) three strategies for the support vector machine (SVM) classification of electrocardiographic signals; ii) a strategy for SVM classification in the context of remote sensing images; iii) combination of spectral and spatial information in the context of active learning for remote sensing image classification; iv) exploitation of active learning to solve the problem of covariate shift, which may occur when a classifier trained on a portion of the image is applied to the rest of the image; moreover, several strategies for regression problems are proposed to estimate v) biophysical parameters from remote sensing data and vi) chemical concentrations from spectroscopic data; vii) a framework for assisting a human user in the design of a ground-truth for classifying a given optical remote sensing image.*

*Experiments conducted on simulated and real data sets are reported and discussed. They all suggest that, despite their complexity, ground-truth collection problems can be tackled satisfactory by the proposed approaches.*

## Keywords

Active learning, classification, electrocardiographic signals, Gaussian processes, ground-truth, regression, remote sensing, spectrometric data analysis, support vector machines.



# Contents

<b>1. INTRODUCTION AND THESIS OVERVIEW .....</b>	<b>1</b>
1.1. CONTEXT .....	2
1.2. PROBLEMS .....	2
1.3. THESIS OBJECTIVE, SOLUTIONS AND ORGANIZATION .....	6
1.4. REFERENCES CITED IN CHAPTER 1 .....	7
<b>2. ACTIVE LEARNING METHODS FOR ELECTROCARDIOGRAPHIC SIGNAL CLASSIFICATION .....</b>	<b>11</b>
2.1. INTRODUCTION .....	12
2.2. SUPPORT VECTOR MACHINE CLASSIFICATION .....	14
2.3. ACTIVE LEARNING METHODS .....	15
2.3.1. <i>Margin Sampling</i> .....	15
2.3.2. <i>Posterior Probability Sampling</i> .....	16
2.3.3. <i>Query by Committee</i> .....	18
2.4. EXPERIMENTS ON SIMULATED DATA .....	18
2.4.1. <i>Data Set Description</i> .....	18
2.4.2. <i>Experimental Results</i> .....	19
2.5. EXPERIMENTS ON REAL ECG DATA .....	22
2.5.1. <i>Data Set Description</i> .....	22
2.5.2. <i>Experimental Results</i> .....	22
2.5.3. <i>Experiments on Unseen Recordings</i> .....	25
2.6. CONCLUSION .....	26
2.7. ACKNOWLEDGMENT .....	27
2.8. REFERENCES CITED IN CHAPTER 2 .....	27
<b>3. SVM ACTIVE LEARNING THROUGH SIGNIFICANCE SPACE CONSTRUCTION.....</b>	<b>29</b>
3.1. INTRODUCTION .....	30
3.2. PROPOSED METHOD.....	30
3.3. EXPERIMENTS .....	32
3.3.1. <i>Experimental Setup</i> .....	32
3.3.2. <i>Experimental Results</i> .....	33
3.4. CONCLUSION .....	36
3.5. ACKNOWLEDGMENT .....	36
3.6. REFERENCES CITED IN CHAPTER 3 .....	36
<b>4. SVM ACTIVE LEARNING USING SPATIAL INFORMATION.....</b>	<b>39</b>
4.1. INTRODUCTION .....	40
4.2. PROPOSED METHOD.....	41
4.2.1. <i>Proposed Active Learning Framework</i> .....	41
4.2.2. <i>Spectral Selection Criterion: Margin Sampling</i> .....	43
4.2.3. <i>Spatial Selection Criteria</i> .....	43
4.2.4. <i>Nondominated Sorting</i> .....	45
4.3. EXPERIMENTS .....	46
4.3.1. <i>Data Set Description</i> .....	46
4.3.2. <i>Experimental Setup</i> .....	48
4.3.3. <i>Experimental Results</i> .....	49
4.4. CONCLUSION .....	55
4.5. ACKNOWLEDGMENT .....	56
4.6. REFERENCES CITED IN CHAPTER 4 .....	56
<b>5. USING ACTIVE LEARNING TO ADAPT REMOTE SENSING IMAGE CLASSIFIERS .....</b>	<b>59</b>
5.1. INTRODUCTION .....	60
5.2. COVARIATE SHIFT AND ACTIVE LEARNING .....	61
5.2.1. <i>The Problem of Covariate Shift</i> .....	61
5.2.2. <i>Active Learning to Correct Data Set Shift</i> .....	63
5.2.3. <i>On the Need of an Exploration-Focused Heuristic</i> .....	64
5.3. DATA AND EXPERIMENTAL SETUP .....	64
5.3.1. <i>Data Sets</i> .....	64

5.3.2. <i>Experimental Setup</i> .....	66
5.4. RESULTS AND DISCUSSION .....	67
5.4.1. <i>Urban Data</i> .....	67
5.4.2. <i>Agricultural Data</i> .....	71
5.5. CONCLUSION .....	73
5.6. ACKNOWLEDGMENT .....	73
5.7. REFERENCES CITED IN CHAPTER 5 .....	73
<b>6. ACTIVE LEARNING METHODS FOR BIOPHYSICAL PARAMETER ESTIMATION .....</b>	<b>77</b>
6.1. INTRODUCTION .....	78
6.2. GAUSSIAN PROCESS AND SUPPORT VECTOR MACHINE REGRESSION .....	79
6.2.1. <i>Gaussian Process Regression</i> .....	79
6.2.2. <i>Support Vector Machine Regression</i> .....	81
6.3. PROPOSED ACTIVE LEARNING METHODS .....	82
6.3.1. <i>Active Learning Strategies for GP Regression</i> .....	84
6.3.2. <i>Active Learning Strategies for SVM Regression</i> .....	86
6.4. EXPERIMENTS .....	87
6.4.1. <i>Data Set Description and Experimental Setup</i> .....	87
6.4.2. <i>Experimental Results</i> .....	89
6.5. CONCLUSION .....	94
6.6. ACKNOWLEDGMENT .....	94
6.7. REFERENCES CITED IN CHAPTER 6 .....	94
<b>7. ACTIVE LEARNING FOR SPECTROSCOPIC DATA REGRESSION .....</b>	<b>97</b>
7.1. INTRODUCTION .....	98
7.2. PARTIAL LEAST SQUARES REGRESSION .....	99
7.3. PROPOSED ACTIVE LEARNING METHODS .....	100
7.3.1. <i>Active Learning Strategies for PLSR</i> .....	102
7.3.2. <i>Active Learning Strategies for SVM Regression</i> .....	103
7.4. EXPERIMENTS .....	104
7.4.1. <i>Data Set Description and Experimental Setup</i> .....	104
7.4.2. <i>Experimental Results</i> .....	106
7.5. CONCLUSION .....	108
7.6. ACKNOWLEDGMENT .....	109
7.7. REFERENCES CITED IN CHAPTER 7 .....	109
<b>8. A FRAMEWORK FOR COMPUTER-AIDED GROUND-TRUTH COLLECTION FOR OPTICAL IMAGE CLASSIFICATION .....</b>	<b>111</b>
8.1. INTRODUCTION .....	112
8.2. PROPOSED FRAMEWORK .....	113
8.2.1. <i>Level Set Segmentation</i> .....	114
8.2.2. <i>Segment Selection</i> .....	116
8.2.3. <i>Segment Labeling and Sampling</i> .....	117
8.3. EXPERIMENTAL RESULTS .....	118
8.3.1. <i>Data Set Description and Experimental Setup</i> .....	118
8.3.2. <i>Experimental Results</i> .....	120
8.4. CONCLUSION .....	127
8.5. ACKNOWLEDGMENT .....	127
8.6. REFERENCES CITED IN CHAPTER 8 .....	127
<b>9. CONCLUSIONS .....</b>	<b>129</b>
<b>10. LIST OF RELATED PUBLICATIONS .....</b>	<b>133</b>
10.1. PUBLISHED JOURNAL PAPERS .....	133
10.2. JOURNAL PAPERS IN REVISION .....	133
10.3. JOURNAL PAPERS IN PREPARATION .....	133
10.4. CONFERENCE PROCEEDINGS .....	133



# 1. Introduction and Thesis Overview

*Abstract – In this chapter, we describe briefly the general context in which the thesis is positioned. In a second step, the specific problems faced in the following chapters are introduced. Finally, the corresponding proposed solutions and an overview of the thesis organization are given.*

## 1.1. Context

Automatic recognition, description, classification, and grouping of patterns are important problems in a variety of engineering and scientific disciplines such as statistics, computer-aided diagnosis, marketing, computer vision, biomedicine, and remote sensing. These problems call for two major questions: 1) what is a pattern that a machine may know? and 2) what is a pattern recognition machine? A pattern can be defined as an entity, vaguely defined, which could be a fingerprint image, a handwritten cursive word, a human face, a set of multispectral observations, etc... [1]. In general, pattern recognition can be seen as a research field that aims at studying how machines can observe the environment, learn to distinguish patterns of interest from their background, and make reasonable decisions about the categories of the patterns. In this context, the general scheme of a pattern recognition system can be subdivided in three main steps as synthesized in Fig. 1.1. In the first step, observations from the physical world are gathered by means of sensors and conveniently converted into digital format for computer-based processing. The pre-processing phase aims at: 1) reducing the possible errors derived from the acquisition phase as they may have a negative influence on the following analysis; 2) providing a suitable representation of the data/objects to recognize. Finally, the analysis phase aims at extracting the required information (product) from the considered data. In particular, two main problems can be identified in the pattern recognition field, namely classification and regression. The purpose of classification is to assign each input value to one of a given set of classes, while in regression problems a real value is associated with each input.

In the literature, classification and regression tasks have been approached from a methodological point of view in two very well-established ways: 1) the supervised approach, in which an input pattern (sample) is identified as a member of a predefined class/value; and 2) the unsupervised approach, in which a sample is assigned to a natural class/value inferred through similarity measures. In the supervised approach, the mapping from the set of input samples  $\mathbf{x} \in \mathcal{R}^d$  (where  $d$  is the feature space dimensionality) to a finite set of labels  $y$  is carried out after inferring a mathematical function  $y=f(\mathbf{x})$  from a training set  $L = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n)\}$ , i.e., a set of  $n$  samples for which the label is *a priori* known (labeled data). We note that for classification problems  $y \in \{1, \dots, T\}$ , where  $T$  is the number of considered classes, while in the regression context  $y$  assumes a real value. The goodness of the obtained function is evaluated by how well it generalizes, i.e., how accurately it performs on new samples, termed as test set, assumed to follow the same statistical distribution characterizing the training data. In the unsupervised approach, no labeled data are *a priori* available. Therefore, the objective becomes the one of partitioning input samples into groups called clusters, in such a way that members of the same cluster are as similar as possible and samples from different clusters are as dissimilar as possible. Accordingly, the availability or not of labeled data heavily constrains the kind of classification approach to be taken into consideration and thus the whole recognition process.

## 1.2. Problems

In general, methods based on supervised approaches have shown very promising performances in many different fields, but they require *a priori* information about the considered classification/regression task, and thus the intervention of human users. In the literature, most of the attention has been given on improving the accuracy of the classification process by acting mainly at the following three levels: 1) data representation; 2) discriminant function model; and 3) criterion on the basis of which the discriminant functions are optimized [2]. These works are based on an essential assumption that is the samples used to train the classifier/regressor are statistically representatives of the classification/regression problem to solve. However, the process of collection of training samples is not trivial, because the human intervention is subject to errors and costs in terms of both time and money. Therefore, the quality and the quantity of such samples are very important, because they have a strong impact on the performances of the classifier/regressor.

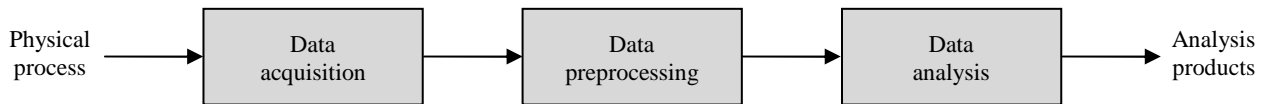


Fig. 1.1. Flow chart of a general pattern recognition system.

Only in the last few years, in the literature there has been a growing interest in developing methods focused on the problem of the construction of the training sample set, also called ground-truth. In particular, the objective is to develop automatic strategies or semi-automatic procedures based on interactive processes with human users.

A first problem in ground-truth collection is given by the mislabeling issue due to errors in the process of sample labeling. For example, focusing on the remote sensing field, ground-truth collection can be done by following two main approaches: 1) in situ observation and 2) photo-interpretation [3]. Each of them has its own advantages and drawbacks, but both are subject to errors. In the first case, this may occur because of georeferencing problems, while in the second one, spectral mismatching errors by human users are the main source of problems. Since the presence of mislabeled training samples has a direct negative impact on the classification/regression process, the development of automatic techniques for validating the collected samples is crucial. In the literature still few solutions for coping with this issue have been proposed. Focusing on classification problems, they are based on two main approaches. The first one admits the presence of mislabeled samples, but aims at designing a classifier that is less influenced by this presence [4]. The second one tends to identify and remove the mislabeled samples from the training set. An early work derived from this strategy for  $k$ -nearest neighbor ( $k$ NN) classification suggested first to apply a 3NN classification over the whole learning set and then to remove misclassified samples in order to produce a new learning set on the basis of which a 1NN classifier is formed for the classification phase [5]. In [6], in order to avoid overfitting of noisy samples, the author proposed to perform the removal process through the C4.5 decision tree classifier. In [7], the suspect samples are identified and removed from the learning set by means of an ensemble of three classifiers (i.e., C4.5,  $k$ NN, and linear classifiers). In particular, a sample is expected to be mislabeled if it is misclassified by the ensemble of classifiers. In [8], the authors propose a preliminary filtering procedure. A sample is suspect when in its neighbourhood defined by a geometrical graph the portion of examples of the same class is not significantly greater than in the entire data set. Such suspect samples in the training data can be removed or relabeled. The filtering training set is then provided as input to a 1NN learning algorithm. While typical works focus on cleaning the training data by either discarding or correcting mislabeled instances, in [9] the authors propose a different approach. For each training sample, a probability vector of class membership is calculated and thus the confidence on the current label is used as a weight during the training phase. The probability vector is calculated such that clean samples have a high confidence on its current label, while mislabeled ones have a low confidence on its current label and a high confidence on its correct label. The probability distribution over the class labels is calculated using a clustering technique based on the expectation maximization algorithm. In [10], the authors present a kernel-based approach able to filter the mislabeled samples. The mislabeled detection issue is viewed as an optimization problem based on the weighted  $k$ NN classifier, a modification of the classic  $k$ NN algorithm that allows taking into account the similarity between samples. In [11], a Bayesian classifier is used to estimate the probabilities of each sample to belong to the considered classes. Then, the value of entropy is calculated from the probabilities and used to evaluate the typicality of the sample to belong to the classes. Finally, the samples with low entropy, but with a wrong prediction, are identified as mislabeled samples. In [12], the mislabeled sample detection issue is viewed as an optimization problem where it is looked for the best subset of learning samples in terms of statistical separability between classes. The method supposes that classes follow a Gaussian distribution.

Another problem frequent in real application scenarios is represented by the scarcity of available training samples due to complexity and cost that characterize the ground-truth construction process. Accordingly, this constrains the classification/regression process to be carried out with small numbers of training samples, thus leading to weak estimates of the classifier/regressor parameters and potentially bad classification/regression performances, in particular if class distributions are overlapped. A possible solution to this issue consists to exploit the large number of unlabeled samples that are typically available at zero cost from the data under analysis. Indeed, the improvement of the classifier/regressor accuracy is obtained by combining automatically labeled and unlabeled samples. Methods dealing with this issue are termed typically as semisupervised methods. Focusing on classification problems, they are based on two main principles, termed as inflation and transduction principles. The inflation principle relies on the idea of augmenting the original training set by exploiting a set of unlabeled samples, which covers a portion of the whole set of samples. For this purpose, the labels of the unlabeled samples need to be beforehand estimated. In the literature, the most intuitive way to perform inflation is the so-called self-training strategy [13], which is based on the following steps: 1) exploit the available training samples to construct an initial classification model; 2) generate a first guess of the label of each unlabeled sample to transform it into a semilabeled sample; 3) inflate the training set; 4) refine up to convergence of both classifier model and the label estimates, using iteratively the resulting inflated training set. Another inflation-based method is the cotraining method, which splits the feature space into two different feature subspaces and trains a classifier on each subspace [14]. Initially, the two classifiers are trained only with the training samples. Then, each classifier is used to classify the unlabeled samples and subsequently retrained with the training set augmented with the semilabeled samples for which both classifiers feel most confident. The process is repeated up to convergence. In this way, each classifier passes its knowledge to the other so that they can cooperatively improve their performance. The expectation-maximization algorithm is one of the most common techniques for integrating unlabeled samples in the classification process. In [15], unlabeled samples are initially classified by means of the expectation-maximization algorithm applied just on the original training samples. Then, the resulting semilabeled samples are merged with the original training samples to update class statistics, and the samples are reclassified by the updated statistics. This process, which assigns full weight to training samples but automatically gives reduced weight to semilabeled samples, is repeated until convergence is reached. In [16], the inflation principle is exploited to improve performance of the  $k$ NN classifier in terms of computational cost. Unlabeled samples are integrated in the learning process to yield a finer cell partitioning of the feature space and to significantly increase the number of predefined decision regions. The result is a drastic diminution of the classification cost. The transduction principle is conceptually completely different from the inflation one. This is due to two main reasons: 1) all available samples and not just part of them contribute to the learning process; and 2) training and classification/regression steps are fused into a unique step. This last point means that there is no explicit and separate classification/regression step, as commonly known. Indeed, the best classification for the whole set of samples is generated during the training process. Such principle was pioneered by Vapnik in the context of the statistical learning theory for classification problems [17]. In particular, the transductive support vector machine (TSVM) proposed in [18] represents a key reference for this category of semisupervised classifiers. The TSVM optimization function is almost similar to that of the standard inductive SVM with the difference that it also integrates the data label estimation problem in order to look for the maximal margin hyperplane over both training and all unknown samples. In [19], the authors propose a TSVM that exploits a weighting strategy for unlabeled samples based on a time-dependent criterion and addresses the problem of suboptimal model selection. In [20], a transductive method is proposed for the classification of hyperspectral data. It formulates the semisupervised classification problem through a graph-based representation in which each sample spreads its label information to its neighbors until a global stable state is reached for the whole image samples. The authors integrate also spatial contextual information in the classification process by means of

opportune composite kernels. In [21], the label estimation process is performed within a multiobjective optimization framework based on genetic algorithms, in which each chromosome of the evolving population encodes the label estimates as well as the SVM classifier parameters for tackling the model selection issue. Such a process is guided by the joint minimization of two different criteria that express the generalization capability of SVM classifier. The two explored criteria are an empirical risk measure and an indicator of the classification model sparseness, respectively. Considering regression problems, very scarce attention has been paid to semi-supervised learning. Among the available methods, one can retain the one based on the cotraining of two  $k$ NN estimators whose tasks during the learning phase are to provide, for each other, guesses of the targets of the unlabeled samples [22]. The final prediction is made by averaging the regression estimates generated by both estimators. In [23], the authors propose a method for SVM regression, in which the integration of the unlabeled samples in the regression process is controlled through a particle swarm optimization (PSO) framework. Two different optimization criteria are adopted, which are empirical and structural expressions of the generalization capability of the resulting semi-supervised regression system.

Given the constraints in terms of time and money related to the acquisition process of training samples, in the last few years there has been also a growing interest in developing strategies for the semi-automatic selection of the training samples. In the machine learning field, the active learning approach represents an interesting solution to face this problem. Considering a small and suboptimal initial training set, few additional samples are selected from a large amount of unlabeled data (learning set). These samples are labeled by the human expert and then added to the training set. The entire process is iterated until a stopping criterion is satisfied. The aim of active learning is to rank the learning set according to an opportune criterion that allows to select the most useful samples to improve the model, thus minimizing the number of training samples necessary to maintain discrimination capabilities as high as possible. In the last few years, different solutions have been proposed and applied successfully in different applications fields. Considering classification problems, in [24] the authors propose the query by committee method, in which a committee of classifiers is used. In particular, the samples with the maximum disagreement between the different classifiers are selected. In [25] a probabilistic active learning strategy based on SVM designed for large data applications is presented. It queries for a set of samples according to a distribution as determined by the current separating hyperplane and an adaptive confidence factor. The confidence factor is estimated from local information using the  $k$ NN principle. In [26], the active sampling-at-the-boundary method is applied using orthogonal pillar vectors lying on the decision boundary to learn the classification decision hyperplane in a multidimensional space. This shows that the proposed strategy can be applied with fewer training samples, rather than randomly selecting training data near the decision hyperplane. Both perceptron algorithm and SVM are used to estimate the decision boundary. In [27], the authors propose the method called confidence-based active learning, for training a wide range of classifiers, such as SVM, neural networks, and Naive Bayesians. The approach selects and requests annotation only for uncertain samples, i.e., for those samples that cannot be classified within a certain conditional error. Thus, it estimates the uncertainty value for each sample according to its output score from a classifier and select only samples with uncertainty value above a user-defined threshold. A dynamic bin width allocation method is proposed to estimate sample conditional error. In [28], the query-by-transduction algorithm is proposed. It is based on  $p$ -values obtained from a transductive learning procedure in a stream-based setting where samples are observed sequentially. When a new example is observed, different classifiers are constructed and statistical information is derived by considering all the possible labels for the new sample. Then, statistical information of the two most likely labels for the new sample is used to decide on whether to select the new sample. The utility of the proposed method is shown on both binary and multiclass classification problems using SVM as classifier. In [29], active learning is applied to the multi-label image classification problem. The authors propose a two-dimensional strategy, in which both the sample and the label dimensions are considered. The reason is that the contributions of different labels to minimize the classification error are different due to the

inherent label correlations. Similarly, the active learning approach has been studied for regression problems by the machine learning and statistics communities, in which it is also known as *optimal experimental design*. After the seminal paper by Cohn et al. [30], in which active learning has been applied to two statistically-based learning architectures, such as mixtures of Gaussians and locally weighted regression, several works have appeared in the last few years. For instance, in [31], the authors focus on the problem of local minima in active learning for neural networks, and two probabilistic solutions are proposed. In [32], after introducing the fundamental limits in a minimax sense of active and passive learning for various function classes, some strategies based on a tree-structured partition of the data are presented. In [33], considering linear regression scenarios, a method using the weighted least-squares learning based on the conditional expectation of the generalization error is proposed. In [34], the authors apply the query by committee approach in the regression context. The main idea is to train a committee of learners and query the labels of the samples where the committee's prediction differ, thus minimizing the variance of the learner by training on samples where variance is largest. In [35], it is suggested to solve the problems of active learning and model selection at the same time in order to improve further the generalization performance. In [36], a solution to the problem of pool-based active learning in linear regression is proposed. In [37], the authors develop a strategy for kernel-based linear regression, in which the proposed greedy algorithm employs a minimum-entropy criterion derived using a Bayesian interpretation of ridge regression. Despite the promising performance given by the active learning approach in the regression field, nothing similar has been proposed in the remote sensing literature.

### 1.3. Thesis Objective, Solutions and Organization

As introduced in the previous subsection, active learning approach is a smart solution to the problem of training sample collection for supervised classification and regression problems. Although in the last few years several strategies have been proposed in the literature, it still represents a research field of great interest because of its important implications. For this reason, the objective of this thesis is to propose new methodologies of active learning in different application fields. In particular, three main fields have been considered, namely remote sensing, biomedical, and chemometrics.

After this introductory section, the thesis is organized into eight chapters. In Section 2, the active learning approach is introduced in the biomedical field. In particular, we focus on the classification of electrocardiographic signals using SVM classification. Three different strategies are described: 1) margin sampling, in which the samples of the learning set more close to the hyperplane between the different classes are chosen; 2) posterior probability sampling, in which posterior probabilities are estimated for each class. Then the samples that maximize the entropy between the posterior probabilities are selected; and 3) query by committee in which a pool of classifiers is trained on different features to label the set of learning samples. Then, the samples with the maximum disagreement between classifiers are chosen. In Section 3, a new strategy for SVM classification in the context of classification of remote sensing images is proposed. It relies on the idea of: 1) reformulating the original classification problem into a new problem where it is needed to discriminate between significant and non significant samples, according to a concept of significance which is proper to SVM theory; and 2) constructing the corresponding significance space so that to suitably guide the selection of the samples potentially useful to better deal with the original classification problem. While strategies proposed in the literature for the remote sensing image classification formulate the active learning problem in the spectral domain only, in Section 4 we propose to combine spectral and spatial information in order to improve the process of training sample selection. In particular, three criteria based on spatial information are introduced in order to encourage the selection of samples distant from the samples already composing the current training set. In the first strategy, we compute the Euclidean distances in the spatial domain from the training samples, while the second one is based on the Parzen window method applied in

the spatial domain. Finally, the last criterion involves the concept of spatial entropy. In Section 5, we investigate the problem of covariate shift in the remote sensing field. A classifier trained on training samples acquired from a region of the image can fail if used to classify the entire image. Indeed, training samples often suffer from a sample selection bias and do not represent the variability of spectra that can be encountered in the entire image. Therefore, to maximize classification performance, it is necessary to adapt the first model to the new data distribution. In this Section, we propose to perform adaptation by sampling new training samples in unknown areas of the image. Our goal is to select these pixels in an intelligent fashion that minimizes their number and maximizes their information content. Two strategies based on uncertainty and clustering of the data space are considered to perform active selection. In particular, the breaking ties active sampling strategy is used with a linear discriminant analysis. After presenting in the previous sections strategies for classification problems, in Section 6 the active learning approach is used in the regression context. In particular, we focus on the estimation of biophysical parameters from remote sensing data. Various strategies specific for Gaussian Process (GP) and SVM regression are proposed. For GP regression, the first two strategies are based on the idea of adding samples that are distant from the current training samples in the kernel space, while the third one uses a pool of regressors in order to select the samples with the greater disagreements between the different regressors. Finally, the last strategy exploits an intrinsic GP regression outcome to pick up the most difficult and hence interesting samples to label. For SVM regression, the method based on the pool of regressors and two additional strategies based on the selection of the samples distant from the current support vectors are proposed. Similarly, in Section 7 the active learning approach is used for regression problems in the chemometrics field. In particular, we consider the problem of the estimation of chemical concentrations from spectroscopic data. In this case, the proposed strategies are specifically developed for partial least squares regression (PLSR) and SVM regression. For PLSR, the first method is based on adding samples that are distant from the current training samples in the feature space, while the second one uses a pool of regressors. For SVM regression, the method based on the pool of regressors and an additional strategy based on the selection of the samples distant from the support vectors are proposed. In Section 8, a novel framework for assisting a human user in the design of a ground-truth for classifying a given optical remote sensing image is proposed. It is based on automatic unsupervised procedures of level set segmentation and clustering to make both spatial and spectral information contribute in the ground-truth design. In particular, it allows identifying the most significant areas of the image and facilitating the manual labeling operation. The resulting ground-truth is classifier-free and can be further improved by making it classifier-driven through an active learning process. Finally, general conclusions on the methodological and experimental developments conveyed by the present thesis are drawn in Section 9.

This dissertation has been written under the assumption that the reader is familiar with the methodological aspects related to pattern recognition processing. In the opposite case, the references available at the end of this section may be used for consultation since they provide a valuable and exhaustive introduction to the concepts used in the following sections. These latter have been structured in such a way to make them self-contained avoiding to the reader the necessity to read all the chapters preceding the one of interest.

## 1.4. References cited in Chapter 1

- [1] A. K. Jain, R. P. Duin, and J. Mao, "Statistical pattern recognition: a review," *IEEE Trans. Pattern Anal. Mach.*, vol. 22, no. 1, pp. 4–37, Jan. 2000.
- [2] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. New York: Wiley, 2001.
- [3] J. A. Richards and X. Jia, *Remote Sensing Digital Image Analysis: An Introduction*. Berlin, Germany: Springer-Verlag, 1999.
- [4] Y. Li, F.A. Wessels, D. De Ridder, D., and M. J. T. Reinders, "Classification in the presence of class noise using a probabilistic kernel Fisher method," *Pattern Recognit.*, vol. 40, no. 12, pp. 3349–3357, Dec. 2007.

- [5] D. R. Wilson, "Asymptotic properties of nearest rules using edited data," *IEEE Trans. Syst., Man, Cybern.*, vol. 2, no. 3, pp. 408–421, Jul. 1972.
- [6] L. A. Breslow and D. Aha, "Simplifying decision trees: a survey," *Knowl. Eng. Rev.*, vol. 12, no. 1, pp. 1–40, Jan. 1997.
- [7] C. E. Brodley and M. A. Friedl, "Identifying mislabeled training data," *J. Artif. Intell. Res.*, vol. 11, pp. 131–167, 1999.
- [8] F. Muhlenbach, S. Lallich, and D. A. Zighed, "Identifying and handling mislabelled instances," *Journal of Intelligent Information Systems*, vol. 22, no. 1, pp. 89–109, 2004.
- [9] U. Rebbapragada and C. E. Brodley, "Class noise mitigation through instance weighting," in *Proc. ECML*, Warsaw, Poland, Sep. 2007, pp. 708–715.
- [10] H. Valizadegan, and P.-N. Tan, "Kernel based detection of mislabeled training examples," in *Proc. SIAM International Conference on Data Mining*, 2007, pp. 309–319.
- [11] J.-W. Sun, F.-Y. Zhao, C.-J. Wang, and S.-F. Chen, "Identifying and correcting mislabeled training instances," in *Proc. Future Generation Communication and Networking*, 2007, pp. 244–250.
- [12] N. Ghoggali and F. Melgani, "Automatic ground-truth validation with genetic algorithms for multispectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 7, pp. 2172–2181, Jul. 2009.
- [13] D. Yarowsky, "Unsupervised word sense disambiguation rivaling supervised methods," in *Proc. Annu. Meeting Assoc. Comput. Linguistics*, 1995, pp. 189–196.
- [14] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," in *Proc. COLT*, 1998, pp. 92–100.
- [15] Q. Jackson and D. A. Landgrebe, "An adaptive classifier design for high-dimensional data analysis with a limited training data set," *IEEE Trans. Geosci. Remote Sens.*, vol. 39, no. 12, pp. 2664–2679, Dec. 2001.
- [16] A. Palau, F. Melgani, and S. B. Serpico, "Cell algorithms with data inflation for non-parametric classification," *Pattern Recognit. Lett.*, vol. 27, no. 7, pp. 781–790, May 2006.
- [17] V. Vapnik, *Statistical Learning Theory*. New York: Wiley, 1998.
- [18] T. Joachims, "Transductive inference for text classification using support vector machines," in *Proc. ICML*, 1999, pp. 200–209.
- [19] L. Bruzzone, M. Chi, and M. Marconcini, "A novel transductive SVM for semisupervised classification of remote-sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 11, pp. 3363–3373, Nov. 2006.
- [20] G. Camps-Valls, T. V. Bandos Marcheva and D. Zhou, "Semi-supervised graph-based hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 10, pp. 3044–3054, Oct. 2007.
- [21] N. Ghoggali, F. Melgani, and Y. Bazi, "A multiobjective genetic SVM approach for classification problems with limited training samples," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 6, pp. 1707–1718, Jun. 2009.
- [22] Z.-H. Zhou and M. Li, "Semi-supervised regression with co-training," in *Proc. 19th Int. Joint Conf. Artif. Intell.*, 2005, pp. 908–913.
- [23] Y. Bazi and F. Melgani, "Semisupervised PSO-SVM regression for biophysical parameter estimation," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 6, pp. 1887–1895, Jun. 2007.
- [24] H. S. Seung, M. Opper, and H. Sompolinski, "Query by committee," in *Proc. Annu. Workshop Comput. Learn. Theory*, 1992, pp. 287–294.
- [25] P. Mitra, C. A. Murthy, and S. K. Pal, "A probabilistic active support vector learning algorithm," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 3, pp. 413–418, Mar. 2004.
- [26] J.-M. Park, "Convergence and application of online active sampling using orthogonal pillar vectors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 9, pp. 1197–1207, Sep. 2004.
- [27] M. Li and I. K. Sethi, "Confidence-based active learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 8, pp. 1251–1261, Aug. 2006.
- [28] S.-S. Ho and H. Wechsler, "Query by transduction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 9, pp. 1557–1571, Sep. 2008.
- [29] G.-J. Qi, X.-S. Hua, J. Tang, and H.-J. Zhang, "Two-dimensional multi-label active learning with an efficient online adaption model for image classification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 10, pp. 1880–1897, Oct. 2009.



- [30] D. Cohn, Z. Ghahramani, and M. Jordan, "Active learning with statistical models," *J. Artif. Intell. Res.*, vol. 4, pp. 129–145, Mar. 1996.
- [31] K. Fukumizu, "Statistical active learning in multilayer perceptrons," *IEEE Trans. Neural Netw.*, vol. 11, no. 1, pp. 17–26, Jan. 2000.
- [32] R. Castro, R. Willett, and R. Nowak, "Faster rates in regression via active learning," *Adv. Neural Inf. Process. Syst.*, vol. 18, pp. 179–186, 2006.
- [33] M. Sugiyama, "Active learning in approximately linear regression based on conditional expectation of generalization error," *The Journal of Machine Learning Research*, vol. 7, pp. 141–166, Jan. 2006.
- [34] R. Burbidge, J. J. Rowland, and R. D. King, "Active learning for regression based on query by committee," *Intelligent Data Engineering and Automated Learning*, pp. 209–218, 2007.
- [35] M. Sugiyama and N. Rubens, "A batch ensemble approach to active learning with model selection," *Neural Networks*, vol. 21, no. 9, pp. 1278–1286, Nov. 2008.
- [36] M. Sugiyama and S. Nakajima, "Pool-based active learning in approximate linear regression," *Mach. Learn.*, vol. 75, no. 3, pp. 249–274, Jan. 2009.
- [37] J. Paisley, X. Liao, and L. Carin, "Active learning and basis selection for kernel-based linear models: a Bayesian perspective," *IEEE Trans. Signal Process.*, vol. 58, no. 5, pp. 2686–2700, May 2010.



## 2. Active Learning Methods for Electrocardiographic Signal Classification

*Abstract* – In this chapter, we present three active learning strategies for the classification of electrocardiographic (ECG) signals. Starting from a small and suboptimal training set, these learning strategies select additional beat samples from a large set of unlabeled data. These samples are labeled manually, and then added to the training set. The entire procedure is iterated until constructing a final training set representative of the considered classification problem. The proposed methods are based on support vector machine classification and on the: 1) margin sampling; 2) posterior probability; and 3) query by committee principles, respectively. To illustrate their performance, we conducted an experimental study based on both simulated data and real ECG signals from the MIT-BIH arrhythmia database. In general, the obtained results show that the proposed strategies exhibit a promising capability to select samples that are significant for the classification process, i.e., to boost the accuracy of the classification process while minimizing the number of involved labeled samples.

The work presented in this chapter has been published in the *IEEE Trans. Inf. Techn. Biomed.*, vol. 14, no. 6, pp. 1405–1416, November 2010; Co-author: F. Melgani.

## 2.1. Introduction

Electrocardiographic (ECG) signals represent a useful information source about the rhythm and functioning of the heart. For this reason, in the last years, there has been a great interest in developing techniques for the automatic analysis of ECG signals. In particular, in the biomedical engineering community, automatic ECG signal classification has received a significant attention because of the practical advantages it offers for the detection and monitoring of cardiac diseases.

In the literature, there are several techniques dealing with this issue. Among the most recently published works, we can find those presented in [1]-[10]. In greater detail, Osowski *et al.* [1] implemented two classification systems based on the support vector machine (SVM) approach. The first exploits features based on high-order statistics, while the second uses the coefficients of Hermite polynomials. For improved performance, Osowski *et al.* proposed to combine the two classifiers by means of a weighting mechanism, whose weights are determined according to a least square estimation method. In [2], an automatic online beat segmentation and classification system based on a Markovian approach is proposed. The system carries out ECG signal analysis through two processing layers. In the first, the ECG signal is segmented into beat waveforms by means of a robust waveform modelling with hidden Markov models. In the second, the system identifies premature ventricular contraction beats using a simple set of rules. In [3], a rule-based rough-set decision system is presented for the development of an inference engine for disease identification using time-domain features. In [4], a patient-adapting heartbeat classifier system based on linear discriminants is proposed. The classification system processes an incoming recording with a global classifier to produce the first set of beat annotations. Then, an expert validates, and if necessary, corrects a fraction of the beats of the recording. The system then adapts by first training a local classifier using the newly annotated beats, and combines both local and global classifiers to form an adapted classification system. Inan *et al.* [5] presented an approach for classifying beats of a large data set by training a neural network (NN) classifier using wavelet and timing features. Inan *et al.* found that the fourth scale of a dyadic wavelet transform with a quadratic spline wavelet together with the pre-/post RR-interval ratio is effective for distinguishing normal and premature ventricular contraction (PVC) from other beats. In [6], an approach for personalized ECG heartbeat pattern classification is presented. It is based on block-based NNs, where a 2-D array of modular component NNs with flexible structures and internal configurations is implemented using reconfigurable digital hardware. Network structure and connection weights are optimized using local gradient-based search and evolutionary operators with the rates changing adaptively according to their effectiveness in the earlier evolution period. Wen *et al.* [7] proposed GreyART, an adaptive resonance theory NN based on the grey relational grade similarity measure for ECG beat classification. The strategy is subdivided in two phases. The first phase is the offline learning phase in which an optimal value for the vigilance threshold and the corresponding cluster centers from the learning results are determined. These results are used as initial settings of the online examining phase in which all ECG beats that pass the vigilance test are classified in real time. For those beats that fail the vigilance test, the classifier online creates new clusters and reports their templates for investigation by an expert. In [8], the generalization capability of the SVM classifier in the classification of ECG beats is improved by a classification system based on particle swarm optimization. For this purpose, the SVM classifier design is optimized by searching for the best value of the parameters that tune its function and by looking for the best subset of features that feed the classifier. Ince *et al.* [9] presented a patient-specific classification system for the detection of ECG heartbeat patterns. The process of feature extraction uses morphological wavelet transform features and temporal features from the ECG data. For the classification step, feedforward and fully connected artificial NNs, which are designed for each patient by the proposed multidimensional PSO technique, are used. In [10], heartbeat time series are classified using the SVM. Statistical methods and signal analysis techniques are used to extract features from the signals.

In general, in order to obtain an efficient and robust ECG classification system, it is necessary to address some important issues in a suitable way. One of them is the choice of the classifier to adopt. In particular, approaches based on SVM have shown great potential in many different research areas and in the ECG classification field too [1], [8], [10]. Indeed, the SVM classifier has a good generalization capability and is less sensitive to the curse of dimensionality than traditional classification techniques [11]. Classification systems based on SVM can give excellent performances, but are supervised. For this reason, the performances depend strongly on the quality and quantity of the labeled data used to train the classifier. Indeed, training (labeled) samples must be representative of the statistical distribution of the data. The process of the collection of training samples is, however complex, i.e., subject to errors and costly in terms of both time and money because done manually by human experts (cardiologists). To overcome this problem, it would be necessary to find a way to choose few training samples, but fundamental for the correct discrimination between the set of considered classes. For this reason, in the last few years, there has been a growing interest in developing strategies for the (semi)automatic construction of the set of training samples.

In the machine learning field, a recent approach focused on this topic is the so-called active learning approach. In general, its principle is relatively simple. Starting from a very small and suboptimal training set, any active learning strategy aims at selecting in some way additional samples, considered important, from a large amount of unlabeled data (learning set). These samples are labeled by the expert and then added to the training set. The entire procedure is iterated until a stopping criterion is satisfied.

In the literature, several active learning methods have been proposed. Mitra *et al.* [12] presented a probabilistic active learning strategy based on SVM designed for large data applications. Their strategy queries for a set of samples according to a distribution as determined by the current separating hyperplane and an adaptive confidence factor. The confidence factor is estimated from local information using the  $k$ -nearest neighbor principle. In [13], the active sampling-at-the-boundary method is applied using orthogonal pillar vectors lying on the decision boundary to learn the classification decision hyperplane in a multidimensional space. This shows that the proposed strategy can be applied with fewer training examples, rather than randomly selecting training data near the decision hyperplane. Both perceptron algorithm and SVM are used to estimate the decision boundary. Li and Sethi [14] proposed the method called the confidence-based active learning for training a wide range of classifiers such as SVM, NNs, and Naive Bayesians. The approach selects and requests annotation only for uncertain samples, i.e., for those samples that cannot be classified within a certain conditional error. Thus, it estimates the uncertainty value for each input sample according to its output score from a classifier and select only samples with uncertainty value above a user-defined threshold. A dynamic bin width allocation method is proposed to estimate sample conditional error. In [15], the query-by-transduction algorithm is proposed. It is based on  $p$ -values obtained from a transductive learning procedure in a stream-based setting, where examples are observed sequentially. When a new example is observed, different classifiers are constructed and statistical information is derived by considering all the possible labels for the new example. Then, statistical information of the two most likely labels for the new example is used to decide on whether to select the new example. The utility of the proposed method is shown on both binary and multiclass classification problems using SVM as classifier. In [16], active learning is applied to the multilabel image classification problem. Qi *et al.* proposed a 2-D strategy in which both the sample and the label dimensions are considered. The reason is that the contributions of different labels to minimize the classification error are different due to the inherent label correlations.

In the current literature, despite the great potential of the active learning solution, very scarce attention has been paid for developing methods derived from this very recent learning approach and applied to the problem of ECG signal classification. Merkwirth *et al.* [17] proposed an approach for regression modelling applied to ECG data, which can be used for data compression and prediction. A sequence of models on small subsets of the entire data set is trained in order to achieve small computation time and memory consumption.

An active learning approach is used to increase the training subset iteratively to cover the full dynamics of the data set without using all observations for the actual training.

In this chapter, we present different active learning strategies for ECG signal classification. All the proposed strategies are based on iterative procedures and use SVM to classify the signals. In particular, three different strategies are described and compared: 1) margin sampling (MS) in which the samples of the learning set more close to the hyperplanes between the different classes are chosen; 2) posterior probability sampling (PPS) in which posterior probabilities are estimated for each class. Then the samples that maximize the entropy between the posterior probabilities are selected; and 3) query by committee (QBC) in which a pool of classifiers is trained on different features to label the set of learning samples. Then, the algorithm chooses the samples with the maximum disagreement between classifiers.

The remaining part of the chapter is organized as follows. The basic mathematical formulation of SVMs for solving binary and multiclass classification problem is recalled in Section 2.2. In Section 2.3., the three active learning algorithms proposed in this study are described. Section 2.4. presents the results obtained on simulated data, while experiments on real ECG data from the MIT-BIH arrhythmia database [18] are shown in Section 2.5. Finally, conclusions are drawn in Section 2.6.

## 2.2. Support Vector Machine Classification

Let us first consider, for simplicity, a supervised binary classification problem. Let us assume that the training set consists of  $n$  vectors  $\mathbf{x}_i \in \mathcal{R}^d$  ( $i = 1, 2, \dots, n$ ) from the  $d$ -dimensional feature space  $X$ , generated from a set of morphological/temporal characteristics of the ECG beat. To each vector  $x_i$ , we associate a target  $y_i \in \{-1, +1\}$  (e.g., normal and abnormal beats). The linear SVM classification approach consists of looking for a separation between the two classes in  $X$  by means of an optimal hyperplane that maximizes the separating margin [11], [19]-[22]. In the nonlinear case, which is the most commonly used as data are often linearly nonseparable, the two classes are first mapped with a kernel method in a higher dimensional feature space, i.e.,  $\Phi(X) \in \mathcal{R}^{d'}$  ( $d' > d$ ). The membership decision rule is based on the function  $\text{sign}[f(x)]$ , where  $f(x)$  represents the discriminant function associated with the hyperplane in the transformed space and is defined as:

$$f(x) = \mathbf{w}^* \cdot \Phi(x) + b^*. \quad (2.1)$$

The optimal hyperplane defined by the weight vector  $\mathbf{w}^* \in \mathcal{R}^{d'}$  and the bias  $b^* \in \mathcal{R}$  is the one that minimizes a cost function that expresses a combination of two criteria: margin maximization and error minimization. It is expressed as [11]:

$$\Psi(\mathbf{w}, \xi) = \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^n \xi_i. \quad (2.2)$$

This cost function minimization is subject to the following constraints

$$y_i(\mathbf{w} \cdot \Phi(\mathbf{x}_i) + b) \geq 1 - \xi_i, \quad i = 1, 2, \dots, n \quad (2.3)$$

and

$$\xi_i \geq 0, \quad i = 1, 2, \dots, n \quad (2.4)$$

where  $\xi_i$ 's are the slack variables introduced to account for nonseparable data. The constant  $C$  represents a regularization parameter that allows to control the shape of the discriminant function. The aforementioned optimization problem can be reformulated through a Lagrange functional, for which the Lagrange multipliers can be found by means of a dual optimization leading to a quadratic programming solution [11], i.e.,

$$\max_{\alpha} \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) \quad (2.5)$$

under the constraints

$$C \geq \alpha_i \geq 0, \quad \text{for } i = 1, 2, \dots, n \quad (2.6)$$

and

$$\sum_{i=1}^n \alpha_i y_i = 0 \quad (2.7)$$

where  $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \dots, \alpha_n]$  is the vector of Lagrange multipliers and  $K(\cdot, \cdot)$  is a kernel function. The final result is a discriminant function conveniently expressed as a function of the data in the original (lower) dimensional feature space  $X$

$$f(\mathbf{x}) = \sum_{i \in S} \alpha_i^* y_i K(\mathbf{x}_i, \mathbf{x}) + b^* \quad (2.8)$$

The set  $S$  is a subset of the indices  $1, 2, \dots, n$  corresponding to the nonzero Lagrange multipliers  $\alpha_i$ , which define the so-called support vectors (SVs). The kernel  $K(\cdot, \cdot)$  must satisfy the condition stated in Mercer's theorem so as to correspond to some type of inner product in the transformed (higher) dimensional feature space  $\Phi(X)$  [11]. A typical example of such kernels is represented by the following Gaussian function:

$$K(\mathbf{x}_i, \mathbf{x}) = \exp\left(-\gamma \|\mathbf{x}_i - \mathbf{x}\|^2\right) \quad (2.9)$$

where  $\gamma$  represents a parameter inversely proportional to the width of the Gaussian kernel.

As described earlier, SVMs are intrinsically binary classifiers. But the classification of ECG signals often involves the simultaneous discrimination of numerous information classes. In order to face this issue, a number of multiclass classification strategies can be adopted [20], [21]. The most popular ones are the one-against-all (OAA) and the one-against-one (OAO) strategies. The former involves a reduced number of binary decompositions (and thus of SVMs), which are, however, more complex. The latter requires a shorter training time, but may incur conflicts between classes due to the nature of the score function used for decision. Both strategies generally lead to similar results in terms of classification accuracy. In this chapter, we shall consider the OAO strategy. Briefly, this strategy is based on the following procedure. Let  $\Omega = \{\omega_1, \omega_2, \dots, \omega_T\}$  be the set of  $T$  possible labels (information classes) associated with the ECG beats we desire to classify. First, an ensemble of  $T(T-1)/2$  (parallel) SVM classifiers is trained. Each classifier aims at solving a binary classification problem defined by the discrimination between one information class  $\omega_i$  ( $i = 1, 2, \dots, T$ ) against another information class  $\omega_j$  ( $j = 1, 2, \dots, T$ ) ( $i \neq j$ ). Then, in the classification phase, in order to decide which label to assign to each beat, the class with the maximum number of votes is chosen.

## 2.3. Active Learning Methods

Let us consider a training set  $L$  of ECG data composed initially of  $n$  labeled samples. Each sample has  $d$  features and is represented by the vector of features  $\mathbf{x}_i = \mathbf{l}_i \in \mathcal{R}^d = [l_{i,1}, l_{i,2}, \dots, l_{i,d}]$  ( $i = 1, 2, \dots, n$ ) and the corresponding label  $y_i$ .  $y_i$  assumes one of  $T$  discrete values, where  $T$  is the number of classes. We consider an additional learning set  $U$ , composed of  $m$  unlabeled samples  $\mathbf{x}_j = \mathbf{u}_j \in \mathcal{R}^d = [u_{j,1}, u_{j,2}, \dots, u_{j,d}]$  ( $j = 1, 2, \dots, m$ ), with  $m \gg n$ .

In order to augment the training set  $L$  with a series of examples chosen from the learning set  $U$  and labeled manually by the expert, an active learning algorithm has the task of choosing them properly so that to maximize the accuracy of the classification process while minimizing the number of active learning samples to label (i.e., number of interactions with the expert). For this purpose, in the remaining part of this section, we present three different active learning methods based on the SVM classifier.

### 2.3.1. Margin Sampling

MS is a simple active learning algorithm proposed specifically for classification problems based on SVM [23]. Considering a simple binary case with linearly separable classes, SVs are the samples of the

training set  $L$  more close to the hyperplane that describes the decision boundary given by the SVM classifier. If we consider the (unlabeled) learning set  $U$ , we can assume that the samples more close to the decision boundary are the most interesting samples, because they have a larger probability to become SVs in the new training set. Therefore, according to MS, the samples to select are the ones characterized by the minimum absolute values of the discriminant function. The same reasoning is applied in case of nonlinearly separable classes.

The assumptions done in the binary case can be used in a multiclass classification problem too. In this context, a solution is given in [24] in which a OAA SVM classifier is adopted. For each sample, the maximum value among the discriminant functions provided by the  $T$  binary classifiers is exploited as a sample indicator. Then, the samples with the minimum indicator values are selected, manually labeled and added to the training set.

In this work, we present an alternative solution based on the OAO SVM classifier, in which  $T(T-1)/2$  binary classifiers are involved. For each sample  $\mathbf{u}_j$  ( $j = 1, 2, \dots, m$ ), we calculate the number of votes of each class  $\mathbf{v}_j \in N^T = [v_{j,1}, v_{j,2}, \dots, v_{j,T}]$ . The class  $\omega_{MAX,j}$  with the largest number of votes  $v_{MAX,j}$  is first identified. Then, considering the  $T-1$  classifiers associated with the class  $\omega_{MAX,j}$ , the minimum absolute value of the discriminant function  $f_{MIN,j}$  is calculated. Finally, the samples characterized by the minimum values of  $v_{MAX,j}$  are selected, labeled, and added to the training set. In case of tie, i.e., several samples have the same value of  $v_{MAX,j}$ , those with the minimum values of  $f_{MIN,j}$  are chosen.

In the following, we describe the different phases on which is based the proposed MS method.

#### 1) Initialization

*Step 1:* Consider the initial training set  $L$ , composed of  $n$  labeled samples of  $T$  different classes.

*Step 2:* Consider the learning set  $U$ , composed of  $m$  ( $m \gg n$ ) unlabeled samples.

*Step 3:* Set  $N_s$  the number of samples to add at every iteration of the active learning process.

#### 2) MS active learning process

*Step 1:* Train a SVM classifier with the training set  $L$ , while estimating its free parameters by cross-validation (CV).

*Step 2:* For each sample  $\mathbf{u}_j$  ( $j = 1, 2, \dots, m$ ) of the learning set  $U$ , compute the maximum number of votes  $v_{MAX,j}$  and the minimum discriminative function value  $f_{MIN,j}$  as follows:

a) Calculate the discriminant function values  $\mathbf{f}_j$  for each binary SVM classifier.

b) Count the number of votes of each class  $\mathbf{v}_j$ .

c) Identify the class  $\omega_{MAX,j}$  with the maximum number of votes  $v_{MAX,j}$ . Let  $f_{MIN,j}$  be the minimum absolute value of the discriminative function associated with  $\omega_{MAX,j}$ .

*Step 3:* Select and label the  $N_s$  samples exhibiting the minimum values of  $v_{MAX,j}$  (and, if necessary, of  $f_{MIN,j}$ ).

*Step 4:* Add the  $N_s$  selected samples to the training set  $L$  and remove them from  $U$ .

3) *Convergence check:* Return to *Phase 2*, if the predefined convergence condition is not satisfied (e.g., the total number of samples to add to the training set is not yet reached).

### 2.3.2. Posterior Probability Sampling

Another active learning strategy (PPS) is the one based on the estimation of the posterior probability distribution of the classes  $p_k = P(y = c_k | \mathbf{u})$  ( $k = 1, 2, \dots, T$ ). After training the classifier using the training samples, the posterior probability of each class is estimated for each sample of the learning set  $U$ . In case of binary classification, we can guess that the best samples to select are those characterized by posterior probabilities close to 0.5, since they are those for which decision uncertainty is maximum. In multiclass problems, a more complex selection rule needs to be adopted. A solution is given in [25] in which samples that maximize the Kullback-Leibler divergence are selected and added to the training set. This kind of



selection rule can be used with any classifier that gives in output the estimate of the posterior probabilities. SVM is not a probabilistic classification approach and thus it does not directly yield in output probabilistic quantities. However, in the literature, some solutions have been proposed to infer posterior probability estimates from discriminant function values provided by SVMs.

In this study, the posterior probabilities are estimated using the strategy presented in [26]. First, the multiclass classification problem is decomposed into several binary classification problems using the OAO approach. For each couple of classes  $(\omega_k, \omega_t)$  ( $k = 1, 2, \dots, T$ ), ( $t = 1, 2, \dots, T$ ), ( $k \neq t$ ), we estimate the class probability  $r_{kt} = P(y = \omega_k | \mathbf{u}) = 1 - P(y = \omega_t | \mathbf{u})$  using the following relationship:

$$r_{kt} = \frac{1}{1 + e^{Af+B}} \quad (2.10)$$

where  $A$  and  $B$  are determined by minimizing the negative log-likelihood function using the training samples and their discriminant function values  $f$  generated through a CV process. At this point, the problem is how to estimate the posterior probabilities  $p_k = P(y = \omega_k | \mathbf{u})$  ( $k = 1, 2, \dots, T$ ) of the original multiclass problem. This issue is tackled through the following formulation:

$$\min_{\mathbf{p}} \frac{1}{2} \sum_{k=1}^T \sum_{t:t \neq k} (r_{tk} p_k - r_{kt} p_t)^2 \quad (2.11)$$

under the constraint

$$\sum_{k=1}^T p_k = 1, \quad p_k \geq 0, \quad \forall k. \quad (2.12)$$

Without reporting all the details, it can be demonstrated that the optimization problem in (2.11) and (2.12) has a unique solution and can be solved as a simple linear system [26].

After estimating posterior probabilities for all the samples of the active learning set  $U$ , an opportune sample selection strategy has to be adopted. For this purpose, we calculate for each sample the value of entropy  $H(\mathbf{u}_j)$

$$H(\mathbf{u}_j) = \sum_{k=1}^T -p_{k,j} \log(p_{k,j}) \quad (2.13)$$

where  $p_{k,j}$  is the posterior probability of  $\omega_k$  given sample  $\mathbf{u}_j$ . Then, the samples with the highest values of entropy are selected. Indeed, high values of entropy mean that the corresponding samples have been classified with low confidence, and thus adding them to the training set could improve the classifier decision regions in the feature space.

In the following, the different steps of the PPS method are summarized.

#### 1) Initialization

*Step 1:* Consider the initial training set  $L$ , composed of  $n$  labeled samples of  $T$  different classes.

*Step 2:* Consider the learning set  $U$ , composed of  $m$  unlabeled samples.

*Step 3:* Set  $N_s$  the number of samples to add at every iteration of the active learning process.

#### 2) PPS active learning process

*Step 1:* Train a SVM classifier with the training set  $L$ , while estimating its free parameters by CV.

*Step 2:* Classify the learning set  $U$  and calculate for each sample  $\mathbf{u}_j$  ( $j = 1, 2, \dots, m$ ) the posterior probability of each class  $p_{k,j}$  ( $k = 1, 2, \dots, T$ ).

*Step 3:* For each sample  $\mathbf{u}_j$ , calculate the entropy  $H(\mathbf{u}_j)$  associated with the estimated posterior probabilities.

*Step 4:* Select and label the  $N_s$  samples characterized by the maximum values of entropy  $H(\mathbf{u}_j)$ .

*Step 5:* Add the  $N_s$  selected samples to the training set  $L$  and remove them from  $U$ .

#### 3) Convergence check: Return to Phase 2, if the predefined convergence condition is not fulfilled.

### 2.3.3. Query by Committee

The QBC approach selects the learning samples to add to the training set using a committee of classifiers [27]. In particular, the samples with the maximum disagreement between the different classifiers are chosen. In the literature, different implementations and adaptations of this strategy have been proposed. In this study, we propose a simple strategy for addressing problems of multiclass active learning. Let  $s$  be an integer value greater than one and defining the feature sampling factor. Considering the original training set  $L$ , we construct  $s$  training subsets  $\{L_1, L_2, \dots, L_s\}$ , where  $L_g$  ( $g = 1, 2, \dots, s$ ) contains only the features  $f$  ( $f = 1, 2, \dots, d$ ) that satisfy the condition  $(f-1) \text{ module } (s) = g-1$ . The number of samples of each subset is equal to the original number of samples, but with a number of features reduced by a factor  $s$ . Similarly, from the original learning set  $U$ ,  $s$  learning subset  $\{U_1, U_2, \dots, U_s\}$  are constructed. At this point, each training subset is considered independently from each other and used to train an ensemble of  $c$  parallel SVM classifiers in which each classifier adopts a different kernel function to inject some diversity in the ensemble. Therefore, in total  $c \cdot s$  parallel classifiers are used. After the training phase, the learning samples are classified to estimate their labels. In particular,  $c \cdot s$  estimations are obtained for each sample. The entropy  $H(u_j)$  is calculated for each sample as follows:

$$H(\mathbf{u}_j) = \sum_{k=1}^T -rf_{k,j} \log(rf_{k,j}) \quad (2.14)$$

where  $rf_{k,j}$  is the relative frequency of class  $\omega_k$  for sample  $\mathbf{u}_j$ . As done in the PPS method, the samples with the maximum values of entropy, and thus characterized by the maximum disagreement between the classifiers, are selected and added to the training set.

Below, we describe the different phases of the QBC strategy.

#### 1) Initialization

*Step 1:* Consider the initial training set  $L$ , composed of  $n$  labeled samples of  $T$  different classes.

*Step 2:* Consider the learning set  $U$ , composed of  $m$  unlabeled samples.

*Step 3:* Set the feature sampling factor  $s$ .

*Step 4:* Construct the training subsets  $L_g$  ( $g = 1, 2, \dots, s$ ) and the learning subsets  $U_g$  ( $g = 1, 2, \dots, s$ ).

*Step 5:* Set the number of classifiers  $c$  to use in the ensemble for each training subset.

*Step 6:* Set  $N_s$  the number of samples to add at every iteration of the active learning process.

#### 2) QBC active learning process

*Step 1:* Train the  $c \cdot s$  SVM classifiers with the training subsets  $L_g$  ( $g = 1, 2, \dots, s$ ), while estimating their free parameters by CV.

*Step 2:* Classify the learning subsets  $U_g$  ( $g = 1, 2, \dots, s$ ) and calculate for each sample  $\mathbf{u}_j$  ( $j = 1, 2, \dots, m$ ) the number of occurrences of each class.

*Step 3:* For each sample  $\mathbf{u}_j$ , calculate the value of entropy  $H(\mathbf{u}_j)$  associated with the occurrences of the estimated class labels.

*Step 4:* Select and label the  $N_s$  samples characterized by the maximum values of entropy  $H(\mathbf{u}_j)$ .

*Step 5:* Add the  $N_s$  selected samples to the training set  $L$  and remove them from  $U$ .

3) *Convergence check:* Return to Phase 2, if the predefined convergence condition is not satisfied.

## 2.4. Experiments on Simulated Data

### 2.4.1. Data Set Description

To evaluate the performance of the proposed active learning strategies, we first conducted an experimental phase based on simulated data in order to better illustrate their properties. In particular, we considered the well-known chessboard problem, i.e., a 2-D multiclass problem with uniformly distributed

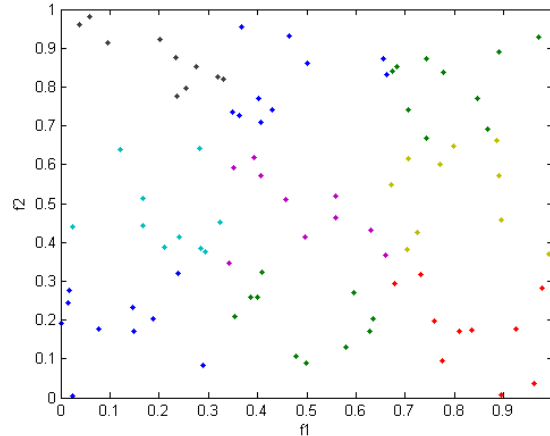


Fig. 2.1. Example of distribution of the 90 initial training samples (ten for each class) characterizing the chessboard classification problem.

classes. For such purpose, we generated a  $3 \times 3$  chessboard composed of nine uniformly distributed classes. The initial training set is shown in Fig. 2.1. The entire learning set  $U$  was composed of 27000 samples, i.e., 3000 samples for each class. The initial training set  $L$  contained 90 samples, i.e., 10 samples for each class. The algorithms were run until the number of training samples was equal to 2000, adding the ten most significant samples at each iteration. For the QBC method, the factor of feature sampling  $s$  and the number of parallel classifier  $c$  were set both to two. In particular classifiers with linear and radial basis function (RBF) kernels were used. The entire active learning process was run ten times, each with a different initial training set so that to yield statistically reliable results. At each run, the initial training samples were chosen in a completely random way.

Classification performance was evaluated in terms of overall accuracy (Acc), which is the percentage of correctly classified samples among all the considered samples, independently of the classes they belong to. For the performance evaluation, a test set of 18000 samples was considered.

A SVM classifier was also trained on the entire learning set (i.e., 27000 labeled samples) in order to have a reference training scenario, called “full” training. On the one hand, the classification results obtained in this way represent an upper bound for the accuracies. On the other, we expect that the lower accuracy bound will be given by the completely random selection strategy (R). We recall that the purpose of any active learning strategy is to converge to the performance of the “full” training scenario faster than the R method.

## 2.4.2. Experimental Results

For the “full” classifier, the Acc is equal to 99.68%. In Fig. 2.2(a), we show the Acc in function of the number of training samples for the three proposed active learning strategies and the random one. All the three active learning algorithms converge to the “full” accuracy using about 1500 training samples, which represent 5.6% of the entire learning set. We note that, before convergence, the MS method gives the best performance.

To better understand the behaviors of the proposed methods, in Fig. 2.2(b)-(c) we show the evolution at each iteration of the CV accuracy and the number of SVs (#SV). It is interesting to observe that the value of CV tends to decrease in the first iterations, while we have an increase of the CV value only when a sufficient number of samples have been added to the training set. The decrease of the CV value means that samples difficult to classify are added to the training set. However, these new samples are highly informative and thus allow improving the generalization performance (i.e., the accuracy on the test samples). A different behavior is obtained for the R strategy, for which the CV value tends to increase from the beginning. Analogously, we note that in our active learning methods the #SV value tends to increase faster than the R

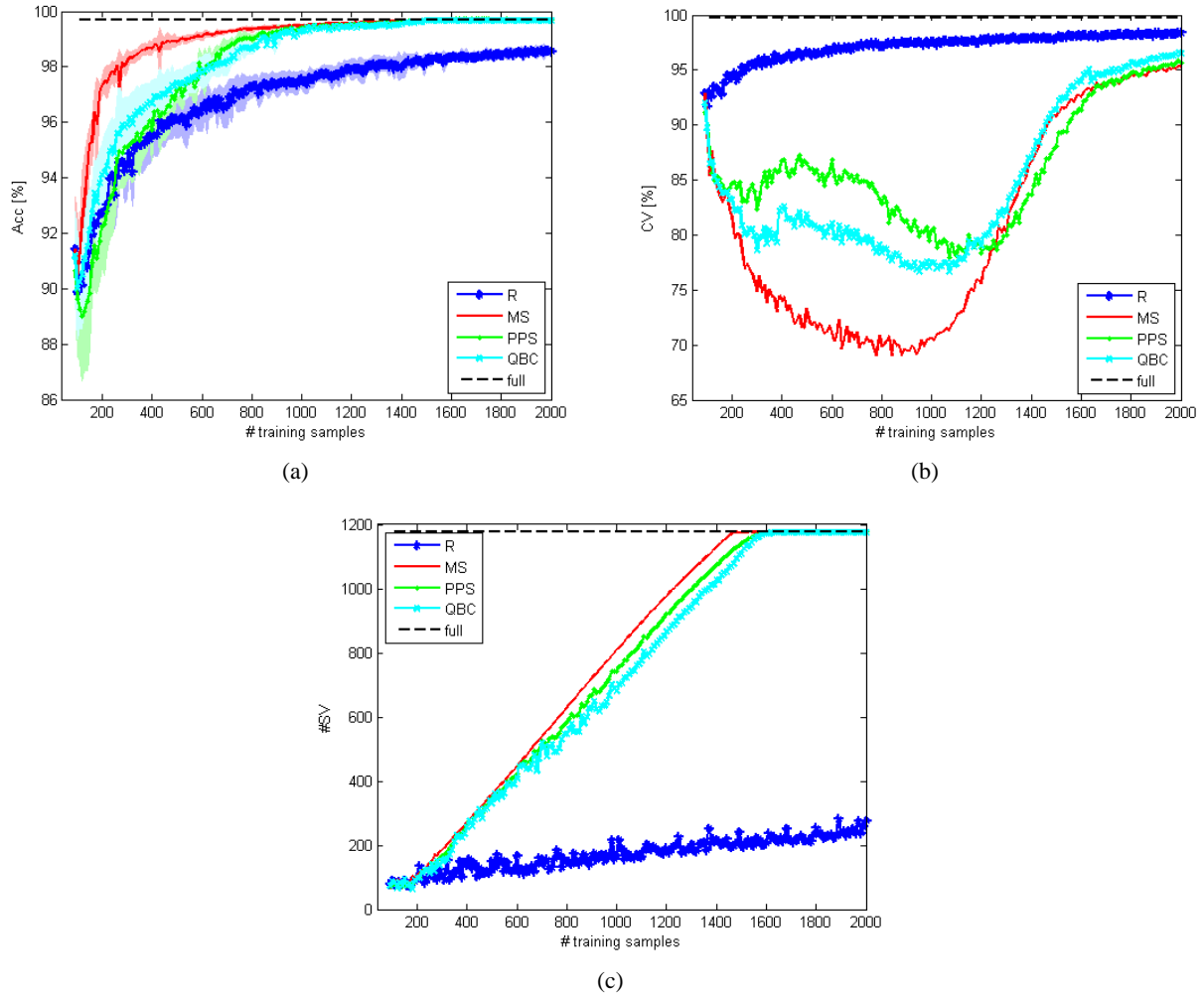


Fig. 2.2. Performances achieved on the chessboard classification problem in terms of (a) Acc, (b) CV accuracy, and (c) #SV. Each graph shows the results in function of the number of training samples and averaged over ten runs of the algorithm, each with a different initial set. The shaded areas in (a) show the standard deviation of the Acc over the ten considered runs. R = random, MS = margin sampling, PPS = posterior probability sampling, QBC = query by committee, full = full SVM.

method. At convergence, i.e., for about 1500 training samples, the #SV value for the active learning strategies is equal to 1176, which correspond to the number of SVs for the “full” classifier. Therefore, about 80% of the samples selected by the active learning methods are SVs, and hence, are important for the discrimination among the nine classes. We observe that this behavior is not verified for the R method, for which the number of SVs tends to increment much slower. The fast increment of the number of SVs for the active learning strategies shows clearly that the samples added to the training set are really important for the classification process.

The obtained results are shown in greater detail in Table 2.I. In particular, we report the values of Acc, standard deviation associated with the Acc  $\sigma_{\text{Acc}}$ , which is an indication of stability of the method, CV accuracy and #SV. In bold, we highlight the best performance in terms of Acc and  $\sigma_{\text{Acc}}$  for each training set size.

In Fig. 2.3(a)-(d) we show the samples selected by the random and the proposed active learning strategies for a training set size equal to 1000. While the R method chooses the samples in a completely random way [see Fig. 2.3(a)], the active learning methodologies [see Fig. 2.3(b)-(d)] tend to select the samples that lie on the boundaries between classes. In this way, the algorithms focus more on difficult samples, while samples that belong to already well-classified areas are almost disregarded.

TABLE 2.I  
ACC AND CV ACCURACIES, STANDARD DEVIATION ( $\sigma_{Acc}$ ), AND #SV ACHIEVED  
ON THE CHESSBOARD CLASSIFICATION PROBLEM BY THE DIFFERENT  
INVESTIGATED LEARNING ALGORITHMS

Method	#training samples	Acc	$\sigma_{Acc}$	CV	#SV
Full	27000	99.68	-	99.75	1176
Initial	90	91.43	1.87	92.89	81
R	500	96.13	0.73	96.72	150
MS		<b>98.91</b>	<b>0.17</b>	73.30	358
PPS		97.17	0.39	86.50	344
QBC		97.40	0.49	81.90	330
R	1000	97.45	0.34	97.64	165
MS		<b>99.48</b>	<b>0.06</b>	70.62	808
PPS		99.39	0.09	80.40	743
QBC		99.30	0.11	77.51	679
R	1500	98.24	0.24	97.95	200
MS		99.68	<b>0.00</b>	90.79	1176
PPS		<b>99.72</b>	0.05	88.08	1141
QBC		99.68	0.07	91.43	1108

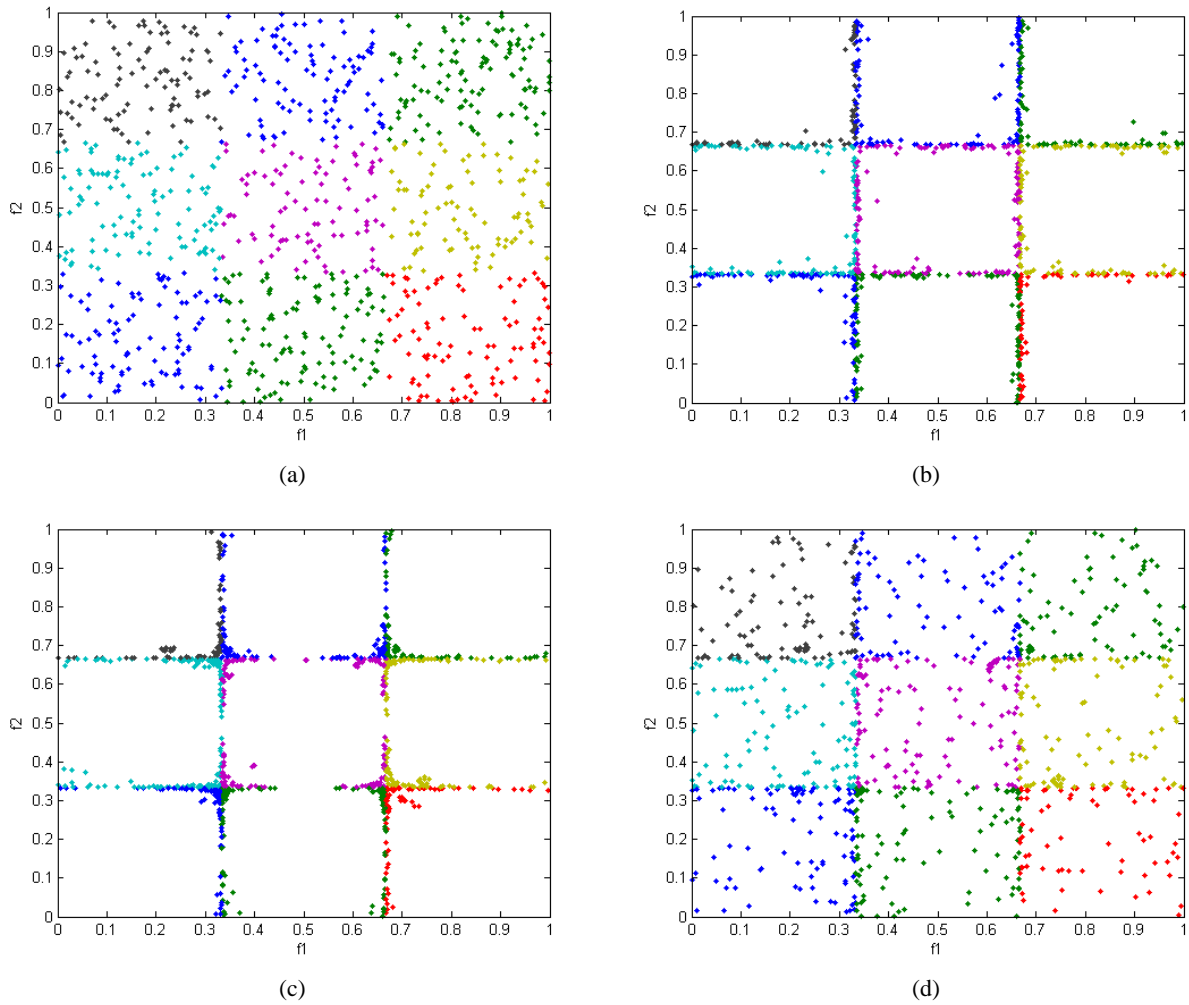


Fig. 2.3. Distribution of the 1000 training samples generated by each method for the chessboard classification problem. (a) R, (b) MS, (c) PPS, and (d) QBC methods.

## 2.5. Experiments on Real ECG Data

### 2.5.1. Data Set Description

In this experimental part, we completed the earlier assessment by considering this time real ECG data, obtained from the MIT-BIH arrhythmia database [18]. In particular, the considered beats refer to the following six classes: normal sinus rhythm (N), atrial premature beat (A), ventricular premature beat (V), right bundle branch block (RB), paced beat (/), and left bundle branch block (LB). The beats were selected from the recordings of 20 patients, which correspond to the following files: 100, 102, 104, 105, 106, 107, 118, 119, 200, 201, 202, 203, 205, 208, 209, 212, 213, 214, 215, and 217. In order to feed the classification process, in this work we adopted a subset of the features described in [4]. In particular, we used the two following kinds of features: 1) ECG morphological features and 2) three ECG temporal features, i.e., the QRS complex duration, the RR interval (the time span between two consecutive R points representing the distance between the QRS peaks of the present and previous beats), and the RR interval averaged over the ten last beats. In order to extract these features, first we performed the QRS detection and ECG wave boundary recognition tasks by means of the *ecgpuwave* software available on <http://www.physionet.org/physiotools/ecgpuwave/src/>. Then, after extracting the three temporal features of interest, we normalized to the same periodic length the duration of the segmented ECG cycles according to the procedure reported in [28]. To this purpose, the mean beat period was chosen as the normalized periodic length, which was represented by 300 uniformly distributed samples. Consequently, the total number of morphology and temporal features equals 303 for each beat.

Fig. 2.4. illustrates the distribution of the six considered classes drawn by means of 25 samples randomly selected for each class and the two best features according to the Principal Component Analysis (PCA) algorithm [29]. From this figure, one can expect that the discrimination task will not be straightforward due to the apparently strong overlap between classes.

In all the following experiments, all the available samples were randomly split in two sets, corresponding to learning  $U$  and test sets. The detailed numbers of learning and test beats are reported for each class in Table 2.II. In this table, we report the number of beats of the initial training set  $L$  for each class too. The initial training beats were selected randomly from the learning set  $U$ . At each iteration, the algorithms of active learning added the 50 most relevant samples up to reaching a total of 4000 training samples. For the QBC technique, the factor of feature sampling  $s$  was set to 3, while only the RBF kernel was used to train the classifiers. As done in the experiments on simulated data, the entire procedure was repeated ten times, each by choosing the initial training set in a completely random way in order to obtain statistically reliable results.

Similarly to [4], classification performance was evaluated in terms of several measures which are: 1) the Acc; 2) the specificity (Sp), which is the accuracy of class N; 3) the sensitivities (Se) of classes A, V, RB, /, LB, which represent the accuracy of each class; 4) the average accuracy (AvAcc), which is the average over the Sp and the five values of Se.

### 2.5.2. Experimental Results

The results achieved on the real ECG data agree with those obtained on the earlier chessboard classification problem. The “full” classifier is characterized by values of Acc and AvAcc equal to 98.35% and 95.58%, respectively. The evolution of the values of Acc and AvAcc in function of the training set size is shown in Fig. 2.5(a)-(b). From these plots, we observe that the proposed active learning methodologies tend to converge to the results given by the “full” classifier for a number of training samples equal to about 2500, which corresponds to 11.7% of the entire learning set.

As seen for the simulated data set, we note that the active learning methods are characterized by better

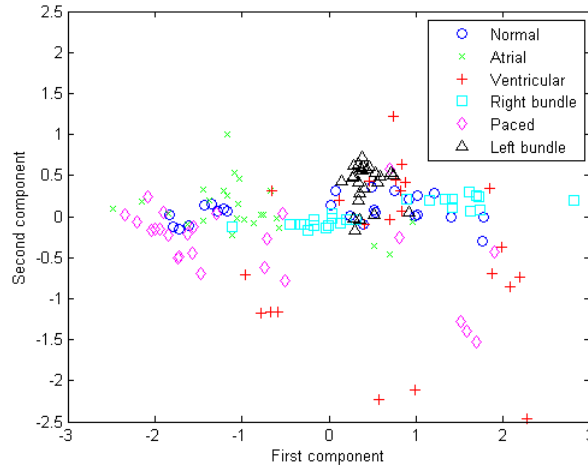


Fig. 2.4. Two-dimensional distribution of the six considered classes in the subspace formed by the best couple of features obtained with the PCA algorithm. For better visualization, just 25 samples were randomly selected for each class.

TABLE 2.II  
NUMBERS OF INITIAL TRAINING, LEARNING AND TEST BEATS USED IN THE EXPERIMENTS

Class	N	A	V	RB	/	LB	Total
Initial training beats	75	50	50	25	25	25	250
Learning beats	12338	344	2194	1982	3498	988	21344
Test beats	12337	344	2195	1982	3498	988	21344

CV and #SV trends with respect to the R sampling [see Fig. 2.5(c)-(d)]. In particular, for the first steps of the iterative process, we have a decrease of the value of CV and a faster increment of the #SV.

In Table 2.III, the results for specific sizes of the training set are reported. It is interesting to note that at convergence the MS and PPS methods give values of accuracies slightly better than the “full” classifier, since active learning aims also at reducing mislabeling risks as it involves significantly smaller numbers of samples to be labeled. Moreover, these methods appear more stable with respect to the R strategy, since characterized by smaller values of Acc and AvAcc standard deviations.

In terms of Sp and Se (see Table 2.III), the active learning strategies appear able to give better results with respect to the R sampling. Moreover, the accuracies at convergence are in some cases better than the “full” classification. In Fig. 2.5(e)-(f), we show the evolution of the number of selected samples and the Se for the atrial premature beat (A) class, which is the most difficult class to discriminate and the less represented in the learning set. At beginning, 20% of the training samples are associated with this class. We note that the percentage of selected samples is very high with respect to the prior probability of this class, which is less than 3%. As the training set size increases, the probability to select randomly a sample of this class becomes very low, and so the percentage of selected samples converges to the prior probability. The selection of few samples involves a decreasing of performance, which is highlighted by the decrease of the Se. Indeed, for a training set size equal to 2500, the Se for class A for the R method is equal to 69.56%. Focusing on the proposed active learning strategies, the iterative process is able to select a greater number of samples, despite their very limited availability. The selection of these samples allows obtaining a significant increment in terms of Se. For example, in the case of MS strategy and for a training size equal to 2500, the Se for class A is equal to 81.95%. Similar performances are achieved by the other active learning methods.

Another important goal of active learning approaches is to decrease the computational burden incurred by the classifier, while keeping the classification accuracies the highest possible. For this purpose, we considered for each active learning strategy the minimum number of training samples for which the Acc is less of at most 1% with respect to the Acc of the “full” classifier. In Table 2.IV, we report the training and

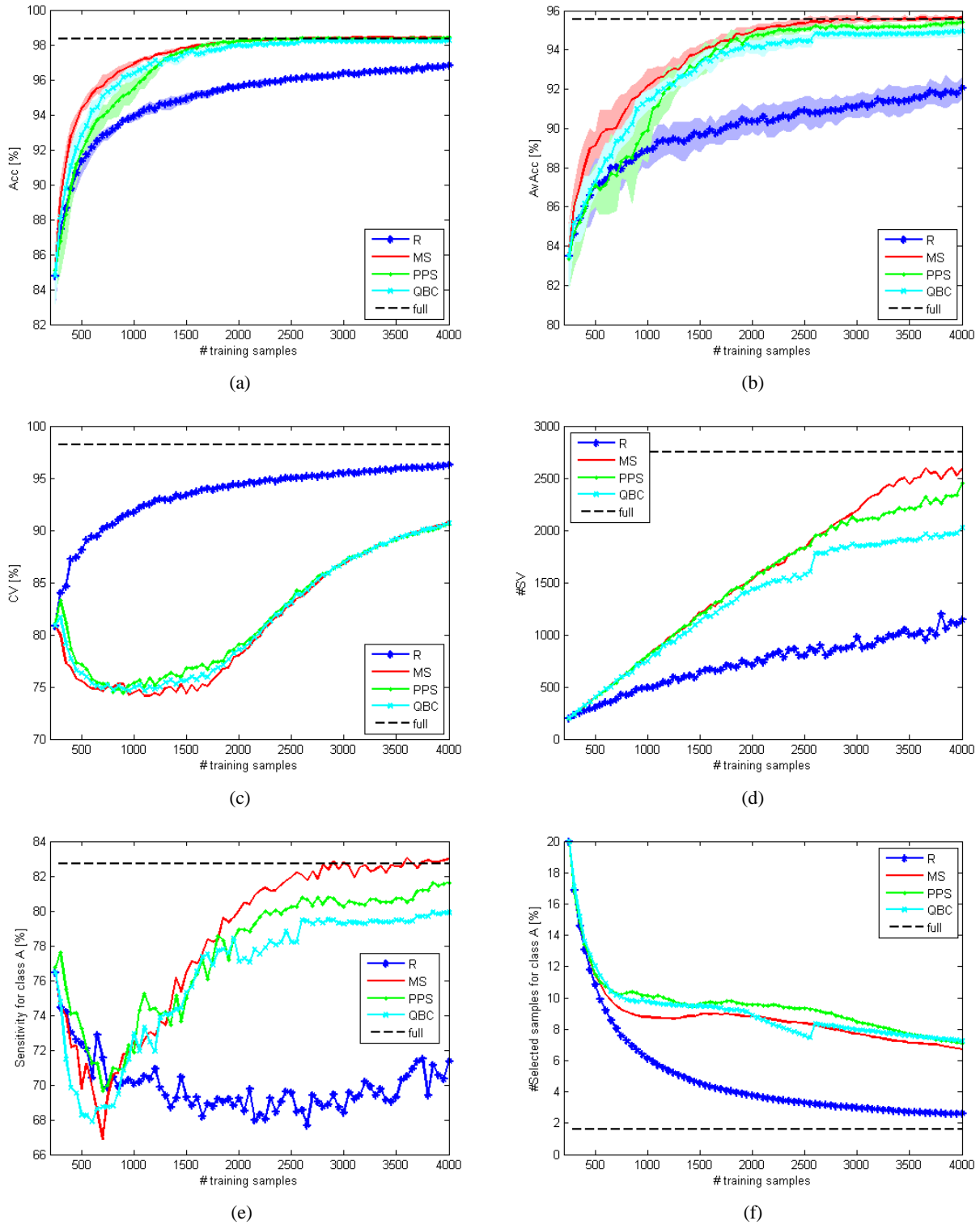


Fig. 2.5. Performances achieved on the ECG data set by the investigated learning methods in terms of: (a) Overall accuracy (Acc), (b) average accuracy (AvAcc), (c) cross-validation (CV) accuracy, (d) number of support vectors (#SV), and (e) sensitivity and (f) number of selected samples both for class A.

test times of the corresponding classifiers. As can be seen, active learning strategies are able to reduce significantly the computational time to train the classifier, together with a decreasing of the manual work for sample labelling. Analogously, using a smaller number of training samples leads to a decrease of the time for classifying unknown samples.



TABLE 2.III

ACC, AVACC AND CV ACCURACIES, SP, SE, STANDARD DEVIATIONS ( $\sigma$ ), AND #SV ACHIEVED ON THE TEST BEATS BY THE INVESTIGATED LEARNING ALGORITHMS

Method	#training samples	Acc	$\sigma_{Acc}$	AvAcc	$\sigma_{AvAcc}$	CV	#SV	Sp	Se				
									A	V	RB	/	LB
Full	21344	98.35	-	95.58	-	98.23	2749	98.98	82.72	95.07	98.59	99.56	98.54
Initial	250	84.79	1.68	83.46	1.62	80.80	201	86.74	76.45	81.32	92.13	75.68	88.46
R	1000	93.90	0.22	88.89	0.98	91.73	494	95.74	70.06	84.32	96.00	95.08	92.14
MS		<b>96.91</b>	<b>0.19</b>	<b>92.16</b>	0.86	74.51	799	<b>98.20</b>	72.53	<b>91.01</b>	<b>97.73</b>	<b>98.56</b>	<b>94.92</b>
PPS		95.49	0.59	89.91	1.67	75.38	800	97.58	71.89	88.94	97.25	95.64	88.15
QBC		96.36	0.29	91.44	<b>0.53</b>	74.95	748	97.88	<b>72.94</b>	90.23	97.24	97.73	92.61
R	2500	96.06	0.13	90.91	0.70	95.03	801	97.56	69.56	89.02	97.20	97.60	94.49
MS		<b>98.32</b>	0.09	<b>95.35</b>	0.28	82.82	1819	99.05	<b>81.95</b>	94.45	<b>98.78</b>	99.53	<b>98.36</b>
PPS		98.31	<b>0.07</b>	95.06	<b>0.16</b>	83.34	1829	<b>99.07</b>	80.23	<b>94.51</b>	98.77	<b>99.59</b>	98.18
QBC		98.07	0.08	94.44	0.33	83.08	1574	99.00	78.20	93.64	98.55	99.30	97.98
R	4000	96.82	0.13	92.06	0.53	96.26	1142	98.10	71.37	90.60	97.73	98.34	96.23
MS		<b>98.44</b>	<b>0.03</b>	<b>95.67</b>	0.14	90.81	2592	<b>99.11</b>	<b>82.99</b>	94.86	98.74	99.64	<b>98.68</b>
PPS		98.43	0.06	95.44	<b>0.10</b>	90.53	2448	99.10	81.63	<b>94.88</b>	<b>98.81</b>	<b>99.68</b>	98.52
QBC		98.25	0.08	94.95	0.25	90.67	2026	99.05	79.91	94.35	98.69	99.48	98.24

TABLE 2.IV

NUMBER OF TRAINING SAMPLES, ACC, AND TRAINING AND TEST TIMES FOR THE INVESTIGATED LEARNING ALGORITHMS

Method	# training samples	Acc	Training time [s]	Test time [s]
Full	21344	98.35	154.0	388.6
R	4000	96.82	14.0	162.8
MS	1200	97.42	3.7	137.8
PPS	1350	97.38	4.8	153.4
QBC	1500	97.50	5.9	170.2

### 2.5.3. Experiments on Unseen Recordings

To conclude the experimental assessment on real ECG data, we considered the remaining 28 recordings from the MIT-BIH arrhythmia database, which were not used to train the classifiers. These recordings, termed as “unseen” recordings, refer to the following files: 101, 103, 108, 109, 111, 112, 113, 114, 115, 116, 117, 121, 122, 123, 124, 207, 210, 219, 220, 221, 222, 223, 228, 230, 231, 232, 233, 234. The corresponding numbers of beats for each class are listed in Table 2.V. Such beats are useful to complete the test of the generalization capabilities of the active learning strategies.

In Fig. 2.6(a)-(b), we plot the evolution of the values of Acc and AvAcc versus the training set size, while in Table 2.VI we report the results for specific sizes of the training set. In general, the proposed active learning strategies exhibit relatively good generalization capabilities when tested on beats belonging to recordings completely new. Indeed, significantly better results both in terms of accuracy and standard deviation (thus stability) are obtained with respect to the standard R method. We note that a strong accuracy decrease is obtained in this set of experiments compared to the results presented in the earlier section in which training and test beats were extracted from the same recordings. This can be explained by the fact that morphological features are not enough robust to handle unseen recordings. Though more complex, other kinds of features such as those based on wavelets or high order statistics could be a good solution to increment the robustness of the classification process.

TABLE 2.V  
NUMBERS OF TEST BEATS IN THE UNSEEN RECORDINGS

Class	N	A	V	RB	/	LB	Total
Test beats	45201	1902	2688	3265	-	6069	59125

TABLE 2.VI  
ACC AND AVACC ACCURACIES, SP, SE, AND STANDARD DEVIATIONS ( $\sigma$ )  
ACHIEVED ON THE TEST BEATS BY THE INVESTIGATED LEARNING ALGORITHMS

Method	#training samples	Acc	$\sigma_{Acc}$	AvAcc	$\sigma_{AvAcc}$	Sp	Se				
							A	V	RB	/	LB
Full	21344	79.11	-	72.40	-	82.43	74.19	71.80	69.16	-	64.44
Initial	250	65.87	1.62	65.91	1.05	66.35	78.88	66.00	53.46	-	65.08
R	1000	72.91	1.06	65.41	1.08	76.30	71.85	61.81	54.00	-	63.10
MS		<b>77.32</b>	1.36	<b>71.53</b>	1.08	<b>80.23</b>	74.32	<b>68.86</b>	<b>69.78</b>	-	64.45
PPS		76.05	1.13	71.01	0.72	78.57	<b>75.28</b>	67.21	68.59	-	<b>65.39</b>
QBC		76.73	<b>0.91</b>	71.41	<b>0.70</b>	79.84	75.25	68.54	68.61	-	64.80
R	2500	76.07	1.07	68.88	0.95	79.49	71.69	66.41	64.10	-	62.69
MS		<b>79.07</b>	0.23	<b>72.95</b>	0.38	82.19	74.70	<b>71.43</b>	<b>72.07</b>	-	<b>64.37</b>
PPS		79.00	<b>0.13</b>	72.59	<b>0.25</b>	<b>82.25</b>	<b>75.26</b>	71.25	69.84	-	64.36
QBC		79.00	0.15	72.56	0.29	82.22	74.94	70.71	71.21	-	63.78
R	4000	75.18	0.86	67.52	0.81	78.76	71.21	66.72	58.27	-	62.62
MS		78.91	0.06	<b>72.33</b>	<b>0.20</b>	82.18	<b>74.42</b>	<b>70.63</b>	<b>70.09</b>	-	64.35
PPS		<b>79.13</b>	<b>0.05</b>	71.86	0.32	82.65	74.20	69.53	68.55	-	64.37
QBC		79.02	0.06	72.14	0.25	<b>82.69</b>	73.85	69.90	69.27	-	<b>64.98</b>

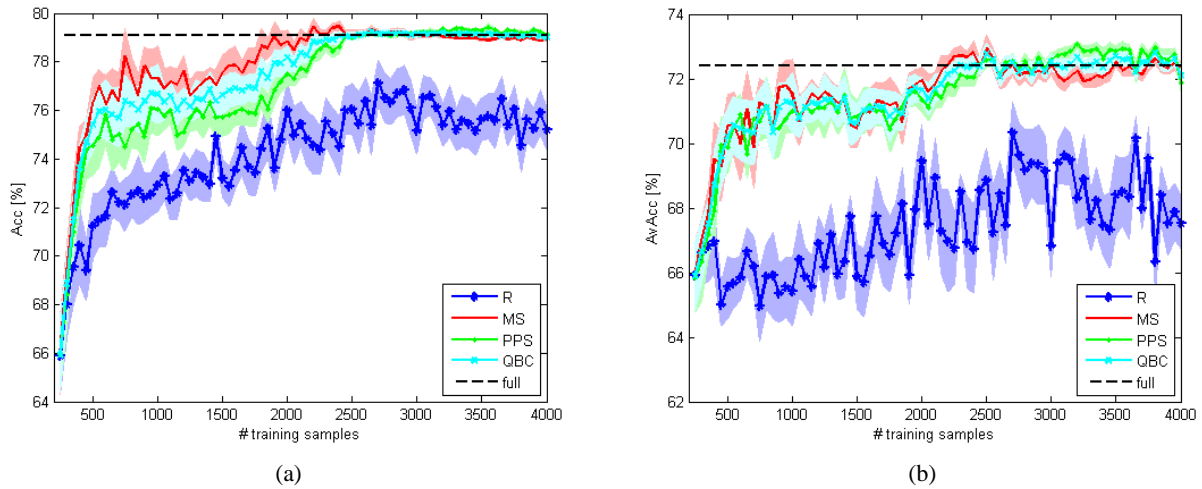


Fig. 2.6. Performances achieved on the ECG data set by the investigated learning methods in terms of: (a) Acc and (b) AvAcc.

## 2.6. Conclusion

In this chapter, three active learning strategies for the SVM classification of electrocardiogram (ECG) signals have been presented. Starting from a small and suboptimal training set, the strategies have the purpose to select from a large unlabeled data set the samples more significant for the classification process, i.e., those able to give high accuracies in terms of classification while minimizing the number of training samples and the computational costs required by the classifier.

The experimental results obtained on simulated and real ECG data show good capabilities of the proposed methods for selecting significant samples. In general, all the proposed methods are characterized by higher performance in terms of both accuracies and stability with respect to a completely random

selection strategy. Comparing them, the strategy based on the MS principle seems the best as it quickly selects the most informative samples. Another interesting result is that active learning methods are able to give accuracies slightly better than the “full” classifier, confirming their usefulness in reducing mislabeling risks.

While in this research the initial training set was chosen in a random way, we think that a more sophisticated initialization strategy could further improve the performance of the active learning process. Research is in progress in this direction. Finally, it is worth noting that, as shown in the literature, a further increase of the accuracies could be achieved by feeding the classifier with other kinds of features (e.g., those based on wavelets or high order statistics) together with or in substitution to the morphological ones.

## 2.7. Acknowledgment

The authors would like to thank C.-C. Chang and C.-J. Lin for supplying the software LIBSVM (<http://www.csie.ntu.edu.tw/~cjlin/libsvm>) used in this research.

## 2.8. References cited in Chapter 2

- [1] S. Osowski, T. H. Linh, and T. Markiewicz, “Support vector machine-based expert system for reliable heart beat recognition,” *IEEE Trans. Biomed. Eng.*, vol. 51, no. 4, pp. 582–589, Apr. 2004.
- [2] R.V. Andraeo, B. Dorizzi, and J. Boudy, “ECG signal analysis through hidden Markov models,” *IEEE Trans. Biomed. Eng.*, vol. 53, no. 8, pp. 1541–1549, Aug. 2006.
- [3] S. Mitra, M. Mitra, and B.B. Chaudhuri, “A rough set-based inference engine for ECG classification,” *IEEE Trans. Instrum. Meas.*, vol. 55, no. 6, pp. 2198–2206, Dec. 2006.
- [4] F. de Chazal and R.B. Reilly, “A patient adapting heart beat classifier using ECG morphology and heartbeat interval features,” *IEEE Trans. Biomed. Eng.*, vol. 53, no. 12, pp. 2535–2543, Dec. 2006.
- [5] T. Inan, L. Giovannardi, and J.T.A. Kovacs, “Robust neural network based classification of premature ventricular contractions using wavelet transform and timing interval features,” *IEEE Trans. Biomed. Eng.*, vol. 53, no. 12, pp. 2507–2515, Dec. 2006.
- [6] W. Jiang and S. G. Kong, “Block-based neural networks for personalized ECG signal classification,” *IEEE Trans. Neural Netw.*, vol. 18, no. 6, pp. 1750–1761, Nov. 2007.
- [7] C. Wen, M.-F. Yeh, and K.-C. Chang, “ECG beat classification using GreyART network,” *IET Signal Process.*, vol. 1, no. 1, pp. 19–28, Mar. 2007.
- [8] F. Melgani and Y. Bazi, “Classification of electrocardiogram signals with support vector machines and particle swarm optimization,” *IEEE Trans. Inf. Technol. Biomed.*, vol. 12, no. 5, pp. 667–677, Sep. 2008.
- [9] T. Ince, S. Kiranyaz, and M. Gabbouj, “A generic and robust system for automated patient-specific classification of ECG signals,” *IEEE Trans. Biomed. Eng.*, vol. 56, no. 5, pp. 1415–1426, May 2009.
- [10] A. Kampouraki, G. Manis, and C. Nikou, “Heartbeat time series classification with support vector machines,” *IEEE Trans. Inf. Technol. Biomed.*, vol. 13, no. 4, pp. 512–518, Jul. 2009.
- [11] V. Vapnik, *Statistical Learning Theory*. New York: Wiley, 1998.
- [12] P. Mitra, C. A. Murthy, and S. K. Pal, “A probabilistic active support vector learning algorithm,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 3, pp. 413–418, Mar. 2004.
- [13] J.-M. Park, “Convergence and application of online active sampling using orthogonal pillar vectors,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 9, pp. 1197–1207, Sep. 2004.
- [14] M. Li and I. K. Sethi, “Confidence-based active learning,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 8, pp. 1251–1261, Aug. 2006.
- [15] S.-S. Ho and H. Wechsler, “Query by transduction,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 9, pp. 1557–1571, Sep. 2008.
- [16] G.-J. Qi, X.-S. Hua, Y. Rui, J. Tang, and H.-J. Zhang, “Two-dimensional multi-label active learning with an efficient online adaption model for image classification,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 10, pp. 1880–1897, Oct. 2009.

- [17] C. Merkwirth, J.D. Wichard, and M.J. Ogorzalek, "Active subset selection approach to nonlinear modeling of ECG data," in *Proc. ISCAS*, Bangkok, Thailand, May 2003, vol. 3, pp. 758–761.
- [18] R. Mark and G. Moody. (1997). *MIT-BIH Arrhythmia Database* [Online]. Available: <http://ecg.mit.edu/dbinfo.html>.
- [19] E. Blanzieri and F. Melgani, "Nearest neighbor classification of remote sensing images with the maximal margin principle," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 6, pp. 1804–1811, Jun. 2008.
- [20] F. Melgani and L. Bruzzone, "Classification of hyperspectral remote sensing images with support vector machine," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 8, pp. 1778–1790, Aug. 2004.
- [21] C.-W. Hsu and C.-J. Lin, "A comparison of methods for multiclass support vector machines," *IEEE Trans. Neural Netw.*, vol. 13, no. 2, pp. 415–425, Mar. 2002.
- [22] A. Massa, A. Boni, and M. Donelli, "A classification approach based on SVM for electromagnetic sub-surface sensing," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 9, pp. 2084–2093, Sep. 2005.
- [23] G. Schohn and D. Cohn, "Less is more: Active learning with support vector machines," in *Proc. 17<sup>th</sup> ICML*, Stanford, CA, 2000, pp. 839–846.
- [24] D. Tuia, F. Ratle, F. Pacifici, M. F. Kanevski, and W. J. Emery, "Active learning methods for remote sensing image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 7, pp. 2218–2232, Jul. 2009.
- [25] N. Roy and A. McCallum, "Toward optimal active learning through sampling estimation of error reduction," in *Proc. ICML*, Williamstown, MA, 2001, pp. 441–448.
- [26] T.-F. Wu, C.-J. Lin, and R. C. Weng, "Probability estimates for multi-class classification by pairwise coupling," *J. Mach. Learn. Res.*, vol. 5, pp. 975–1005, Aug. 2004.
- [27] H. S. Seung, M. Opper, and H. Sompolinski, "Query by committee," in *Proc. Annu. Workshop Comput. Learn. Theory*, New York, 1992, pp. 287–294.
- [28] J. J. Wei, C.J. Chang, N.K. Shou, and G.J. Jan, "ECG data compression using truncated singular value decomposition," *IEEE Trans. Biomed. Eng.*, vol. 5, no. 4, pp. 290–299, Dec. 2001.
- [29] R. Duda, P. Hart, and D. Stork, *Pattern Classification*, 2nd ed. New York: Wiley, 2001.

### **3. SVM Active Learning Through Significance Space Construction**

*Abstract* – Active learning is showing to be a useful approach to improve the efficiency of the classification process for remote sensing images. This chapter presents a new active learning strategy specifically developed for support vector machine (SVM) classification. It relies on the idea of: 1) reformulating the original classification problem into a new problem where it is needed to discriminate between significant and non significant samples, according to a concept of significance which is proper to SVM theory; and 2) constructing the corresponding significance space so that to suitably guide the selection of the samples potentially useful to better deal with the original classification problem. Experiments were conducted on both multispectral and hyperspectral images. Results show interesting advantages of the proposed method in terms of convergence speed, stability and sparseness.

The work presented in this chapter has been published in the *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 3, pp. 431–435, May 2011; Co-authors: F. Melgani, Y. Bazi.

### 3.1. Introduction

In order to obtain an efficient supervised classification system, it is necessary to address properly some important issues. One of them is the choice of the classifier to adopt. In particular, approaches based on support vector machines (SVM) have shown great potential in different research areas [1]-[3]. Classification systems based on SVM can give excellent performances, but, as for traditional classifiers, they depend strongly on the quality and quantity of the labeled data used to train the classifier. Indeed, training samples have to be representative of the statistical distribution of the data. However, the process of collection of training samples is not trivial. Indeed, it is performed by a human expert and thus subject to errors. Moreover, it is costly in terms of time and money. For these reasons, it is necessary to find a strategy to choose few training samples but fundamental for the correct discrimination between the set of considered classes.

In the last few years, there has been a growing interest in developing strategies for the (semi)automatic construction of the set of training samples. In the machine learning field, a recent approach focused on this topic is the so-called active learning approach. Starting from a small and suboptimal training set, additional samples, considered important, are selected in some way from a large amount of unlabeled data (learning set). These samples are labeled by the expert and then added to the training set. The entire procedure is iterated until a stopping criterion is satisfied.

In the literature, active learning methods have been applied successfully in different application fields. However, few works can be found for the problem of remote sensing image classification. In [4], a method based on the Fisher information matrix is used to construct the training set in the application of buried object detection. In [5], the authors propose a probabilistic method based on maximum likelihood classifiers for learning or adapting classifiers when significant changes in the spectral signatures between labeled and unlabeled data are present. In [6], the method proposed in [4] is extended to improve the detection of buried objects. The method fuses a graph-based semisupervised algorithm with an active learning procedure based on a mutual information measure. In [7], the authors discuss the margin sampling (MS) algorithm [8], a state-of-the-art active learning method based on the SVM classifier. Additionally, two novel methods are proposed and applied to the classification of very high resolution images.

In this chapter, an alternative method of active learning for the SVM classification of remote sensing images is presented.

The remaining part of the chapter is organized as follows. The proposed method is described in Section 3.2., while Section 3.3. presents the experimental results. Finally, conclusions are drawn in Section 3.4.

### 3.2. Proposed Method

First, let us focus on a generic binary classification problem. The extension to multiclass problems will be described at the end of this Section. Let us consider a training set composed initially of  $n$  labeled samples  $L = \{\mathbf{x}_i, y_i\}_{i=1}^n$  and an additional learning set composed of  $m$  unlabeled samples  $U = \{\mathbf{x}_j\}_{j=n+1}^{n+m}$ , with  $m \gg n$ . In order to increase the training set  $L$  with a series of samples chosen from the learning set  $U$  and labeled manually by the expert, an active learning algorithm has the task of choosing them properly so that to maximize the accuracy of the classification process while minimizing the number of learning samples to label (i.e., number of interactions with the expert).

The active learning method developed in this chapter is proposed specifically for classification problems based on SVM. The block diagram of the method is shown in Fig. 3.1.

The first step is called significance analysis and consists of detecting the most significant samples in the initial training set  $L$ . This operation is done by training a SVM classifier (named SVM1 in the block diagram) on the training set  $L$ . We define those that the classifier has found as support vectors (SVs) as significant samples, while the remaining samples are simply defined as nonsignificant. We construct a new

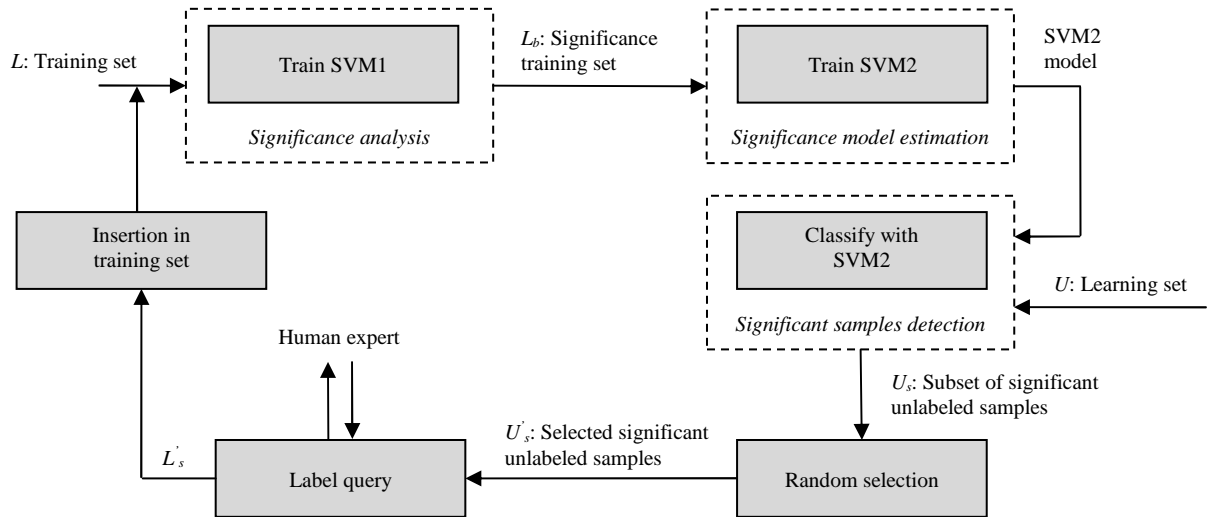


Fig. 3.1. Flow chart of the proposed active learning method.

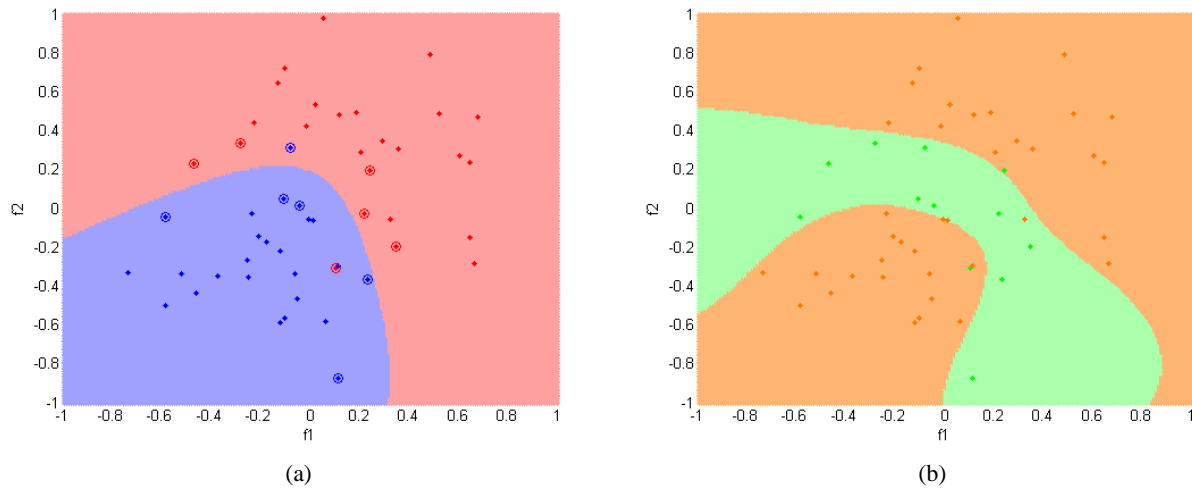


Fig. 3.2. Illustration with a toy classification problem. (a) Original classification space (SVs are circled). (b) Corresponding significance space.

set  $L_b$ , in which the samples of the original training set  $L$  are relabeled in function of the concept of significance. Therefore,  $L_b$  is a binary set containing significant and nonsignificant samples of  $L$ . In the second step, the task is to build a model able to discriminate the significant samples from the nonsignificant ones. For this purpose, another SVM classifier (called SVM2 in the block diagram) is trained on the new training set  $L_b$ . The model defined by this second classifier is used to classify the unlabeled samples of the learning set  $U$ . We define with  $U_s$  the samples of the learning set  $U$  classified as significant. The last step consists to select randomly  $N_s$  samples from the set  $U_s$ , where  $N_s$  is the number of samples to be added in the training set  $L$ . Successively, the selected samples  $U'_s$  are labeled by the expert and then added to the training set  $L$ . This entire active learning process is iterated until a predefined convergence condition is not satisfied (e.g., the total number of samples to add to the training set is not yet reached).

To better understand the proposed method, a toy example is shown in Fig. 3.2. In Fig. 3.2(a), we show the training samples with the original labels and the corresponding decision regions obtained after training the SVM1 classifier. The SVs, namely the significant samples, are the encircled points. In our method, we define a new problem, in which the labels of the training samples are redefined according to the significance concept. Such reformulation of the classification problem requires the training of a second SVM classifier

(SVM2) and leads to the identification of a region of significance [green area in Fig. 3.2(b)]. Such region represents the portion of the feature space, which conveys the samples potentially useful to better deal with the original classification problem.

SVMs are intrinsically binary classifiers. However, the classification of remote sensing images often involves the simultaneous discrimination of several information classes. In order to face this issue, multiclass classification strategies can be adopted. In this work, the SVM1 classification is performed by means of the one-against-one (OAO) strategy. Let  $\Omega = \{\omega_1, \omega_2, \dots, \omega_T\}$  be the set of  $T$  possible classes. First, an ensemble of  $T \cdot (T-1)/2$  (parallel) SVM1 binary classifiers is trained on data. Each classifier aims at solving a binary classification problem defined by the discrimination between two different classes. Then, in the classification phase, in order to decide which label assign to each sample, the class with the maximum number of votes is chosen. In our active learning method, the SVM2 classification is also implemented through the OAO strategy. To each SVM1 binary classifier, we associate an SVM2 binary classifier to determine the significance region of the corresponding couple of original classes. After training the ensemble of  $T \cdot (T-1)/2$  SVM2 binary classifiers, we decide that a given sample is globally (in reference to the multiclass problem) significant if the majority of the  $T-1$  binary classifiers associated with the class estimated by the SVM1 classification agrees on its significance.

### 3.3. Experiments

#### 3.3.1. Experimental Setup

In order to validate the proposed active learning method, experiments were conducted on two different remote sensing data sets. The first data set represents a multispectral VHR image, acquired by the QuickBird sensor in April 2002. Four spectral bands with a spatial resolution of 0.6 m were considered to feed the classification process. The image refers to a portion of the city of Boumerdes (Algeria), in which four land cover types are dominant: water, soil, vegetation, and man-made structures. The second data set is a hyperspectral image and is characterized by 102 bands, acquired by the Reflective Optics System Imaging Spectrometer (ROSIS) sensor over a part of the city of Pavia (Italy) in July 2002. The spatial resolution is equal to 1.3 m. Nine classes were considered: water, trees, asphalt, bricks, bitumen, tiles, shadow, meadows, and bare soil.

In all the following experiments, for both data sets, all the available samples were split in two sets, corresponding to learning  $U$  and test sets. The detailed numbers of learning and test samples are reported in Table 3.I. In this table, we report the number of samples of the initial training set  $L$  for each class too. The initial training samples were selected randomly from the learning set  $U$ . For the first data set, the active learning algorithm was run until the number of training samples was equal to 2991, adding 25 samples at each iteration. Analogously, for the second data set, 50 samples were added at each iteration up to 2000 samples. The entire active learning process was run ten times, each with a different initial training set so that to yield statistically reliable results. At each run, the initial training samples were chosen in a completely random way.

Classification performance was evaluated in terms of several measures which are: 1) the overall accuracy (OA), which is the percentage of correctly classified samples among all the considered samples, independently of the classes they belong to; 2) the average accuracy (AA), which is the average over the classification accuracies obtained for the different classes; 3) the standard deviations ( $\sigma$ ) of OA and AA, in order to evaluate the stability of the active learning method; 4) the number of SVs (#SV).

An SVM classifier was also trained on the entire learning set in order to have a reference-training scenario, called “full” training. On the one hand, the classification results obtained in this way represent an upper bound for the accuracies. On the other, we expect that the lower accuracy bound will be given by the



TABLE 3.I  
NUMBERS OF INITIAL TRAINING, LEARNING, AND TEST SAMPLES FOR (a) THE BOUMERDES AND (b) THE PAVIA DATA SETS

(a)

Class	Water	Soil	Vegetation	Man-made	Total
Initial training samples	4	4	4	4	16
Learning samples	6000	3380	4499	3978	17857
Test samples	6000	2957	4455	4113	17525

(b)

Class	Water	Trees	Asphalt	Bricks	Bitumen	Tiles	Shadow	Meadows	Bare soil	Total
Initial training samples	5	6	5	6	5	6	5	6	6	50
Learning samples	824	820	816	808	808	1260	476	824	820	7456
Test samples	65147	6778	8432	1891	6479	41566	2387	2266	5764	140710

completely random selection strategy (R). We recall that the purpose of any active learning strategy is to converge to the performance of the “full” training scenario faster than the R method. Moreover, the proposed approach is compared to the performances given by the state-of-the-art active learning method based on the MS approach [8].

### 3.3.2. Experimental Results

For the “full” classifier, the OA is equal to 95.12% and 97.75% for the Boumerdes and Pavia data sets, respectively. In Fig. 3.3(a)-(d), we show the OA in function of the number of training samples for the proposed active learning strategy, the MS, and the random ones. For the Boumerdes data set, the proposed strategy converges to the “full” accuracy using about 1100 training samples, which represent 5.6% of the entire learning set. Instead, about 2000 samples are necessary for MS and R methods to converge. For the Pavia data set, 700 and 800 samples are required to converge for the proposed and the MS methods respectively, while the R strategy converges for a number of training samples greater than 2000. We note that, before convergence, the proposed method gives the best performance. In particular, for the Boumerdes data set, for which the initial accuracies are low, the MS method presents bad performances in the first iterations of the active learning process, while the proposed strategy is characterized by good accuracies in the first steps too. This can be explained by the fact that, when few training samples are available, the inferred decision boundary is precarious and thus relying only on samples which are closest to it, as the MS method does, can be counterproductive. Our method allows to pick up samples not just along the decision boundary but also in the surrounding (as shown in Fig. 3.2(b)), making it more appropriate to face this problem.

To better understand the behavior of the proposed method, we show in Fig. 3.3. the evolution of the cross validation (CV) accuracy and the #SV, identified on the basis of the following definition: “a sample is an SV if it is an SV for at least one binary classifier of the OAO multiclass architecture”. It is interesting to observe that the value of CV tends to decrease in the first iterations, while we have an increase of the CV value only when a sufficient number of samples have been added to the training set. The decrease of the CV value means that samples difficult to classify are added to the training set. However, these new samples are highly informative and thus allow improving the generalization performance (i.e., the accuracy on the test samples). A completely different behavior is obtained for the R strategy, for which the CV value tends to increase from the beginning. Analogously, we note that in our active learning method the #SV value tends to increase faster than the R method. Therefore, most of the samples selected by the active learning method are

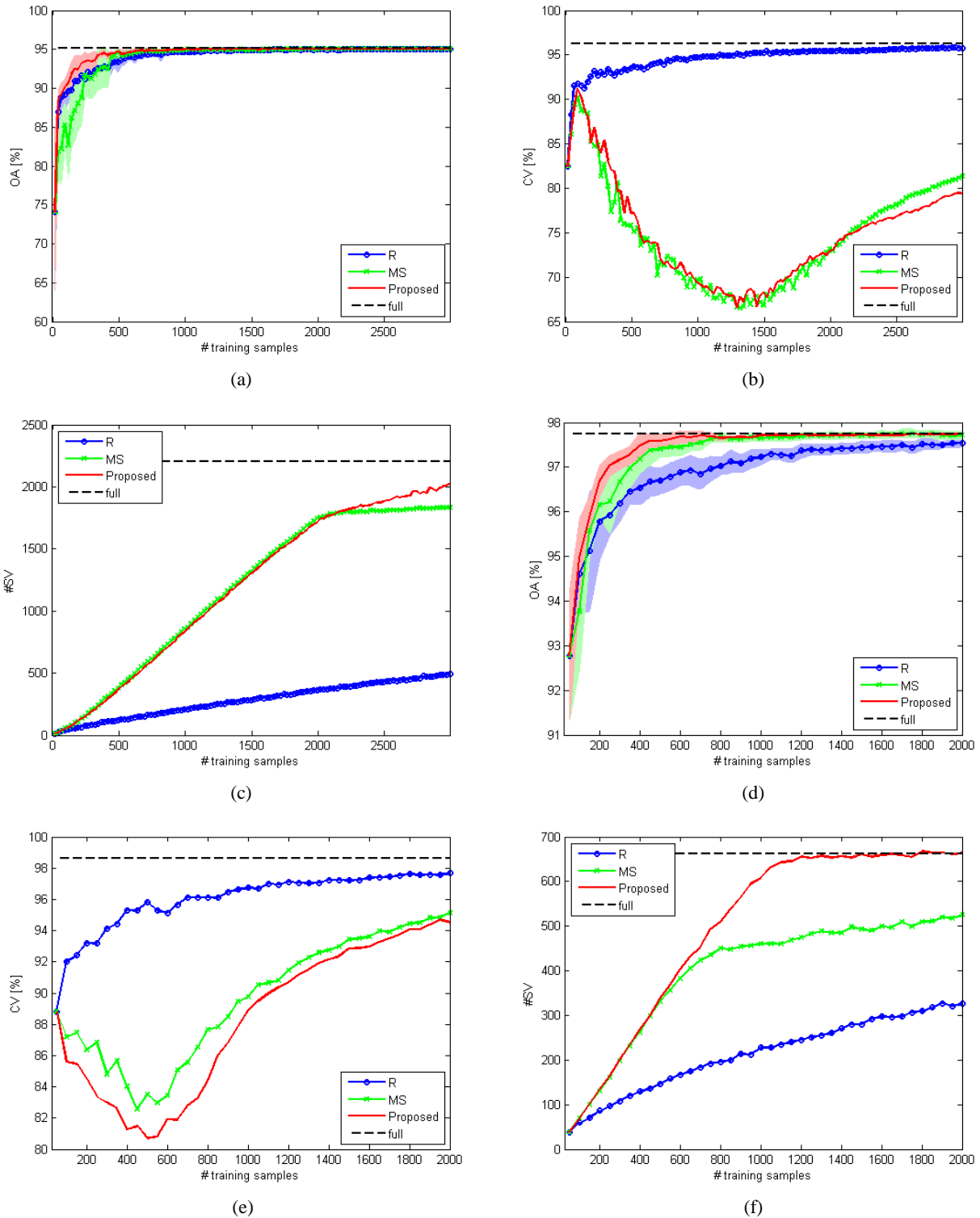


Fig. 3.3. Performances achieved on (a, b, c) the Boumerdes and (d, e, f) the Pavia data sets in terms of (a, d) OA, (b, e) CV accuracy, (c, f) #SV. Each graph shows the results in function of the number of training samples and averaged over ten runs of the algorithm, each with a different initial set. The shades areas in (a, d) show the standard deviation of the OA over the ten considered runs. R= random, MS= margin sampling, Proposed= proposed method, full= full SVM.

SVs and hence are important for the discrimination among the classes. For the R method, the number of SVs tends to increment much slower. The fast increment of the number of SVs for the active learning strategy shows clearly that the samples added to the training set are really important for the classification process. Finally, we observe that the curves of #SV present a breakpoint after which the proposed method is

TABLE 3.II  
 OA, AA, AND CV ACCURACIES, STANDARD DEVIATIONS ( $\sigma$ ), AND #SV  
 ACHIEVED ON (a) THE BOUMERDES AND (b) THE PAVIA DATA SETS

(a)

Method	#training samples	OA	$\sigma_{OA}$	AA	$\sigma_{AA}$	CV	#SV
<b>Full</b>	17857	95.12	-	94.92	-	96.26	2201
<b>Initial</b>	16	74.09	10.15	75.66	7.35	82.50	14
<b>R</b>	641	94.20	0.50	93.96	0.54	93.93	147
<b>MS</b>	466	94.18	1.07	93.95	1.17	75.79	357
<b>Proposed</b>	266	94.21	0.26	93.98	0.27	83.95	165
<b>R</b>	866	94.62	0.33	94.40	0.33	94.57	189
<b>MS</b>	716	94.63	0.51	94.44	0.48	71.68	589
<b>Proposed</b>	391	94.74	0.13	94.55	0.14	79.59	267

(b)

Method	#training samples	OA	$\sigma_{OA}$	AA	$\sigma_{AA}$	CV	#SV
<b>Full</b>	7456	97.75	-	94.77	-	98.62	663
<b>Initial</b>	50	92.78	1.47	84.87	2.91	88.80	39
<b>R</b>	700	96.85	0.35	92.25	1.28	96.13	183
<b>MS</b>	350	96.97	0.43	93.07	0.99	85.66	232
<b>Proposed</b>	200	97.04	0.24	93.08	0.41	84.45	134
<b>R</b>	1200	97.37	0.18	93.65	0.94	97.13	245
<b>MS</b>	600	97.45	0.20	94.05	0.58	83.42	382
<b>Proposed</b>	350	97.28	0.11	93.68	0.26	82.63	234

characterized by a greater increment of the #SV. This means that, after a given point, the MS starts to oversample along the decision boundary between classes leading to redundant samples, while the proposed method, being less constrained by the decision boundary, samples also away from it [as illustrated in Fig. 3.2(b)].

The obtained results are shown in greater detail in Table 3.II. In particular, we considered for each method the minimum number of training samples for which we have a decrease of 1% of the OA with respect to the OA of the “full” classifier. The same analysis has been done for an OA decrease of 0.5%. We report the values of OA and AA, standard deviations ( $\sigma_{OA}$  and  $\sigma_{AA}$ ) associated with the accuracies, CV accuracy, and #SV. As can be seen, the proposed strategy is characterized by a better performance with respect to the MS strategy from different points of view. First, similar values of accuracies (OA and AA) are obtained using a minor number of training samples. In this way, we have a reduction of the manual work for sample labeling and a decreasing of the computational time necessary to train the classifier. Another improvement is given by the better values of standard deviation associated with the accuracies ( $\sigma_{OA}$ ,  $\sigma_{AA}$ ). Indeed, minor values of standard deviation mean that the proposed method exhibits a greater level of stability respect to the random selection of the initial training set. Another interesting result is the decreasing of #SV. In general, the reduction of the #SVs is an important task in SVM approaches because sparseness permits to simplify the classification model and to increment the generalization capabilities. Moreover, the classification of a generic test sample will require a lower computational burden.

In Fig. 3.4., we report the percentage of selected samples for the soil and man-made classes of the Boumerdes data set. The soil class is classified with a very high accuracy (99.19% for the full case), while the man-made class is more difficult to discriminate (88.69% for the full case). For both classes, the percentage of samples selected by the R strategy tends to the prior probability of the classes. Conversely, for the active learning strategies (the proposed one and the MS), a completely different behavior comes out. For the soil class, we observe a fast decrease of the percentage of selected samples, while for the man-made class the active learning process tends to select a large number of samples. Therefore, for both active learning

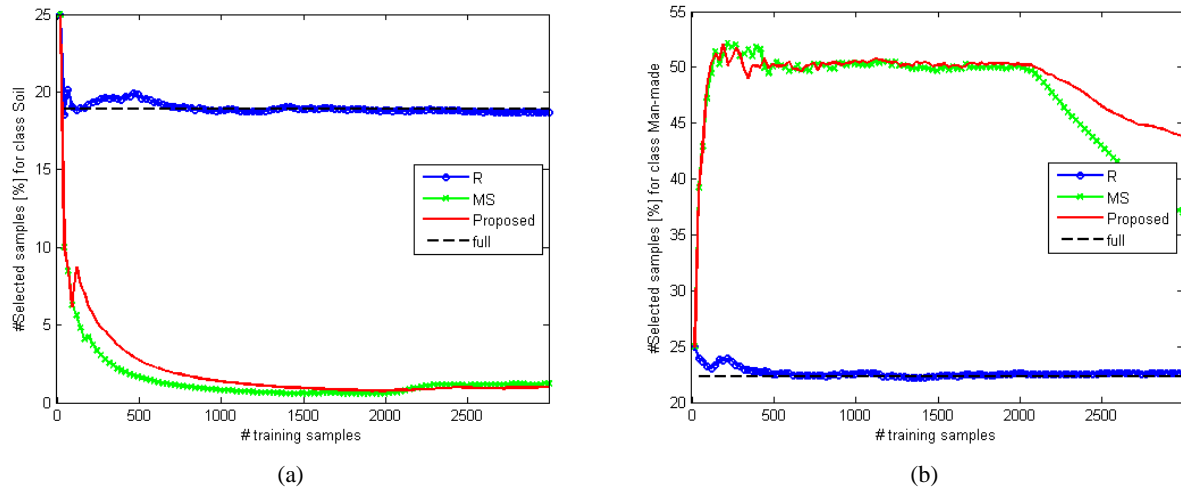


Fig. 3.4. Percentage of selected samples for classes (a) soil and (b) man-made of the Boumerdes data set.

strategies, the most difficult classes are more sampled than others, with a faster increment of the accuracy values.

### 3.4. Conclusion

In this chapter, we have proposed a new active learning strategy specifically developed for SVM classification. The experimental results obtained on VHR and hyperspectral images show good capabilities of the proposed method for selecting significant samples. Advantages in terms of convergence speed, stability, and reduction of the number of SVs have been empirically evaluated with respect to state-of-the-art MS strategy.

The drawbacks of the proposed strategy are as follows: 1) in case of overfitting (due for instance to model selection problems), most of the samples become SVs; and so, most of the learning samples are detected as significant, thus making the proposed algorithm tend to a simple random sample selection; 2) an increment of the computational cost, given by the training of two stages of SVM classifiers. For the Boumerdes data set, the MS method required 130 min to perform the entire selection process and accuracy evaluation, while the proposed strategy consumed 233 min. Similarly, for the Pavia data set, 60 and 105 min were necessary for the MS and the proposed method, respectively.

While in this research the initial training set was chosen in a random way, we think that a more sophisticated initialization strategy could improve the performance of the active learning process. A further enhancement could be obtained by combining our method with another sampling strategy, such as MS, to better explore the margin surroundings. Research is in progress in these directions.

### 3.5. Acknowledgment

The authors would like to thank Prof. P. Gamba (University of Pavia, Italy) for providing the hyperspectral image and Dr. C.-C. Chang and Dr. C.-J. Lin for supplying the software LIBSVM used in this study.

### 3.6. References cited in Chapter 3

- [1] F. Melgani and Y. Bazi, "Classification of electrocardiogram signals with support vector machines and particle swarm optimization," *IEEE Trans. Inf. Technol. Biomed.*, vol. 12, no. 5, pp. 667–677, Sep. 2008.
- [2] E. Pasolli, F. Melgani, and M. Donelli, "Automatic analysis of GPR images: a pattern recognition approach," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 7, pp. 2206–2217, Jul. 2009.

- [3] N. Ghoggali and F. Melgani, “Automatic Ground-Truth Validation with Genetic Algorithms for Multispectral Image Classification”, *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 7, pp. 2172–2181, Jul. 2009.
- [4] Y. Zhang, X. Liao, and L. Carin, “Detection of buried targets via active selection of labeled data: application to sensing subsurface UXO,” *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 11, pp. 2535–2543, Nov. 2004.
- [5] S. Rajan, J. Ghosh, and M. M. Crawford, “An active learning approach to hyperspectral data classification,” *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 4, pp. 1231–1242, Apr. 2008.
- [6] Q. Liu, X. Liao, and L. Carin, “Detection of unexploded ordnance via efficient semisupervised and active learning,” *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 9, pp. 2558–2567, Sep. 2008.
- [7] D. Tuia, F. Ratle, F. Pacifici, M. F. Kanevski, and W. J. Emery, “Active learning methods for remote sensing image classification,” *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 7, pp. 2218–2232, Jul. 2009.
- [8] G. Schohn and D. Cohn, “Less is more: Active learning with support vectors machines,” in *Proc. 17th ICML*, Stanford, CA, 2000, pp. 839–846.



## 4. SVM Active Learning using Spatial Information

*Abstract – The performances of supervised approaches for the classification of remote sensing images depend strongly on the quality of the labeled data used to train the classifier. In this chapter, the problem of the collection of the training samples is faced through a new active learning approach. While strategies available in the literature formulate the active learning problem in the spectral domain only, we propose to combine spectral and spatial information in the iterative process of training sample selection. In particular, three criteria based on spatial information are introduced in order to encourage the selection of samples distant from the samples already composing the current training set. In the first strategy, we compute the Euclidean distances in the spatial domain from the training samples, while the second one is based on the Parzen window method applied in the spatial domain. Finally, the last criterion involves the concept of spatial entropy. Experiments on two very high resolution images acquired by QuickBird show the effectiveness of regularization in spatial domain and open challenging perspectives for terrain campaigns planning.*

## 4.1. Introduction

In the remote sensing community, two main approaches for the classification of images have been proposed: supervised and unsupervised. The supervised methods have shown promising performances, but they depend strongly on the quality of the labeled data used to train the classifier. Indeed, training samples have to be representative of the statistical distribution of the data. However, the process of training sample collection is not obvious, because it is performed manually by human experts and thus it is characterized by errors and costs. The acquisition of training samples directly on the field or through the visual inspection of the images can be performed only on a limited portion of the available data given the constraints in terms of time and money. For this reason, in the last few years there has been a growing interest in developing strategies for the semi-automatic selection of the training samples. In the machine learning field, the active learning approach represents an interesting solution to face this problem. Considering a small and suboptimal initial training set, few additional samples are selected from a large amount of unlabeled data (learning set) through an iterative process. The aim of active learning is to rank the learning set according to an opportune criterion, or a heuristic, that allows to select the most useful samples to improve the model, thus minimizing the number of training samples necessary to maintain discrimination capabilities as high as possible. In the last few years, different solutions have been proposed and applied successfully in different applications fields [1], [2] and for remote sensing problems [3]-[12].

The active learning strategies proposed in the remote sensing field are based on heuristics in the spectral domain. In [3] the authors present a strategy based on support vector machines (SVM). This strategy queries for the most ambiguous samples as measured by their distance from the current separating hyperplane as done in the margin sampling (MS) method [13]. In [4], the authors propose a solution for problems of buried object detection. The signatures for which knowledge of the associated labels is most relevant in the context of detector design are selected. In [5], the active learning approach is applied in the context of satellite image retrieval in order to minimize redundancy between the images shown to the user. In [6], the authors propose a probabilistic method using maximum likelihood and binary hierarchical classifiers. The samples that mostly change the existing belief in the *a posteriori* probability distribution function are selected. The method proposed in [4] is extended in [7], in which a graph-based semisupervised algorithm is fused with an active learning procedure based on a mutual information measure. In [8], two methods are proposed. The first strategy is an extension of the MS, in which sample distribution is considered in order to avoid oversampling on dense regions. The second one is based on classification disagreement using a committee of classifiers. In [9], the authors propose a strategy to label samples grouped with hierarchical clustering in order to match the data relationship discovered by the clustering algorithm with the class semantics desired by the user. In [10], a new semisupervised classification approach is introduced, in which unlabeled training samples are selected by means of an active-selection strategy based on the entropy of the samples and used to improve the estimation of the class distributions. In [11], different query functions for the SVM classification are investigated. In particular, they are based on the evaluation of two criteria: uncertainty and diversity. In [12], the original classification problem is reformulated into a new problem where it is needed to discriminate between significant and nonsignificant samples, according to a concept of significance which is proper to the SVM theory.

The common denominator of active learning methods introduced up-to-now in the literature is that they are all formulated in the spectral domain and all ignore the spatial dimension characterizing images to classify. However, in the remote sensing literature, it has been demonstrated how the integration of spectral and spatial information is important for solving problems in different contexts [14]-[29]. For instance, in the field of hyperspectral images, classification problems are faced in different works by adopting different approaches, such as composite kernels [14], morphological operators [15], and Markov random field (MRF) regularization [16]. In [17], the authors propose to solve the problem of resolution enhancement through



spectral unmixing and superresolution mapping, by which spatial and spectral information are fused. In [18]-[19], spatial information is incorporated into the spectral-based endmember search process, which has the purpose of selecting a collection of pure signature spectra of the materials present in the hyperspectral scene. Similarly to hyperspectral images, several works are proposed for very high resolution (VHR) images. For instance, solutions based on morphological operators [20], textural metrics [21], and composite kernels [22] are presented for classification problems. For synthetic aperture radar (SAR) images, a specific kernel is used by combining both radiometric and texture information in a semisupervised strategy for oil-slick detection [23]. Images acquired at different times can be used for change detection problems, as done for data acquired by different sensors through MRFs [24], SAR images by markovian fusion [25], and optical images using linear spatial-oriented operators [26]. A natural use of spatial information is represented by image registration techniques. For instance, in [27] spatial and spectral information are combined for this purpose. Finally, in [28]-[29], the authors propose methodologies for the contextual reconstruction of cloud-contaminated areas in multitemporal images by opportunely capturing spatial and spectral correlations characterizing the considered image.

In this chapter, we investigate how spatial information can be useful in the process of training sample collection for classification of remote sensing images. In particular, we propose to evaluate each sample of the learning set using two different heuristics, the first one spectral and the second one spatial. After that, the two different heuristics are opportunely combined through the realization of the Pareto front in order to consider simultaneously spectral and spatial information. While in terms of spectral criterion we adopt, for its simplicity and effectiveness, the traditional MS strategy thought for classification based on SVM approaches, for the spatial information we propose three different criteria. The criteria are based on adding samples that are distant spatially from the samples already composing the current training set. In the first strategy we compute explicitly the Euclidean distances in the spatial domain from the training samples, while the second one is based on the Parzen window method in the spatial domain. Finally, the last criterion involves the concept of spatial entropy. To investigate the performance of the proposed approach and to compare the three spatial heuristics, we conducted an experimental study based on two VHR images acquired by QuickBird. The obtained results show that interesting performances can be achieved. Advantages in terms of classification accuracy and classification reliability have been empirically evaluated with respect to strategies that do not exploit spatial information.

The remaining part of the chapter is organized as follows. In Section 4.2., the strategy of integration of spatial and spectral information and the three relative spatial criteria are described. Section 4.3. presents the data sets used in the experimental analysis and the corresponding results. Finally, conclusions are drawn in Section 4.4..

## 4.2. Proposed Method

### 4.2.1. Proposed Active Learning Framework

Let us consider a training set composed initially of  $n$  labeled samples  $L = \{\mathbf{x}_i, y_i\}_{i=1}^n$  and an additional learning set composed of  $m$  unlabeled samples  $U = \{\mathbf{x}_j\}_{j=n+1}^{n+m}$ , with  $m \gg n$ . In order to increase the training set  $L$  with a series of samples chosen from the learning set  $U$  and labeled manually by the expert, an active learning algorithm has the task of choosing them properly so that to maximize the accuracy of the classification process while minimizing the number of learning samples to label (i.e., number of interactions with the expert).

In Fig. 4.1., we show the flow chart of the active learning strategy proposed in this work. The objective is to combine opportunely spectral and spatial information in the active selection process of the training

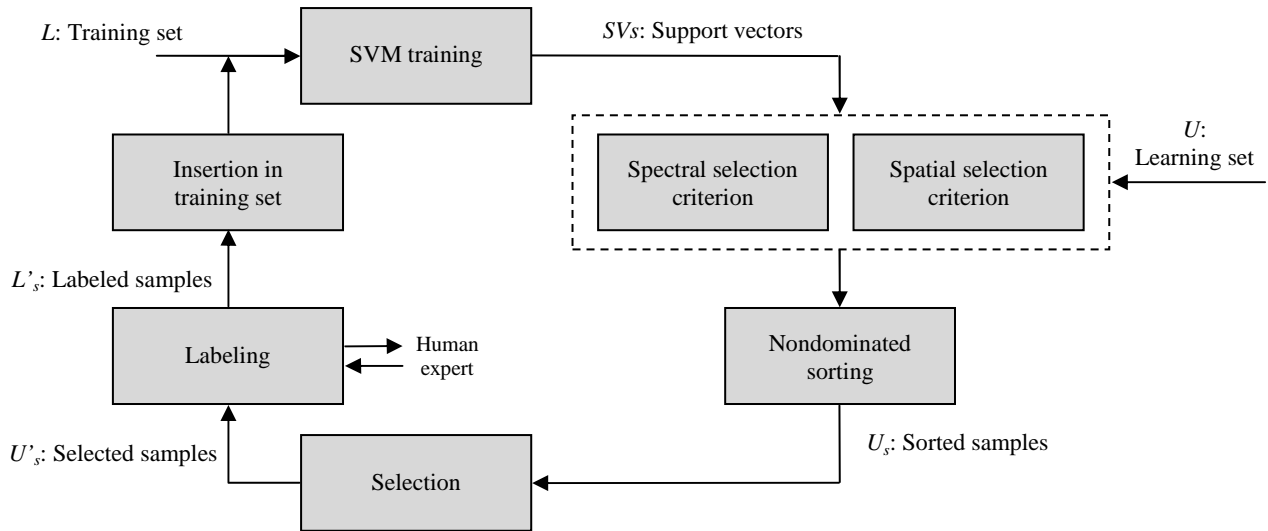


Fig. 4.1. Flow chart of the active learning method proposed for integration of spatial and spectral information.

samples. The method is proposed specifically for classification problems based on SVM. We refer the reader to [30], [31] for more details about SVM. Starting from the small and suboptimal training set  $L$ , a multiclass SVM classifier is trained on this set of samples. The classification model constructed in this way is used to evaluate the unlabeled samples of the learning set  $U$ . In particular, each sample is evaluated using two different heuristics. The first heuristic  $f_1$  represents a spectral criterion, while the second heuristic  $f_2$  is based on spatial information. At this point the two different heuristics  $f=[f_1, f_2]$  have to be opportunely combined. For this purpose, we form the Pareto front, composed of all the nondominated samples (solutions). In this way the combined criterion represents a tradeoff between spectral and spatial information. Finally, from the sorted samples  $U_s$ ,  $N_s$  samples belonging to the Pareto front are selected from the learning set  $U$ , where  $N_s$  is the number of samples to be added in the training set  $L$ . Successively, the selected samples  $U'_s$  are labeled by the human expert and added to the training set  $L$ . The entire process is iterated until the total number of samples to add to the training set is reached.

Algorithm 4.1. resumes the proposed active learning strategy.

---

**Algorithm 4.1.:** Proposed Active Learning Framework

---

**Inputs:**

$L$ : initial training set, composed of  $n$  labeled samples.

$U$ : learning set, composed of  $m$  ( $m \gg n$ ) unlabeled samples.

$N_s$ : number of samples to add at every iteration of the active learning process.

**Output:**

$L$ : final training set.

---

**Repeat**

1. Train the SVM classifier with the current training set  $L$ , while estimating its free parameters by crossvalidation (CV).
  2. Compute the criterion  $f_1$  based on spectral information for each sample  $x_j$  ( $j = n+1, n+2, \dots, n+m$ ) of the learning set  $U$ .
  3. Analogously to the previous step, compute the criterion  $f_2$  based on spatial information for each sample  $x_j$  ( $j = n+1, n+2, \dots, n+m$ ).
  4. Combine spectral  $f_1$  and spatial  $f_2$  criteria by identifying the nondominated solutions. The samples are now ranked in the set  $U_s$ .
  5. Select the first  $N_s$  samples from  $U_s$ .
-

---

6. Label the selected samples  $U'_s$ .

7. Add the labeled samples  $L'_s$  to the training set  $L$  and remove them from  $U$ .

**Until** the predefined convergence condition is not satisfied (e.g., the total number of samples to add to the training set is not yet reached).

---

In the following, we describe in greater detail the main ingredients of the proposed methodology, namely spectral and spatial criteria and nondominated sorting.

#### 4.2.2. Spectral Selection Criterion: Margin Sampling

Regarding the spectral criterion, for its simplicity and effectiveness, we adopt the MS strategy [13], which has been proposed specifically for classification problems based on SVM. Considering a simple binary case with linearly separable classes, support vectors (SVs) are the samples of the training set  $L$  which are closest to the hyperplane that describes the decision boundary given by the SVM classifier. If we consider the unlabeled learning set  $U$ , we can assume that the samples that are the closest the decision boundary are the most interesting samples, because they have a larger probability to become SVs when added to the training set. Therefore, the samples selected by MS are the ones showing the minimum absolute values of the discriminant function. The same reasoning is applied in case of nonlinearly separable classes.

The assumptions done in the binary case can be used in a multiclass classification problem too. In this context, a solution is given in [13], in which an OAA SVM classifier is adopted. For each sample, the maximum value among the discriminant functions provided by the  $T$  binary classifiers is exploited as a sample indicator, where  $T$  is the number of different classes. Then, the samples with the minimum indicator values are selected. In this work, we use an alternative solution based on the OAO SVM classifier, which has shown its effectiveness for active learning problems in the field of electrocardiographic signal classification [2]. In this context,  $T \cdot (T-1)/2$  binary classifiers are involved. For each sample  $\mathbf{x}_j$  ( $j = n+1, n+2, \dots, n+m$ ), we calculate the number of votes of each class  $\mathbf{v}_j \in N^T = [v_{j,1}, v_{j,2}, \dots, v_{j,T}]$ . The class  $\omega_{MAX,j}$  with the largest number of votes  $v_{MAX,j}$  is first identified. Then, considering the  $T-1$  classifiers associated with the class  $\omega_{MAX,j}$ , the minimum absolute value of the discriminant function  $f_{MIN,j}$  is calculated. Finally, the samples characterized by the minimum values of  $f_{MIN,j}$  are selected.

Algorithm 4.2. resumes the spectral criterion based on the MS strategy.

---

#### **Algorithm 4.2.:** Spectral Selection Criterion – Margin Sampling

---

1. Compute the number of votes of each class  $\mathbf{v}_j \in N^T = [v_{j,1}, v_{j,2}, \dots, v_{j,T}]$  for each sample  $\mathbf{x}_j$  ( $j = n+1, n+2, \dots, n+m$ ) of the learning set  $U$ .
  2. Identify the class  $\omega_{MAX,j}$  with the largest number of votes  $v_{MAX,j}$ .
  3. Calculate the minimum absolute value of the discriminant function  $f_{MIN,j}$  by considering the  $T-1$  classifiers associated with the class  $\omega_{MAX,j}$ .
  4. Set  $f_j(j) = f_{MIN,j}$ .
- 

In this study, three criteria are proposed to integrate spatial information in the heuristic. They all consider the subset of support vectors identified by the training process using the training set  $L$ . We define  $S_n$  as the number of support vectors identified.

#### 4.2.3. Spatial Selection Criteria

##### 4.2.3.1. Spatial Distance from the Closest SV

The first criterion, named Sp1 in the rest of the chapter, consists to calculate for each sample  $\mathbf{x}_j$  ( $j =$

$n+1, n+2, \dots, n+m$ ) the spatial Euclidean distances from the support vectors  $\mathbf{d}_j \in R^{S_n} = [d_{j,1}, d_{j,2}, \dots, d_{j,S_n}]$ :

$$d_{j,i} = \|\mathbf{p}_j - \mathbf{p}_i\| \quad (4.1)$$

where  $\mathbf{p}$  represents the two-dimensional vector containing the position of the considered sample in the spatial domain of the image. After that, the nearest support vector  $s_{MIN,j}$  is identified and the corresponding distance  $d_{MIN,j}$  is considered for the spatial criterion. In particular, the negative value is adopted in order to convert the maximization problem into a minimization one. In this way, we favor the selection of samples placed in areas of the image not covered by SVs.

Algorithm 4.3. synthesizes the proposed spatial criterion based on the distance from the closest SV.

---

**Algorithm 4.3.:** Spatial Selection Criterion – Spatial Distance from the Closest SV

---

1. Compute the spatial Euclidean distances from the  $S_n$  different support vectors  $\mathbf{d}_j \in R^{S_n} = [d_{j,1}, d_{j,2}, \dots, d_{j,S_n}]$  for each sample  $\mathbf{x}_j$  ( $j = n+1, n+2, \dots, n+m$ ) of the learning set  $U$ .
  2. Identify the support vector  $s_{MIN,j}$  nearest to the sample.
  3. Consider the distance  $d_{MIN,j}$  associated with the support vector  $s_{MIN,j}$ .
  4. Set  $f_2(j) = -d_{MIN,j}$ .
- 

#### 4.2.3.2. Parzen Window Method in the Spatial Domain

In the second strategy (Sp2), we apply the Parzen window method in the spatial domain. Such method represents a standard way to estimate probability density functions of random variables [32]. After calculating the distances from SVs, we do not consider the nearest SV only as done in the strategy Sp1, but we combine opportunely all the distance values. For this purpose, we use a combination of the distances, where distance values are defined using a kernel operator. The spatial criterion  $d_{KER,j}$  for the sample  $\mathbf{x}_j$  ( $j = n+1, n+2, \dots, n+m$ ) is given by the following formulation

$$d_{KER,j} = \sum_{i=1}^{S_n} K(d_{j,i}) \quad (4.2)$$

where  $K(\cdot)$  is a Gaussian function, i.e.,

$$K(d_{j,i}) = \exp\left(-d_{j,i}^2 / \lambda^2\right). \quad (4.3)$$

Note that the kernel operator is not applied in the spectral domain but in the spatial one. The parameter  $\lambda$  is related to the width of the kernel and has to be set by the user. In this study, we suggest to set it empirically as follows

$$\lambda = \frac{1}{m} \sum_{j=n+1}^{n+m} d_{MIN,j}. \quad (4.4)$$

In this way the parameter  $\lambda$  is adaptive and is modified throughout the iterations according to the distances observed. One can reasonably expect that it tends to become smaller, as the distance values  $d_{MIN,j}$  tend to decrease through the iterations of the algorithm.

Algorithm 4.4. synthesizes the proposed spatial criterion based on the Parzen window method in the spatial domain.

---

**Algorithm 4.4.:** Spatial Selection Criterion – Parzen Window Method in the Spatial Domain

---

1. Compute the spatial Euclidean distances from the  $S_n$  different support vectors  $\mathbf{d}_j \in R^{S_n} = [d_{j,1}, d_{j,2}, \dots, d_{j,S_n}]$  for each sample  $\mathbf{x}_j$  ( $j = n+1, n+2, \dots, n+m$ ) of the learning set  $U$ .
  2. Identify the support vector  $s_{MIN,j}$  nearest to the sample.
  3. Consider the distance  $d_{MIN,j}$  associated with the support vector  $s_{MIN,j}$ .
  4. Compute the parameter  $\lambda$  using (4.4).
-

---

5. Compute the kernel distance  $d_{KER,j}$  using (4.2).

6. Set  $f_2(j)=d_{KER,j}$ .

---

#### 4.2.3.3. Spatial Entropy Variation

The last strategy (Sp3) proposed involves the concept of spatial entropy. In information theory, for a discrete random variable  $Z$  with possible values  $\{z_1, z_2, \dots, z_v\}$ , the entropy  $H(Z)$  can be written as

$$H(Z) = -\sum_{k=1}^v p(z_k) \log_b p(z_k) \quad (4.5)$$

where  $p(\cdot)$  denotes the probability function of  $Z$  and  $b$  is the base of the logarithm used. The entropy value is maximized when  $p(\cdot)$  assumes a uniform distribution. In our active learning problem, in order to have a spatial distribution statistically significant, we subdivide (quantize) the entire image in  $h$  different regions. In particular, we consider a value of  $h$  equal to

$$h = \lfloor \sqrt{Sn} \rfloor^2 \quad (4.6)$$

and subdivide the image in both the horizontal and vertical directions into  $\sqrt{h}$  uniform intervals in order to obtain  $h$  rectangles of equal area. At this point, the probability value for each region is computed as the number of SVs present in that region divided by the total number of support vectors  $Sn$ . First, the entropy value  $H_L$  is calculated considering the SVs associated with the current training set  $L$  only. Then, for each sample  $\mathbf{x}_j$  ( $j = n+1, n+2, \dots, n+m$ ) we calculate the corresponding entropy value  $H_j$  by supposing that it would become a SV in the new training set. Consequently, we derive the spatial entropy variation  $H_{V,j}$ , which is defined as the difference between the values of entropy with and without the insertion of the sample in the training set. The purpose is to maximize the spatial entropy variation value, or equivalently to minimize the negative value of this quantity, in order to distribute spatially as most as possible the training samples.

Algorithm 4.5. resumes the proposed spatial criterion based on the spatial entropy variation.

---

#### Algorithm 4.5.: Spatial Selection Criterion – Spatial Entropy Variation

---

1. Compute the parameter  $h$  using (4.6) and subdivide the image in  $h$  rectangular regions of equal area.
  2. Compute the spatial entropy  $H_L$  value by considering the training set  $L$  using (4.5).
  3. Compute the spatial entropy  $H_j$ , for each sample  $\mathbf{x}_j$  ( $j = n+1, n+2, \dots, n+m$ ) of the learning set  $U$ .
  4. Compute the spatial entropy variation  $H_{V,j} = H_j - H_L$ .
  5. Set  $f_2(j) = -H_{V,j}$ .
- 

#### 4.2.4. Nondominated Sorting

The concept of nondominated sorting arises when multiple measures of competing objectives (criteria) have to be simultaneously optimized, a common scenario in several practical applications. Optimizing multiple objectives involves finding a set of optimal solutions rather than a single one. The selection of a solution from this set is not trivial and is usually user-dependent. From a mathematical viewpoint, a general multiobjective optimization problem can be formulated as follows.

Find the vector  $\mathbf{p}^*$  that minimizes the ensemble of  $Q$  objective functions (in our case,  $Q = 2$ ), i.e.,

$$f(\mathbf{p}) = [f_i(\mathbf{p}), \quad i = 1, 2, \dots, Q] \quad (4.7)$$

subject to the  $J$  equality constraints, i.e.,

$$g_j(\mathbf{p}) = 0, \quad j = 1, 2, \dots, J \quad (4.8)$$

and the  $K$  inequality constraints, i.e.,

$$h_k(\mathbf{p}) \leq 0, \quad k = 1, 2, \dots, K \quad (4.9)$$

where  $\mathbf{p}$  is a solution to the considered optimization problem.

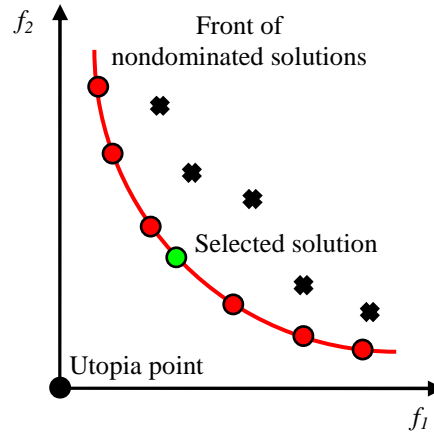


Fig. 4.2. Illustration of a front of nondominated solutions.

Solving a multiobjective optimization problem relies on an important concept, which is that of domination. A solution  $\mathbf{p}_i$  is said to dominate another solution  $\mathbf{p}_j$  if and only if

$$\forall k \in \{1, 2, \dots, M\}, f_k(\mathbf{p}_i) \leq f_k(\mathbf{p}_j) \wedge \exists k \in \{1, 2, \dots, M\}: f_k(\mathbf{p}_i) < f_k(\mathbf{p}_j) \quad (4.10)$$

This concept leads to the definition of *Pareto optimality*: a solution  $\mathbf{p}_i^* \in \Omega$  ( $\Omega$  is the solution space) is said to be *Pareto optimal* if and only if there exists no other solution  $\mathbf{p}_j \in \Omega$  that dominates  $\mathbf{p}_i^*$ . The latter is said to be *nondominated*, and the set of all nondominated solutions forms the *Pareto front* of optimal solutions.

Once the Pareto front has been identified, a single solution has to be selected from the set of nondominated solutions. For this purpose, different strategies can be adopted. In this study, we suggest to choose the median solution, in order to maintain a tradeoff between the different criteria. In case that it is necessary to extract  $N_s$  solutions simultaneously, the  $N_s$  solutions closest to the median one are considered.

An example of the Pareto front is given in Fig. 4.2, in which the optimization of two criteria  $f_1$  and  $f_2$  is involved. The nondominated solutions are drawn with red circles, while the selected (median) solution is represented in green. Other dominated solutions are drawn with black crosses.

## 4.3. Experiments

### 4.3.1. Data Set Description

In order to validate the proposed active learning strategies, experiments were conducted on two multispectral VHR remote sensing images acquired by QuickBird with 0.6 m resolution. The data sets reflect different types of urban settlements at different levels of complexity. Details of sites and images are reported in Table 4.I.

The first data set was acquired in 2002 and refers to a portion of the city of Las Vegas (Nevada). The scene, shown in Fig. 4.3(a), contains regular criss-crossed roads and examples of buildings with similar heights (about one or two floors) but different dimensions, from small residential houses to large commercial structures. It represents a common American sub-urban landscape, including small houses and large roads, which is different from the European style of old cities built with more complex structures.

To take into account this second situation, a second test area acquired in 2004, shown in Fig. 4.3(c), was used including a sub-urban scene of Rome (Italy) composed of a more complex urban structure with buildings showing a variety in heights (from four floors to twelve), dimensions and shapes including apartment blocks and towers. In particular, the Rome image has two completely different urban architectures separated by a railway. The area located in the upper right part of the scene was built during the 60s; buildings are very close to each other and have a maximum of five floors, while roads are. The other side of

TABLE 4.I  
CHARACTERISTICS OF THE IMAGES USED FOR THE EXPERIMENTS

Site information		Image information		
Location	Dimension [pixels]	Satellite	Acquisition date	Spatial resolution [m]
Las Vegas (Nevada)	756x723	QuickBird	May 10, 2002	0.6
Rome (Italy)	1188x973	QuickBird	July 19, 2004	0.6

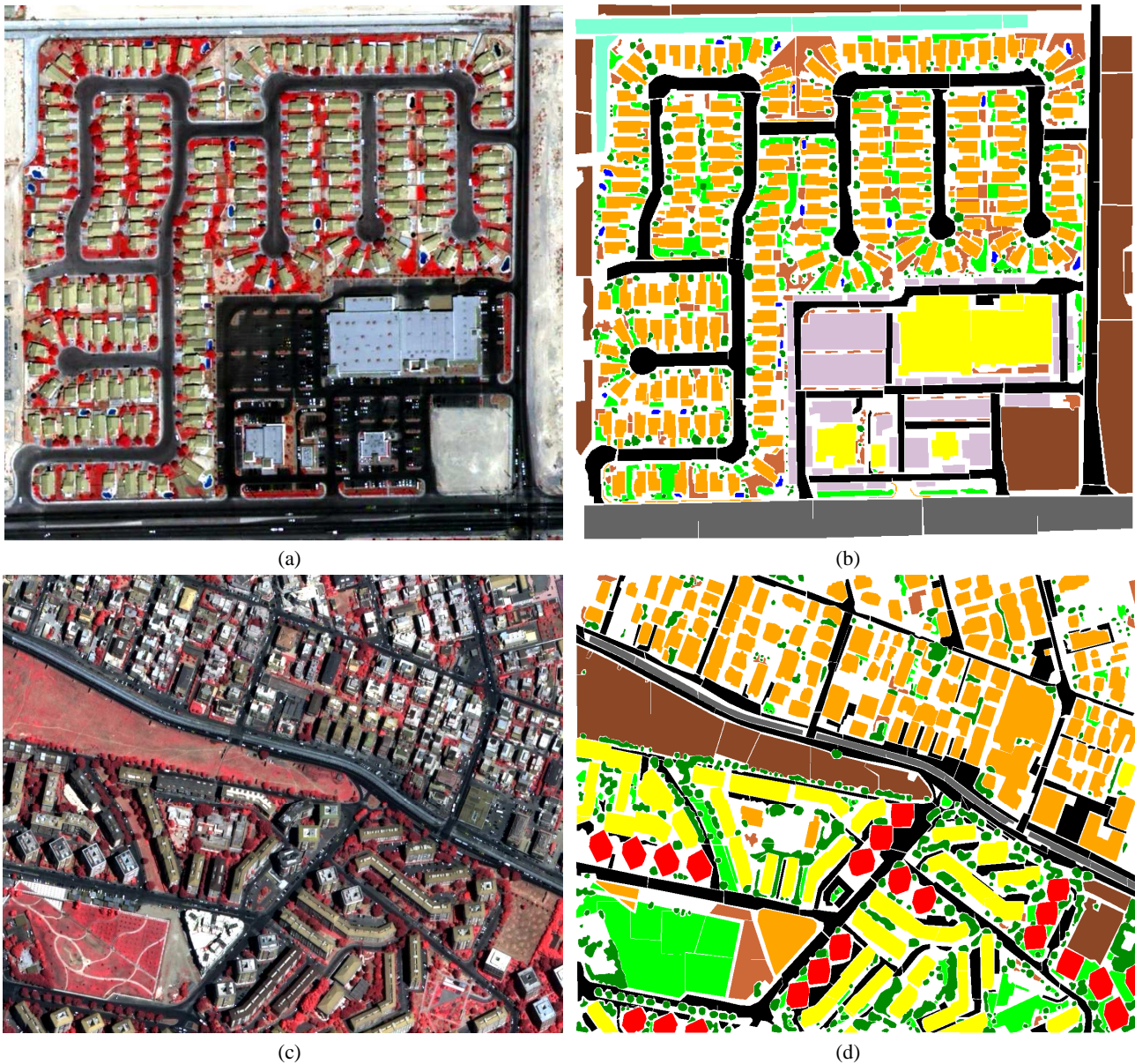


Fig. 4.3. Data sets used for the experiments. False-color image for (a) the Las Vegas and (c) the Rome data sets. Ground-truth for (b) the Las Vegas and (d) the Rome data sets.

the railway was developed during the 80s and 90s; buildings have a variety of architectures, from apartment blocks (eight floors) to towers (twelve floors), while roads are wider than those on the other side of the railroad tracks.

Several different surfaces of interest have been identified, many of which are particular to the specific scene. For the Las Vegas data set, one goal was to distinguish the different uses of the asphalt surfaces, which included *Roads* (i.e., roads that link different residential houses), *Highways* (i.e., roads with more than two lanes) and *Parking lots*. An unusual structure within the scene was a *Drainage channel* located in the



upper part of the image. A further discrimination was made between *Residential houses* and *Commercial buildings*, and between *Bare soil* (terrain with no use) and *Soil* (generally, backyards with no vegetation cover). Finally, more traditional classes, such as *Trees*, *Short vegetation* and *Water* were added for a total of eleven classes of land-use. The areas of shadow were very limited in the scene since the modest heights of buildings and relative sun elevation ( $65.9^\circ$ ).

Due to the dual nature of the architecture of the Rome test case, the selection of the classes was made to investigate the potential of discriminating between structures with different heights, including *Buildings* (structures with a maximum of 5 floors), *Apartment blocks* (rectangular structures with a maximum of 8 floors) and *Towers* (more than 8 floors). As for the previous case, other surfaces of interest were recognized, including *Roads*, *Trees*, *Short vegetation*, *Soil*, *Bare soil* and the peculiar *Railway* for a total of nine classes. Differently from the previous case, in this scene shadows occupy a larger portion of the image.

The ground-truths are reported in Fig. 4.3(b) and Fig. 4.3(d) for the Las Vegas and Rome data sets, respectively. They have been obtained by careful visual inspection of separate data sources, including aerial images, cadastral maps and *in situ* inspections (for the Rome scene only). An additional consideration regards objects within shadows that reflect little radiance because the incident illumination is occluded. These surfaces were assigned to one of the corresponding classes of interest described above. When classifying images at sub-meter spatial resolution, many of the errors may occur in the boundaries between objects. On the other hand, often it is not possible to correctly identify an edge. To limit this effect, we defined the two ground-truths by not including boundary areas.

We note that for both data sets several classes have very similar spectral signatures. In order to differentiate them, we applied on the original images contextual filters based on mathematical morphology [33], which have shown to have desirable properties when applied to urban VHR classification problems [15], [20]. In particular, four very common morphological filters have been considered: opening (O), closing (C), opening by reconstruction (OR), and closing by reconstruction (CR). For each of these filters, we used a structuring element (SE) whose dimensions increased from 9 to 25 pixels with steps of 2 pixels, resulting in 9 morphological features. The size of the SEs has been chosen according to the image resolution. For the Las Vegas data set, a square SE has been used in order to take into account the major direction of the objects on the image, which are  $0^\circ$  and  $90^\circ$ . For the Rome data set, being characterized by an overall  $45^\circ$  angle in the disposition of the objects, a diamond-shaped SE has been used instead. This shape allows a better reconstruction of the borders of the objects in the case of O and C features. The process of reconstruction for OR and CR operators has been performed using a small (3-pixel diameter) SE. The entire process of morphological filtering increases the dimensionality of the data sets from 4 to 40 features.












### 4.3.2. Experimental Setup

In all the following experiments, for both data sets, all the available samples were split in two sets, corresponding to learning set  $U$  and test set. The detailed numbers of learning and test samples are reported in Table 4.II. The initial training samples were selected randomly from the learning set  $U$ . For the first data set, starting from 55 samples, i.e., 5 samples per class, the active learning algorithm was run until the number of training samples was equal to 7995, adding 20 samples at each iteration. Analogously, for the second data set, starting from 36 samples, i.e., 4 sample for each class, 20 samples were added at each iteration up to 11996 samples. The entire active learning process was run ten times, each time with a different initial training set to yield statistically reliable results. At each run, the initial training samples were chosen in a completely random way.










Classification performances were evaluated in terms of several measures: 1) the overall accuracy (OA), which is the percentage of correctly classified samples among all the considered samples, independently of the classes they belong to; 2) the Kappa statistic [34]; 3) the classification accuracies obtained for the different classes; 4) the average accuracy (AA), which is the average over the classification accuracies



TABLE 4.II  
NUMBER OF INITIAL TRAINING, LEARNING, AND TEST SAMPLES FOR (A) THE LAS VEGAS AND (B) THE ROME DATA SETS

(a)			
Class	# learning samples	# test samples	
	<b>Bare soil</b>	4276	48908
	<b>Commercial buildings</b>	1831	20938
	<b>Drainage channel</b>	1149	13138
	<b>Highways</b>	2851	32594
	<b>Parking lots</b>	2269	25939
	<b>Residential houses</b>	7044	80546
	<b>Roads</b>	6130	70088
	<b>Short vegetation</b>	1803	20611
	<b>Soil</b>	1480	16918
	<b>Trees</b>	1049	11989
	<b>Water</b>	118	1354
	<b>Total</b>	30000	343023

(b)			
Class	# learning samples	# test samples	
	<b>Apartment blocks</b>	7081	102735
	<b>Bare soil</b>	5241	76031
	<b>Buildings</b>	11688	169568
	<b>Railway</b>	1036	15024
	<b>Roads</b>	10545	152992
	<b>Short vegetation</b>	4489	65128
	<b>Soil</b>	971	14086
	<b>Tower</b>	3089	44827
	<b>Trees</b>	5860	85020
	<b>Total</b>	50000	725411

obtained for the different classes; 5) the standard deviations ( $\sigma$ ) of OA, Kappa and AA, in order to evaluate the stability of the active learning method.

An SVM classifier was also trained on the entire learning set in order to have a reference-training scenario, called “full” training. On one hand, the classification results obtained in this way represent an upper bound for the accuracies. On the other hand, we expect that the lower accuracy bound will be given by the completely random selection strategy (R). We recall that the purpose of any active learning strategy is to converge to the performance of the “full” training scenario faster than the R method. Moreover, the proposed strategy for spectral-spatial information integration is compared to the performances given by the MS method based on spectral information only.

### 4.3.3. Experimental Results

Considering the Las Vegas data set, the OA for the “full” classifier is equal to 95.47%. In Fig. 4.4(a),(c),(e) we show the results in terms of OA, Kappa, and AA in function of the number of training samples for the proposed active learning strategies, the MS, and the random ones. First, it is evident how the active selection of the training samples allows us a faster convergence to the “full” accuracy with respect to the random strategy. Comparing the different active learning strategies, we note that the integration of the spatial information is useful in the process of training sample collection. In particular, the strategies based on the criteria Sp1 and Sp2 converge to the “full” accuracy using about 5000 training samples, which represent about 17% of the entire learning set. Instead, about 6000 and 7000 samples are necessary for the methods based on the criterion Sp3 and the spectral information only, respectively. Moreover, we note that, before convergence, the proposed strategies give an improvement with respect to the traditional MS criterion. This means that similar values of accuracies can be obtained using a minor quantity of training samples, which

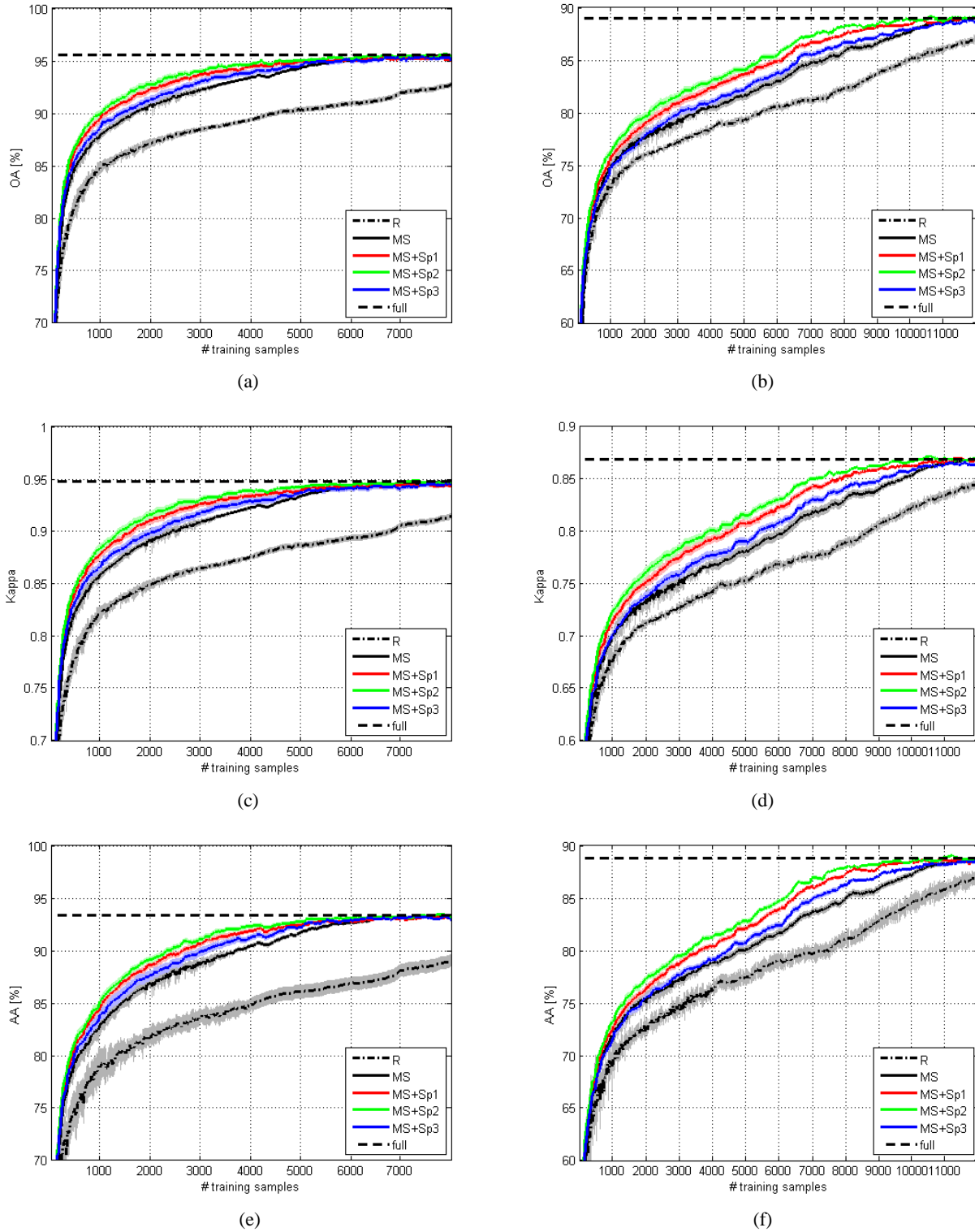


Fig. 4.4. Performances achieved on (a), (c), (e) the Las Vegas and (b), (d), (f) the Rome data sets in terms of (a), (b) OA, (c), (d) Kappa, (e), (f) AA. Each graph shows the results in function of the number of training samples and averaged over ten runs of the algorithm, each with a different initial set. The shaded areas show the standard deviation over the ten considered runs. R = random, MS = margin sampling, Sp1-Sp3 = spatial criterion, full = full SVM.

implies a reduction of the manual work for sample labeling and a decreasing of the computational time necessary to train the classifier.

The obtained results are shown in greater detail in Table 4.III(a). In particular, we considered the performances obtained after 50 and 100 iterations of the iterative process, which corresponds to 1035 and

TABLE 4.III

OA, KAPPA, AA, CLASS ACCURACIES, AND STANDARD DEVIATIONS ( $\sigma$ ) ACHIEVED ON (A) THE LAS VEGAS AND (B) THE ROME DATA SETS

(a)

Method	Full	Initial	R	MS	MS +Sp1	MS +Sp2	MS +Sp3	R	MS	MS +Sp1	MS +Sp2	MS +Sp3
#training samples	30000	55	1035					2035				
OA	95.47	58.98	84.89	88.09	89.73	90.25	88.83	87.18	90.54	92.13	92.61	91.21
$\sigma_{OA}$	-	5.74	0.56	0.40	0.24	0.25	0.40	0.42	0.89	0.19	0.19	0.35
Kappa	0.947	0.533	0.823	0.860	0.880	0.886	0.870	0.850	0.889	0.908	0.914	0.897
$\sigma_{KAPPA}$	-	0.060	0.007	0.005	0.002	0.002	0.004	0.005	0.010	0.001	0.001	0.003
AA	93.35	59.33	79.22	83.15	84.86	85.37	83.98	82.00	86.55	88.39	88.93	87.46
$\sigma_{AA}$	-	4.10	1.47	0.80	0.38	0.42	0.76	1.00	1.07	0.27	0.28	0.61
Bare soil	99.53	65.74	98.09	98.30	99.32	99.41	99.10	98.37	97.45	98.39	98.42	98.36
Commercial buildings	98.22	72.86	88.95	93.36	94.09	94.61	93.28	91.10	95.52	96.31	96.85	95.38
Drainage channel	99.43	58.61	91.82	96.42	97.14	97.78	96.04	95.03	97.48	98.18	98.53	97.29
Highways	97.22	45.50	86.68	90.77	93.77	94.23	91.85	90.65	93.85	96.06	96.55	94.34
Parking lots	86.73	52.67	63.14	66.11	68.96	69.83	67.52	64.83	73.68	76.77	78.04	75.03
Residential houses	97.76	61.27	91.25	94.11	96.41	96.83	95.69	93.21	95.31	97.38	97.54	96.66
Roads	96.26	59.98	87.99	91.00	91.64	92.37	90.67	89.56	92.88	93.61	94.27	92.59
Short vegetation	91.06	60.03	75.56	81.52	83.72	83.96	82.67	79.21	83.91	86.49	87.03	85.35
Soil	88.26	42.66	51.59	57.31	57.67	58.71	56.82	58.82	69.23	69.15	70.37	68.02
Trees	82.39	55.96	62.03	66.74	69.49	70.00	68.80	65.48	70.07	74.13	74.78	73.18
Water	90.03	77.32	74.33	78.99	81.21	81.37	81.31	75.70	82.70	85.81	85.85	85.88

(b)

Method	Full	Initial	R	MS	MS +Sp1	MS +Sp2	MS +Sp3	R	MS	MS +Sp1	MS +Sp2	MS +Sp3
#training samples	50000	36	2016					4016				
OA	88.89	40.49	75.91	77.77	78.95	79.73	77.80	78.54	80.50	82.30	83.03	81.04
$\sigma_{OA}$	-	4.50	0.27	0.50	0.19	0.21	0.42	0.33	0.42	0.08	0.09	0.24
Kappa	0.868	0.322	0.713	0.735	0.753	0.762	0.739	0.744	0.768	0.792	0.800	0.777
$\sigma_{KAPPA}$	-	0.043	0.003	0.006	0.002	0.002	0.005	0.004	0.005	0.001	0.001	0.002
AA	88.74	45.50	72.89	75.47	76.52	77.28	75.46	76.50	78.80	80.47	81.21	79.25
$\sigma_{AA}$	-	2.98	0.76	0.64	0.31	0.30	0.58	0.72	0.39	0.10	0.09	0.25
Apartment blocks	82.18	15.97	59.90	63.38	65.00	65.82	63.16	64.69	68.35	71.04	71.98	68.92
Bare soil	95.82	75.86	91.93	91.73	93.94	94.05	93.48	92.85	92.81	95.26	95.41	94.67
Buildings	88.00	24.02	73.88	75.87	75.63	76.77	73.92	75.88	79.86	79.71	80.94	77.90
Railway	96.61	78.59	91.02	90.15	93.12	93.44	92.27	92.99	91.64	94.50	94.85	93.55
Roads	92.82	57.20	89.21	89.14	92.04	92.42	91.33	89.86	88.66	92.54	92.63	91.93
Short vegetation	87.70	42.72	70.42	75.51	72.64	74.28	71.56	74.87	79.62	77.69	79.22	76.25
Soil	88.45	51.36	57.99	66.53	65.68	66.39	64.67	65.64	72.44	73.07	73.70	71.79
Tower	74.18	24.07	34.11	39.02	40.32	41.44	39.09	43.38	47.81	48.67	50.34	47.26
Trees	92.89	39.73	87.57	87.89	90.34	90.94	89.67	88.35	88.04	91.71	91.79	90.97

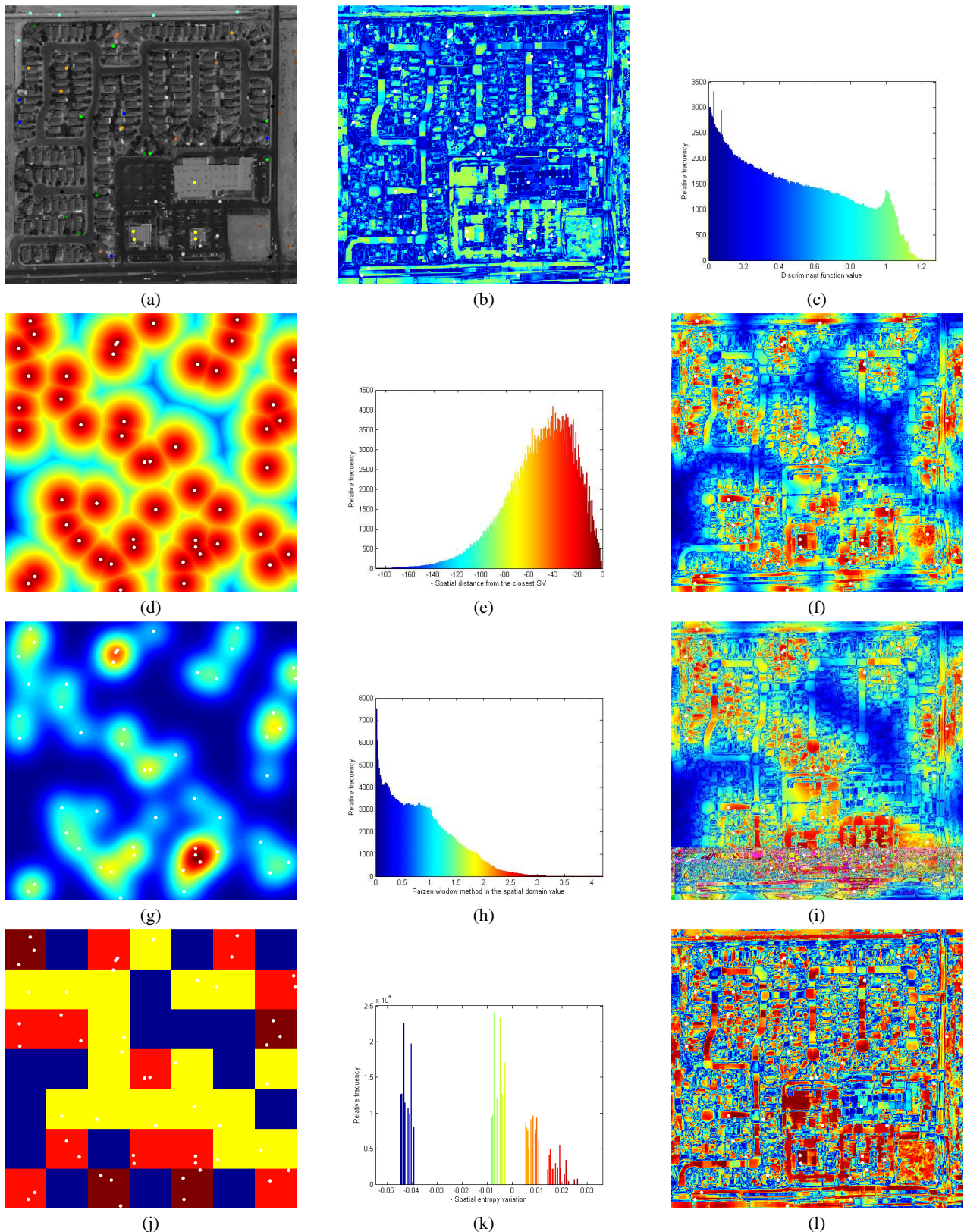


Fig. 4.5. Maps for the Las Vegas data set in terms of (a) set of initial training samples, (b) criterion based on the discriminant function value (MS) and (c) relative histogram, (d) criterion based on the spatial distance from the closest SV (Sp1) and (e) relative histogram, (f) combined criterion (MS+Sp1), (g) criterion based on the Parzen window method in the spatial domain and (h) relative histogram, (i) combined criterion (MS+Sp2), (j) criterion based on the spatial entropy variation (Sp3) and (k) relative histogram, (l) combined criterion (MS+Sp3).

2035 samples used to train the classifier, respectively. We report the values of OA, Kappa, AA, standard deviations associated with the accuracies ( $\sigma_{OA}$ ,  $\sigma_{KAPPA}$ , and  $\sigma_{AA}$ ), and class accuracies. As it can be seen, the proposed strategies are characterized by a better performance with respect to the MS criterion from different point of views. First, better values of accuracies (OA, Kappa, AA, class accuracies) are obtained using the same number of training samples. Then, better values of standard deviations associated with the accuracies are verified. Indeed, smaller values of standard deviation mean that the proposed strategies exhibit a greater level of stability with respect to the random selection of the initial training set.

To better understand the proposed strategies, a set of maps are depicted in Fig. 4.5. In Fig. 4.5(a) we report the grey-level representation of the remote sensing image with an example of training sample set. In particular, for such analysis we considered the initial training set that gives the value of OA more close to the mean value obtained at the first iteration, i.e., equal to 58.98 as reported in Table 4.III. The training samples are depicted with circles colored with the corresponding class colors. In Fig. 4.5(b) we report the map of the discriminant function value, which is given for each sample  $j$  of the image by the minimum absolute value of the discriminant function  $f_{MIN,j}$  obtained by training the classifier on the considered set of training samples. In this map SVs are highlighted with white circles. The correspondence between the discriminant function value and the color is given in Fig. 4.5(c), in which the histogram of the map is reported. In particular, for completeness, the histogram is not referred to the considered single map only, but has been obtained by averaging the results on the ten different experiment runs. It is evident how many samples are associated with very low values of discriminant function, which are depicted in dark blue, and therefore are placed in the proximity of the boundary between different classes. This map corresponds to the map associated with the MS criterion, in which the samples more close to the boundary are selected. Similarly, we report in the other figures the maps associated with the spatial information. In particular, the map and the relative histogram associated with the criterion based on the spatial distance from the closest SV (Sp1) are shown in Fig. 4.5(d),(e), respectively. It appears how dark blue samples are placed in area of the image not covered by SVs. In Fig. 4.5(f) we illustrate the final map obtained by combining the spectral MS and the spatial Sp1 criteria. Again, we show with dark blue color the samples to select, i.e., in this case the samples belonging to the set of nondominated solutions. Similarly, in Fig. 4.5(g),(h) we report the map and the relative histogram associated with the criterion based on the Parzen window method in the spatial domain (Sp2). The map that combines the MS and the Sp2 criteria is depicted in Fig. 4.5(i). We note that this map appears similar to that represented in Fig. 4.5(f). This justifies similar performances of the two different criteria as described previously. Finally, the maps and the histogram related to the Sp3 strategy, in which the maximization of the spatial entropy variation is desired, are illustrated in Fig. 4.5(j)-(l).

The analysis of the discriminant function value allows us to conduct further considerations on the different compared strategies. In particular, the discriminant function value represents an information related to the reliability of the class estimation given by the classification process. Considering the Fig. 4.5(b),(c), in which the map of the discriminant function and the relative histogram associated with the initial training set are depicted, we highlighted previously how most of the samples have been characterized by low values of discriminant function. This means that they have been estimated with poor levels of confidence. This aspect is confirmed by the fact that very few training samples have been used to construct the classification model. We note that the histogram has approximately a monotonous decreasing. In Fig. 4.6(a),(b), we illustrate the map of the discriminant function and the relative histogram using the “full” training set. In this case, in which a high number of training samples has been considered, the samples are characterized by high discriminant function values and thus high confidence. The histogram is completely different with respect to that shown in Fig. 4.5(c). We have not a monotonous decreasing, but a peak placed in correspondence of a relatively high discriminant function value. In order to perform a more detailed analysis, we consider two measures which are the following: 1) the mean value of the discriminant function calculated on the entire map and 2) the standard deviation of the discriminant function value normalized with respect to the



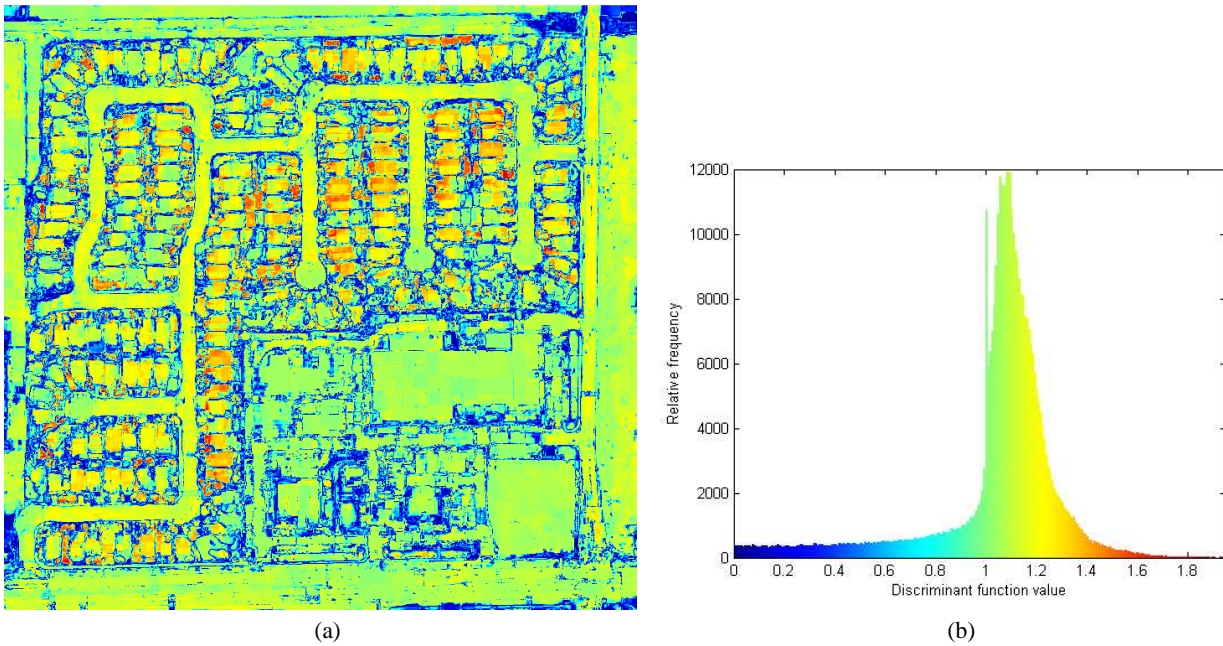


Fig. 4.6. Maps for the Las Vegas data set in terms of (a) discriminant function value and (b) relative histogram for the full SVM.

corresponding mean value. In Fig. 4.7(a),(b), we report the results in function of the number of training samples for the proposed active learning strategies, the MS, and the random ones in terms of discriminant function mean value and normalized discriminant function standard deviation, respectively. For a better visualization, we show the results until 200 iterations of the iterative process. Considering the initial training set, which corresponds to the starting point of the curves, it is confirmed that the discriminant function has a low mean value and a high standard deviation. In particular, a high value of standard deviation implies that the confidence map tends to be not homogenous, i.e., some samples are classified with low reliability and other ones with high confidence. This result is not desirable, because the classification process tends to classify the samples with different levels of reliability. Instead, using the “full” training set, we have an increment of the mean value and a substantial decrement of the standard deviation value, which comes from a confidence map more homogenous. This scenario tends to the “ideal” case, in which all the samples are classified with high reliability. This situation involves a confidence map approximately homogenous and with high values of discriminant function. Considering the different selection strategies, we note that active learning methods are able to increment the discriminant function mean value faster than the random selection. Moreover, a faster convergence to the “full” result has been obtained using the two proposed strategies that combine the MS criterion with the Sp1 and Sp2 criteria. These two heuristics allow us to have quicker decrements of the standard deviation value also, which have been verified since the first iterations. Adopting the MS criterion only, an improvement with respect to the random strategy has been obtained only when about 1500 samples have been added to the training set. These results show how the integration of the spatial information in the active learning process allows us to obtain better performance not in terms of accuracy only, but also in terms of classification reliability, which can be estimated by analyzing the discriminant function map. The mean and standard deviation values are summarized in Table 4.IV(a), in which the results obtained after 50 and 100 iterations of the iterative process are reported.

Concerning the Rome data set, the results confirm the observations done for the Las Vegas one. The graphs with the accuracies in function of the number of training samples are illustrated in Fig. 4.4(b),(d),(f). For the “full” classifier the OA is equal to 88.89. Also for this set of experiments, the proposed active learning strategies give a faster convergence to the “full accuracy” and better performances before convergence with respect to the random and the MS methods. The criteria Sp1 and Sp2 allow to converge to

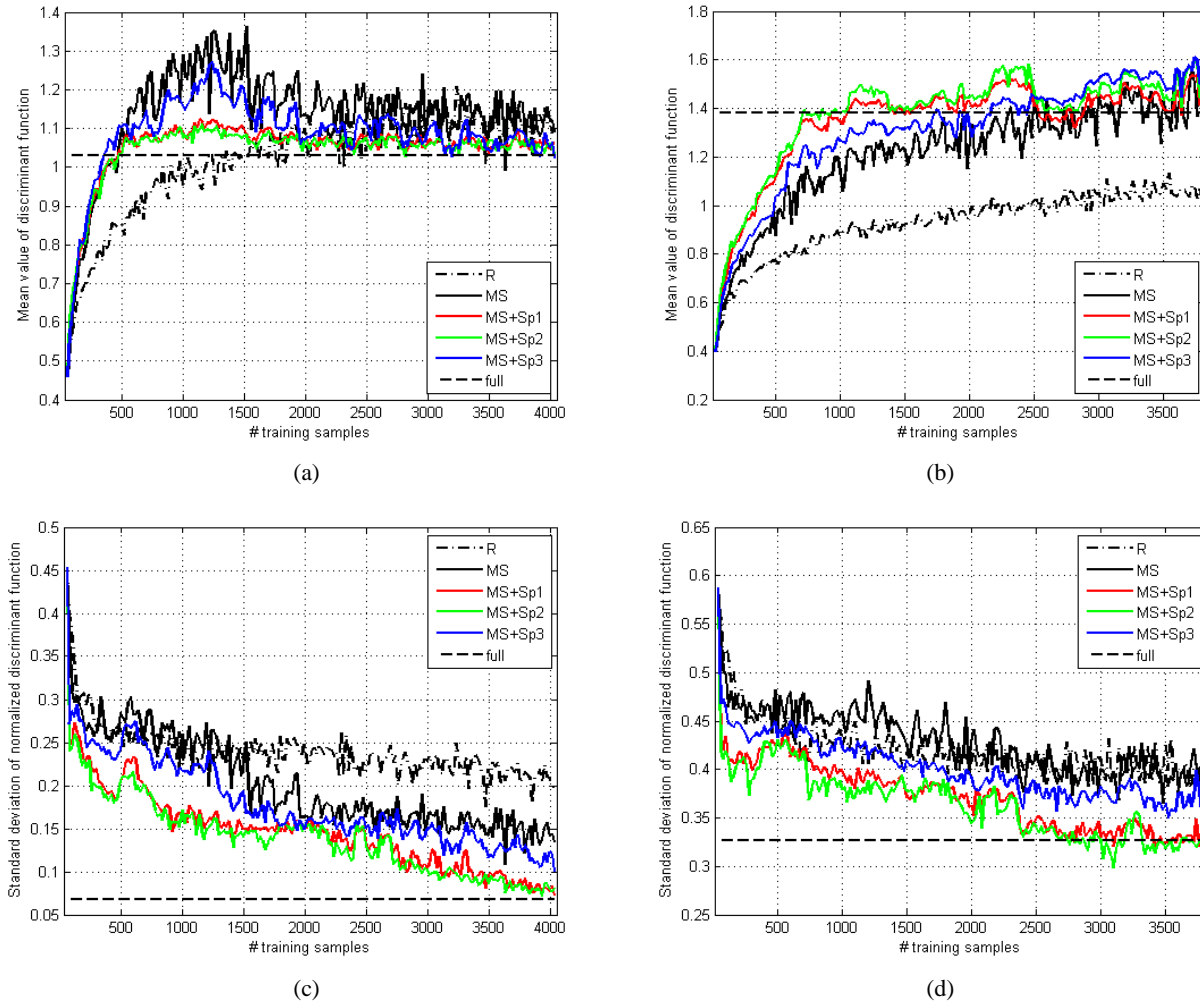


Fig. 4.7. Results achieved on (a), (c) the Las Vegas and (b), (d) the Rome data sets in terms of (a), (b) mean value of discriminant function, (c), (d) standard deviation of discriminant function. Each graph shows the results in function of the number of training samples and averaged over ten runs of the algorithm, each with a different initial set. R = random, MS = margin sampling, Sp1-Sp3 = spatial criterion, full = full SVM.

the “full” accuracy using about 9000 training samples, while about 11000 samples are necessary for the methods based on the criterion Sp3 and the spectral information only. The results obtained after 100 and 200 iterations of the active learning process, which correspond to 2016 and 4016 training samples respectively, are summarized in Table 4.III(b).

In terms of discriminant function value, the results in function of the number of training samples until 190 iterations of the iterative process are shown in Fig. 4.7(b),(d). The criteria Sp1 and Sp2 confirm the best performance both in terms of mean value increasing and standard deviation value decreasing. In particular, the mean and standard deviation values obtained after 100 and 200 iterations are reported in Table 4.IV(b).

## 4.4. Conclusion

In this chapter, the active learning approach has been considered to solve the problem of training sample collection for classification of remote sensing images. While the active learning strategies presented in the literature work in the spectral domain only, we have proposed to combine spectral and spatial information in the iterative process of active sample selection. For this purpose, we introduced three different criteria in the spatial domain in order to favor the selection of samples distant from the samples already

TABLE 4.IV  
MEAN VALUE AND STANDARD DEVIATION OF THE DISCRIMINANT FUNCTION ACHIEVED ON  
(A) THE LAS VEGAS AND (B) THE ROME DATA SETS

(a)

Method	Full	Initial	R	MS	MS +Sp1	MS +Sp2	MS +Sp3	R	MS	MS +Sp1	MS +Sp2	MS +Sp3
#training samples	30000	55	1035					2035				
Mean value	1.03	0.46	1.02	1.23	1.07	1.07	1.16	1.02	1.11	1.05	1.05	1.08
Standard deviation	0.06	0.45	0.25	0.25	0.16	0.17	0.22	0.22	0.18	0.16	0.15	0.16

(b)

Method	Full	Initial	R	MS	MS +Sp1	MS +Sp2	MS +Sp3	R	MS	MS +Sp1	MS +Sp2	MS +Sp3
#training samples	50000	36	2016					4016				
Mean value	1.38	0.40	1.00	1.34	1.41	1.46	1.29	1.09	1.37	1.37	1.42	1.47
Standard deviation	0.33	0.59	0.42	0.44	0.35	0.35	0.38	0.40	0.38	0.33	0.32	0.35

composing the current training set. The three criteria are based on Euclidean distances, Parzen window method, and entropy variation, respectively.

In order to validate the proposed approach, we conducted experiments on two VHR images acquired by Quickbird. The obtained results show good capabilities of the proposed approach for the selection of significant samples. In particular, advantages in terms of classification accuracy and classification reliability have been empirically evaluated with respect to strategies that do not exploit spatial information. Therefore, the integration of spatial information has shown worthy for reducing the manual sample labeling work and to decrease the computational time necessary to train the classifier.

The main drawback of the proposed approach is represented by an increment of the computational cost, given by the calculation of further measures in order to take into account the spatial contribution.

While in this work we considered, for its simplicity and effectiveness, the state-of-the-art MS strategy as spectral heuristic, the proposed approach can be in general applied in conjunction with any traditional active learning method that exploits the samples in the spectral domain.

## 4.5. Acknowledgment

The authors would like to thank DigitalGlobe for providing the data sets used in this paper.

## 4.6. References cited in Chapter 4

- [1] P. Mitra, C. A. Murthy, and S. K. Pal, "A probabilistic active support vector learning algorithm," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 3, pp. 413–418, Mar. 2004.
- [2] E. Pasolli and F. Melgani, "Active learning methods for electrocardiographic signal classification," *IEEE Trans. Inf. Technol. Biomed.*, vol. 14, no. 6, pp. 1405–1416, Nov. 2010.
- [3] P. Mitra, B. Uma Shankar, and S. Pal, "Segmentation of multispectral remote sensing images using active support vector machines," *Pattern Recogn. Lett.*, vol. 25, no. 9, pp. 1067–1074, Jul. 2004.
- [4] Y. Zhang, X. Liao, and L. Carin, "Detection of buried targets via active selection of labeled data: Application to sensing subsurface UXO," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 11, pp. 2535–2543, Nov. 2004.



- [5] M. Ferecatu and N. Boujemaa, “Interactive remote-sensing image retrieval using active relevance feedback,” *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 4, pp. 818–826, Apr. 2007.
- [6] S. Rajan, J. Ghosh, and M. Crawford, “An active learning approach to hyperspectral data classification,” *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 4, pp. 1231–1242, Apr. 2008.
- [7] Q. Liu, X. Liao, and L. Carin, “Detection of unexploded ordnance via efficient semisupervised and active learning,” *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 9, pp. 2558–2567, Sep. 2008.
- [8] D. Tuia, F. Ratle, F. Pacifici, M. Kanevski, and W. Emery, “Active learning methods for remote sensing image classification,” *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 7, pp. 2218–2232, Jul. 2009.
- [9] D. Tuia, M. Kanevski, J. Muñoz Marí, and G. Camp-Valls, “Cluster-based active learning for compact image classification”, in *Proc. IGARSS*, Honolulu, HI, Jul. 2010, p. 2824–2827.
- [10] J. Li, J. Bioucas-Dias, and A. Plaza, “Semisupervised hyperspectral image segmentation using multinomial logistic regression with active learning,” *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 11, pp. 4085–4098, Nov. 2010.
- [11] B. Demir, C. Persello, and L. Bruzzone, “Batch-mode active-learning methods for the interactive classification of remote sensing images,” *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 3, pp. 1014–1031, Mar. 2011.
- [12] E. Pasolli, F. Melgani, and Y. Bazi, “SVM active learning through significance space construction”, *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 3, pp. 431–435, May 2011.
- [13] G. Schohn and D. Cohn, “Less is more: Active learning with support vectors machines,” in *Proc. 17th ICML*, Stanford, CA, 2000, pp. 839–846.
- [14] G. Camps-Valls, L. Gomez-Chova, J. Muñoz-Marí, J. Vila-Francés, and G. Calpe-Maravilla, “Composite kernels for hyperspectral image classification,” *IEEE Geosci. Remote Sens. Lett.*, vol. 3, no. 1, pp. 93–97, Jan. 2006.
- [15] M. Fauvel, J. A. Benediktsson, J. Chanussot, and J. R. Sveinsson, “Spectral and spatial classification of hyperspectral data using SVMs and morphological profiles,” *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 11, pp. 3804–3814, Nov. 2008.
- [16] Y. Tarabalka, M. Fauvel, J. Chanussot, and J. A. Benediktsson, “SVM- and MRF-based method for accurate classification of hyperspectral images,” *IEEE Geosci. Remote Sens. Lett.*, vol. 7, no. 4, pp. 736–740, Oct. 2010.
- [17] Y. Gu, Y. Zhang, and J. Zhang, “Integration of spatial-spectral information for resolution enhancement in hyperspectral images,” *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 5, pp. 1347–1358, May. 2008.
- [18] D. M. Rogge, B. Rivard, J. Zhang, A. Sanchez, J. Harris, and J. Feng, “Integration of spatial-spectral information for the improved extraction of endmembers,” *Remote Sens. Environ.*, vol. 110, pp. 287–303, Oct. 2007.
- [19] M. Zortea and A. Plaza, “Spatial preprocessing for endmember extraction,” *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 8, pp. 2679–2693, Aug. 2009.
- [20] D. Tuia, F. Pacifici, M. Kanevski, and W. J. Emery, “Classification of very high spatial resolution imagery using mathematical morphology and support vector machines,” *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 11, pp. 3866–3879, Nov. 2009.
- [21] F. Pacifici, M. Chini, and W. J. Emery, “A neural network approach using multi-scale textural metrics from very high-resolution panchromatic imagery for urban land-use classification,” *Remote Sens. Environ.*, vol. 113, no. 6, pp. 1276–1292, Jun. 2009.
- [22] D. Tuia, F. Ratle, A. Pozdnoukhov, and G. Camps-Valls, “Multisource composite kernels for urban-image classification,” *IEEE Geosci. Remote Sens. Lett.*, vol. 7, no. 1, pp. 88–92, Jan. 2010.
- [23] G. Mercier and F. Girard-Ardhuin, “Partially supervised oil-slick detection by SAR imagery using kernel expansion,” *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 10, pp. 2839–2846, Oct. 2006.
- [24] F. Melgani and Y. Bazi, “Markovian fusion approach to robust unsupervised change detection in remotely sensed imagery,” *IEEE Geosci. Remote Sens. Lett.*, vol. 3, no. 4, pp. 457–461, Oct. 2006.

- [25] G. Moser and S. B. Serpico, "Unsupervised change detection from multichannel SAR data by markovian data fusion," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 7, pp. 2114–2128, Jul. 2009.
- [26] R. Dianat and S. Kasaei, "Change detection in optical remote sensing images using difference-based methods and spatial information," *IEEE Geosci. Remote Sens. Lett.*, vol. 7, no. 1, pp. 215–219, Jan. 2010.
- [27] G.-J. Wen, J.-J. Lv, and W.-X. Yu, "A high-performance feature-matching method for image registration by combining spatial and similarity information," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 4, pp. 1266–1277, Apr. 2008.
- [28] F. Melgani, "Contextual reconstruction of cloud-contaminated multitemporal multispectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 2, pp. 442–455, Feb. 2006.
- [29] S. Benabdelkader and F. Melgani, "Contextual spatio-spectral postreconstruction of cloud-contaminated images," *IEEE Geosci. Remote Sens. Lett.*, vol. 5, no. 2, pp. 204–208, Apr. 2008.
- [30] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in *Proc. 5th COLT*, Pittsburgh, PA, Jul. 1992, p. 144–152.
- [31] V.N. Vapnik, *Statistical Learning Theory*, New York: Wiley, 1998.
- [32] R. Duda, P. Hart, and D. Stork, *Pattern Classification*, 2nd ed. New York: Wiley, 2001.
- [33] P. Soille, *Morphological image analysis*. Berlin-Heidelberg: Springer-Verlag, 2004.
- [34] J. Cohen, "A coefficient of agreement for nominal scales," *Educational and Psychological Measurement*, vol. 20, no. 1, pp. 37–46, 1960.

## 5. Using Active Learning to Adapt Remote Sensing Image Classifiers

*Abstract – The validity of training samples collected in field campaigns is crucial for the success of land use classification models. However, such samples often suffer from a sample selection bias and do not represent the variability of spectra that can be encountered in the entire image. Therefore, to maximize classification performance, one must perform adaptation of the first model to the new data distribution. In this chapter, we propose to perform adaptation by sampling new training examples in unknown areas of the image. Our goal is to select these pixels in an intelligent fashion that minimizes their number and maximizes their information content. Two strategies based on uncertainty and clustering of the data space are considered to perform active selection. Experiments on urban and agricultural images show the great potential of the proposed strategy to perform model adaptation.*

The work presented in this chapter has been published in the *Remote Sens. Environ.*, vol. 115, no. 9, pp. 2232–2242, September 2011; Co-authors: D. Tuia, W. J. Emery.

## 5.1. Introduction

Today, the access to remote sensing images has been made easier by the availability of images sensed by commercial satellites with short revisit periods. Sensors such as QuickBird or World-View II provide imagery at very high geometrical resolution, thus providing an unprecedented detail in the scenes described and allowing fine reconstruction of urban objects such as buildings. However, such a fine resolution leads to the increase of variability of the classes to be detected. For mid-resolution problem such as land use classification, sub-meter resolution comes with strong intraclass variability caused by geometrical properties of the objects, changes in illumination and details detected only at the higher resolution (e.g., chimneys on buildings).

Even if they are able to treat well-defined classification tasks, the majority of current classification methods relies on supervision and may fail if the data used to build the model (the training set) are not representative of the true distribution generating the classes. Note that when dealing with remote sensing image classification, a user is often confronted with large archives of digital information to be classified and that the spatial extent of such images makes the definition of exhaustive training sets a difficult and time-consuming task. In this sense, providing exhaustive ground truth for large remote sensing images is often not possible. As a consequence, the labeled information only covers a part of the true variability of the class distribution. Moreover, a user can afford only partial ground surveys and can rely on previous studies about the ground cover. This is even more critical when adapting a model to a multitemporal sequence, where differences in illumination and reflectance can make the adaptation of a model fail [1].

These constraints result in the user not having the economic and temporal resources to label the entire area or being confronted to a new classification task including a previously unconsidered and contiguous region in a second moment only. In both cases, one must then focus on subsections of the images, in order to retrieve a coherent training set representing the classes to be described and then apply the model obtained from the sub-image to the entire scene.

This field of investigation is primordial for remote sensing data analysis and has been considered for mid-resolution optical data as signature extension. In the pioneering paper by Fleming et al. [2], the authors studied the effect of clustering the data to account for data multimodality in Gaussian classifiers, thus considering the issue of non-stationary data across the image. This principle has been applied in applications for Landsat imagery [3]-[6]. In [7], the approach proposed in [2] was successfully extended to hyperspectral data, thus showing the interest of considering model adaptation to unsampled areas for this type of imagery.

However, in recent methodological research this aspect has been overlooked by the focus put on the classification of local regions and by claiming that the new algorithms proposed were powerful enough to generalize to unseen areas. A common assumption in such developments became that data are homogeneous throughout the image, i.e. class statistics remain constant over the image. This seems unrealistic, especially when the training set only covers small subsets of the scene. In recent years, emphasis has been put on optimizing the classifiers for situations where the training set is minimal [8]-[11], but the problem of adaptation to slightly varying test distributions has been considered only rarely in recent literature using spectral data. By this, we mean that a shift between the distribution of the training set and the test data has occurred, leading thus to an incompatibility of the model optimized for the first set of observations when they are used to describe the unseen pixels. In the machine learning community, the problem, also known as covariate shift [12], has been considered from different perspectives: by weighting the observations according to the position of the training samples with respect to the support of the test ones [13], [14] or by adding regularizers on the test data distribution [15]. Covariate shift is being considered nowadays in several applications, covering brain computer interfaces [16] or genomic sequence analysis [17]. In remote sensing literature, the field is relatively young: in [18], the samples in the new image are used to assess the class parameters in the expectation maximization algorithm. In [1], a classifier built on an image is updated using

the unlabeled data distribution of another scene in an hyperspectral image classification problem. In [19], this idea is further developed with an iterative procedure adapting a training set to shifted images: the model discards contradictory old training samples and uses the distribution of the new image to adapt the model to the new conditions. Finally, in [20], matching of the first order statistics in a projected space is studied under the name of kernel mean matching: the model is then applied to a series of images for cloud detection.

A strategy to learn the data set shift is to sample additional pixels from the unknown distribution to check if they are consistent with the model obtained from training set generated by partial sampling. In particular, when dealing with very high resolution imagery, the problem of finding pixels lying in the shifted areas can be a difficult task. In this chapter, we propose a simple, yet effective way to correct a training set for its application to a new area where a data set shift may have occurred. We propose to use active queries to learn the shift and sample the areas in which the classifier would become suboptimal, since they do not contain any labeled instance. These methods are new to the remote sensing community [21]-[24], but they are rapidly gaining interest in this community [25]-[27], as they allow one to build an optimal training set with a minimum of queries (or labeled pixels).

Although appealing, the use of active learning for adapting a classifier to new data must be done carefully. Traditional supervised active learning algorithms focus on discrepancies near the classification boundary, resulting in new contradictory areas that may appear in the unseen distribution (the new image). However, such contradictions may happen far from those boundaries, for instance if a new class has appeared. In this case, an active learning algorithm risks failure and can lead to slower convergence than random sampling that may find these regions by chance.

In this chapter, we study the effectiveness of using active learning to detect a data set shift and we pay particular attention to the problem of the appearance of new classes that may not have been observed in the initial training set. To illustrate the proposed strategy, the breaking ties (BT) active sampling proposed in [28] is used with a linear discriminant analysis (LDA), which is a classifier widely used in real applications and also strongly prone to fail in case of covariate shift. Exploration of the data distribution through clustering is also used to cope with common situations, where one or several classes would not have been observed in the training set, but appear in the rest of the image. The proposed approach is tested on two urban and two agricultural remote sensing images, where the relevance of completing an existing training set with smartly selected pixels can be appreciated.

The remainder of the chapter is organized as follows. Section 5.2. presents the problem of covariate shift and the proposed correction based on active learning. Section 5.3. details the data and the setup of the experiments discussed in Section 5.4. Section 5.5. concludes the chapter.

## 5.2. Covariate Shift and Active Learning

This section briefly exposes the problem of covariate shift and converts it to a sampling problem. Active learning is then proposed as an alternative to fill the covariate shift gap. Finally, the problem of exploration is considered and a cluster-based heuristic is proposed to comply with the emergence of new, unexpected, classes.

### 5.2.1. The Problem of Covariate Shift

Covariate shift is a common problem for any statistical model aiming at classifying a series of pixel vectors  $\mathbf{x}$  into a series of land use classes  $y$ . The common assumption that the data are independent and identically distributed (iid) usually does not hold for real applications, since the data distribution  $p_t(\mathbf{x})$  used for training the model only partially represents the true data distribution, that is represented by the test data distribution  $p_n(\mathbf{x})$ . Nonetheless, it is a common assumption for machine learning algorithms to consider that test data follow the same joint probability distribution as the training data, i.e.  $p(y|\mathbf{x})p_t(\mathbf{x})$ , where  $y$  is the

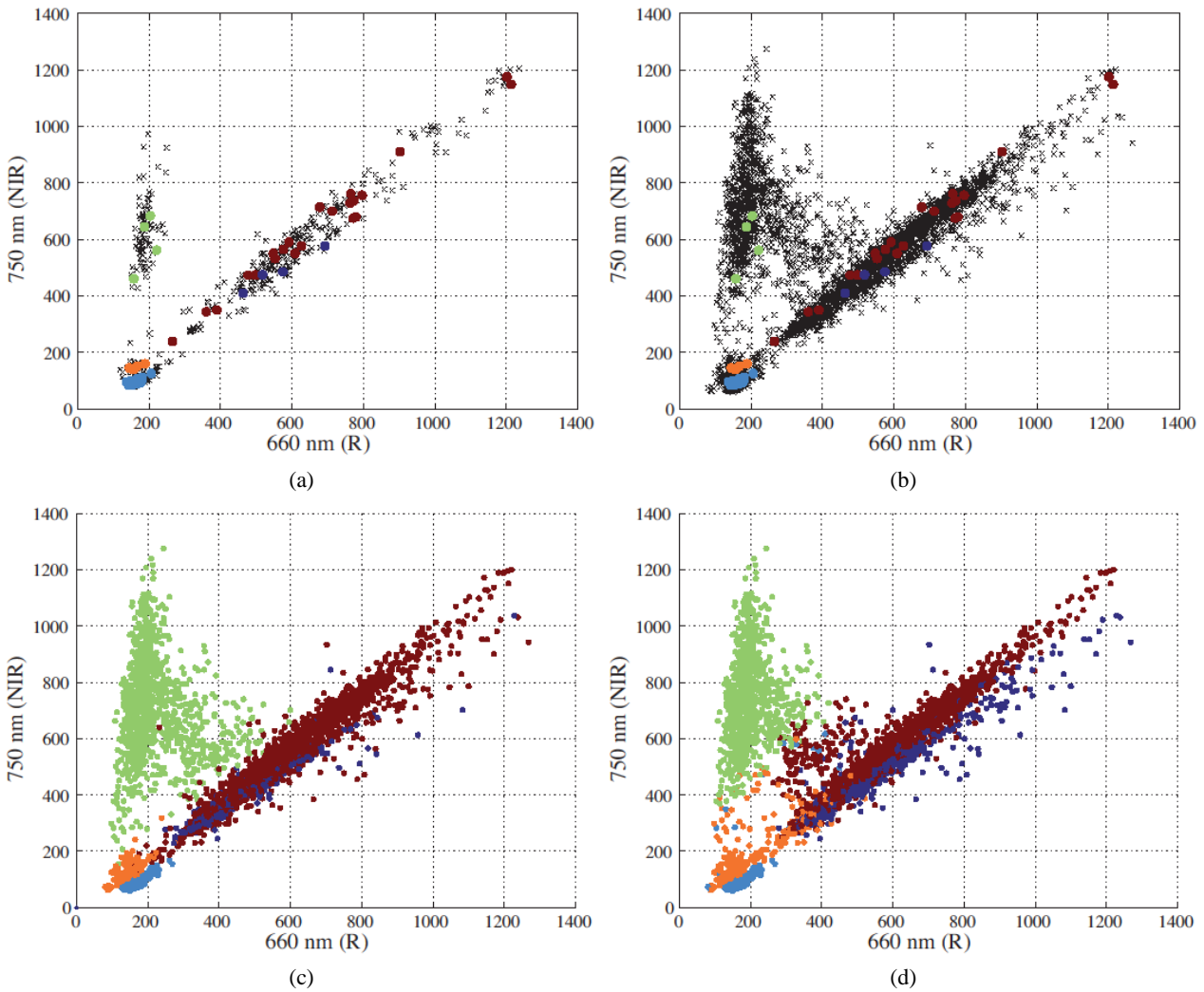


Fig. 5.1. Data set shift problem: (a) the training set (in color) is well suited to describe the unlabeled data (in black); (b) if using these training data to a larger amount of test data, the available training points become suboptimal with respect to the true labeling of the larger test set, shown in (c) subfigure and corresponding to the labeling of the ROSIS image reported in the bottom left part of Fig. 5.4; (d) a classifier such as LDA is thus prone to fail at classifying the test data. The data cloud is the one of the ROSIS image presented in the left part of Fig. 5.4.

class label and  $p(y|\mathbf{x})$  is their conditional distribution. However, there is the risk that the new test data follow a slightly different distribution  $p_{ts}(\mathbf{x}) \approx p_{tr}(\mathbf{x})$ . This situation is known as *covariate shift* and can result in a model that is optimal for a part of the data, but becomes sub-optimal if applied to the entire image. Fig. 5.1. illustrates this phenomenon: a model trained on data coming from a part of a satellite image (the ‘A’ region of Fig. 5.4.) can optimally describe the distribution of this sub-image, represented by the black crosses in Fig. 5.1(a). When this same training set is used to describe the class distribution in the entire image [black crosses of Fig. 5.1(b)], the model fails because some areas of the feature space are not covered by this training set. Some of these areas were not present in the subset image, and represent the shift between the subset and the entire scene. Such a shift is related to differences in geometry that were not taken into account in the first place or to reflectances of the objects that were not covered by the available training set. When using LDA on this data, the true class memberships [shown in Fig. 5.1(c)] are not correctly represented in the outcome of the model [illustrated in Fig. 5.1(d)]: the model built without adaptation models poorly at the interface between classes, thus resulting in an important decrease in the classification performance.

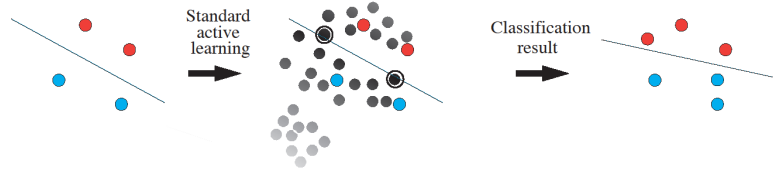


Fig. 5.2. Uncertainty-based active learning algorithm general flowchart: (left) given an incomplete training set, (center) the unlabeled candidates are ranked according to a specific heuristic (represented by the gray tones attributed to the unlabeled pixels); (right) the candidates maximizing the heuristic are labeled and added to the training set.

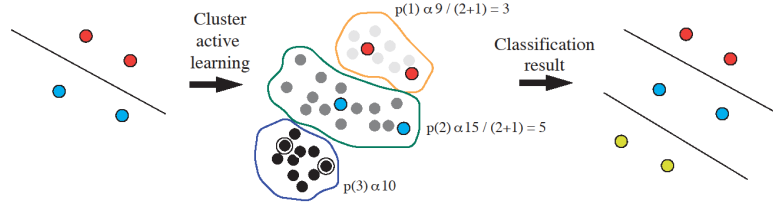


Fig. 5.3. Cluster-based active learning algorithm general flowchart: (left) given an incomplete training set, (center) the unlabeled candidates are ranked according to the heuristic of (5.2) (in the computation, only the numerator is reported); (right) the candidates maximizing the heuristic are labeled and added to the training set, allowing the discovery of a third class.

### 5.2.2. Active Learning to Correct Data Set Shift

Since the training and test distributions come from the same image, illumination conditions do not change and it is rather unlikely to find complex distortions between the two feature spaces: in this case, the shift is to be found in missing parts of the true data distribution [see Fig. 5.1(a)-(b)]. Adapting the classifier trained on the subset to the entire image can be thus seen as efficiently finding the uncovered areas and sample useful pixels to classify them.

This is a typical setting for active learning algorithms [29], which are algorithms aiming at finding efficient training sets to solve classification problems. For this particular problem, active learning results in a search for pixels enhancing the adaptation of the model to the rest of the image, i.e., refining the description of the boundaries between classes.

Active learning algorithms can be briefly summarized as follows (see Fig. 5.2): starting with a suboptimal training set composed by  $n$  pixels  $S = \{\mathbf{x}_i, y_i\}_{i=1}^n$ , an active learning algorithm exploits a ranking criterion, or heuristic, to rank all the  $m$  unlabeled pixels  $U = \{\mathbf{x}_j\}_{j=n+1}^{n+m}$  in order to select the most informative and add them to  $S$ . By so doing, the model is forced to focus on conflicting areas and to improve its generalization capabilities.

In this study, the breaking ties heuristic proposed in [28] is used: for each candidate, the two highest posterior class probabilities are subtracted, forming the ranking criterion that is exploited by the algorithm

$$\hat{\mathbf{x}}^{BT} = \arg \min_{\mathbf{x}_j \in U} \left\{ \max_{\omega \in N} p(y_j^* = \omega | \mathbf{x}_j) - \max_{\omega \in N \setminus \omega^+} p(y_j^* = \omega | \mathbf{x}_j) \right\} \quad (5.1)$$

where  $y_j^*$  is the class prediction for the pixel  $\mathbf{x}_j$ ,  $\omega \in N$  corresponds to one among the  $N$  possible classes and  $\omega^+ = \arg \max_{\omega \in N} \{p(y_j^* = \omega | \mathbf{x}_j)\}$  is the most probable class for pixel  $\mathbf{x}_j$ .

After ranking, the pixels minimizing (5.1) are then taken from the  $U$  set, labeled by the user, and finally added to the current training set  $S = \{S \cup \hat{\mathbf{x}}^{BT}\}$ . This heuristic uses the following intuition: the more a pixel shows a similar posterior probability between the two most probable classes, the more it is uncertain

and thus capable of providing useful information if added to the training set. In previous experiments the BT approach has shown to be capable of providing good performance with remote sensing data [30].

### 5.2.3. On the Need of an Exploration-Focused Heuristic

Using active queries to learn data sets seems an appealing solution for the classification of remote sensing data. However, the use of such models must be handled with care, since it relies on the quality of the initial training set (in our case, the available labeled pixels in the subimage). If these pixels do not cover the entire distribution of the classes (which is reasonable in a covariate shift setting), there is also the possibility that a class will be ignored in the available training set. Consider again Fig. 5.2: in the central plot, there is a cluster of pixels in the bottom left part of the distribution. A traditional active learning algorithm, since it focuses on the uncertainty in the vicinity of the classification boundary only, will never check on the uncertainty of this region, since it is related to the data structure and not the current model uncertainty. As a consequence, this cluster will never be sampled by such an active algorithm. This may be problematic if this cluster corresponds to a new, unknown class. Approaches trying to constrain traditional heuristic to make them explore the feature space have been proposed in [31],[24], but they focus on the classification boundary and thus will also fail in this context.

Another view can be gained by using general data clustering, as in [32],[33]: to cover the entire data distribution, we proceed to a pre-clustering of the image in a given number of clusters to decide whether there are some unexplored areas of the image. Contrary to these results, this process is not intended to create the initial training set, since a fair amount of labeled data are already available. Therefore, this knowledge about the availability of labeled samples can be used to direct sampling. We use a cost function aware of the presently available training samples, in the sense of [34]. After clustering of the image in  $k$  clusters using, for instance,  $k$ -means, pixels are iteratively chosen from the cluster  $c_i$  with a probability proportional to the following heuristic

$$p(c_i) \propto \frac{\frac{n_i}{l_i + 1}}{\sum_{j=1}^k \frac{n_j}{l_j + 1}} \quad (5.2)$$

where  $n_i$  is the size of the cluster and  $l_i$  is the number of labeled pixels already present in the cluster. In this way we sample from large cluster with no labeled pixels, where we suppose the new classes to lie. This cluster-based strategy is summarized in Fig. 5.3. After an iteration of this procedure, traditional active learning can be used to refine the classification boundaries defined.

## 5.3. Data and Experimental Setup

This section presents the data set considered and details the setup of the experiments performed in Section 5.4.

### 5.3.1. Data Sets

Two urban data sets at metric spatial resolution have been considered.

The first data set is a 1.3 m resolution image of the city of Pavia (Italy), shown on the left side of Fig. 5.4. The image was taken by the airborne ROSIS-03 sensor [35]. The image is 1400×512 pixels and has a spectral resolution from 0.43 to 0.86  $\mu\text{m}$  divided into 102 spectral bands. The proposed approach has been tested on a 5-class problem, namely, Buildings, Roads, Water, Vegetation and Shadows. These classes of interest have been included in a labeled data set of 206009 samples extracted by visual inspection.

The second case study considers a 2.4 m resolution image of a suburb of the city of Zurich (Switzerland), shown on the right side of Fig. 5.4. The image has been acquired by the sensor on the Quick-



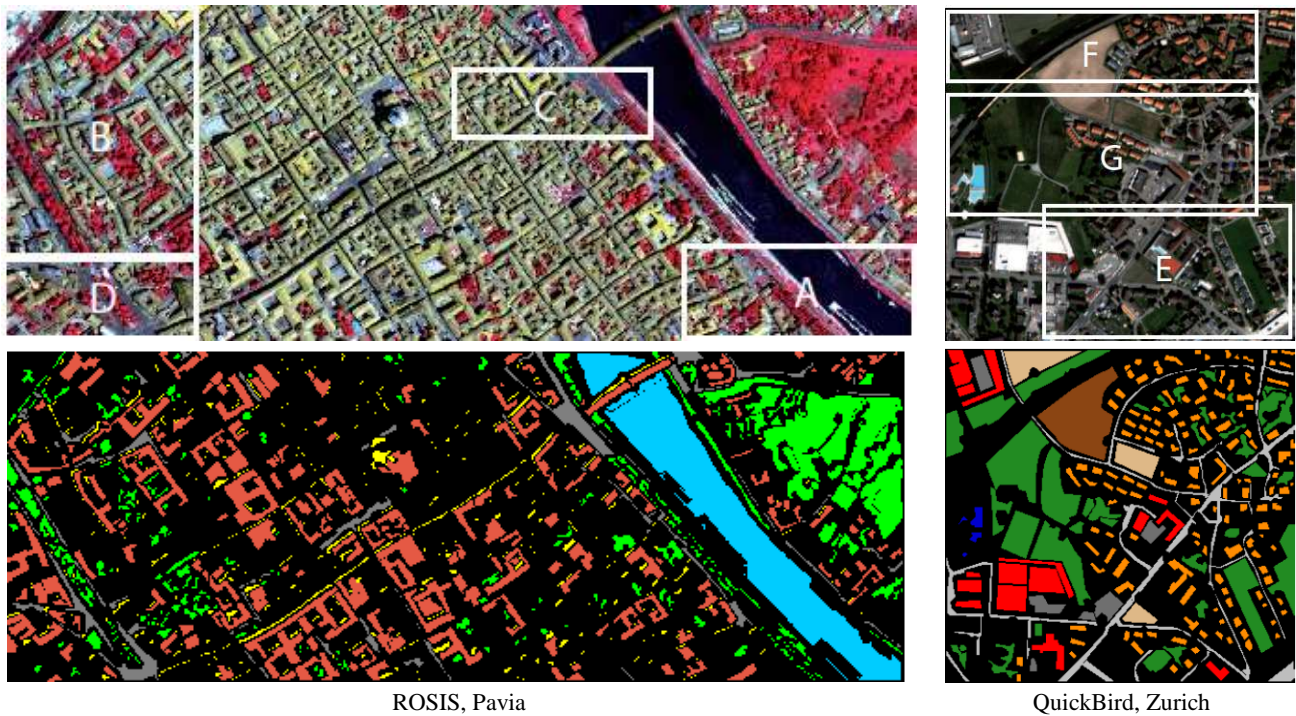


Fig. 5.4. Top row: considered urban datasets. Areas marked by ‘A’ and ‘B’ (respectively ‘E’ and ‘F’) are the training areas of the experiments shown in Section 5.4. ‘C’ and ‘D’ (respectively ‘G’) areas are only used for graphics of an unseen area. Bottom row: available ground truth pixels.

Bird satellite and is a  $329 \times 347$  pixel image with four spectral bands in the visible and near-infrared portions of the spectrum. A total of 43398 pixels have been labeled by visual inspection on the image with eight land use classes have been selected for analysis (Residential, Commercial, Vegetation, Soil, Mixed soil/vegetation, Roads, Pools, Parkings). Note that several classes have very similar spectral signatures and, in order to differentiate them, contextual filters using mathematical morphology [36] with perband opening and closing filters using spherical structure elements of 3 and 5 pixels diameter have been added to the data set. This increases the dimensionality of the data set from 4 to 20 features. These filters have been shown to have desirable properties when applied to urban VHR classification problems [37], [38].

In addition, two agricultural data sets at medium spatial resolution have been considered.

The third data set called Flightline C1 is a 12-bands multispectral image taken over Tippecanoe County, IN by the M7 scanner in June 1966 [8]. The image is  $949 \times 220$  pixels and contains 10 classes, mainly crop types. A ground survey of 70847 pixels has been used.

The fourth data set is the classical 220-bands AVIRIS image taken over Indiana's Indian Pine test site in June 1992. The image is  $145 \times 145$  pixels, contains 12 major crop types classes (with more than 100 labeled samples), and a total of 10172 labeled pixels. This image is a classical benchmark to validate model accuracy and constitutes a very challenging classification problem because of the strong mixture of the classes' signatures and unbalanced number of labeled pixels per class. Before training the classifiers, we removed 20 noisy bands covering the region of water absorption and reduced the dimensionality to 6 features with principal component analysis (accounting for 99.9% of data variance) to ensure correct estimation of the covariance matrix. As for the Zurich image, morphological opening and closing bands have been added to the extracted features. This is justified by the fact that the image has been taken shortly after plantation of the crops, thus showing class signatures that are, in fact, mixtures between soil and crops. Therefore, in order to achieve correct detection, contextual information must be added.

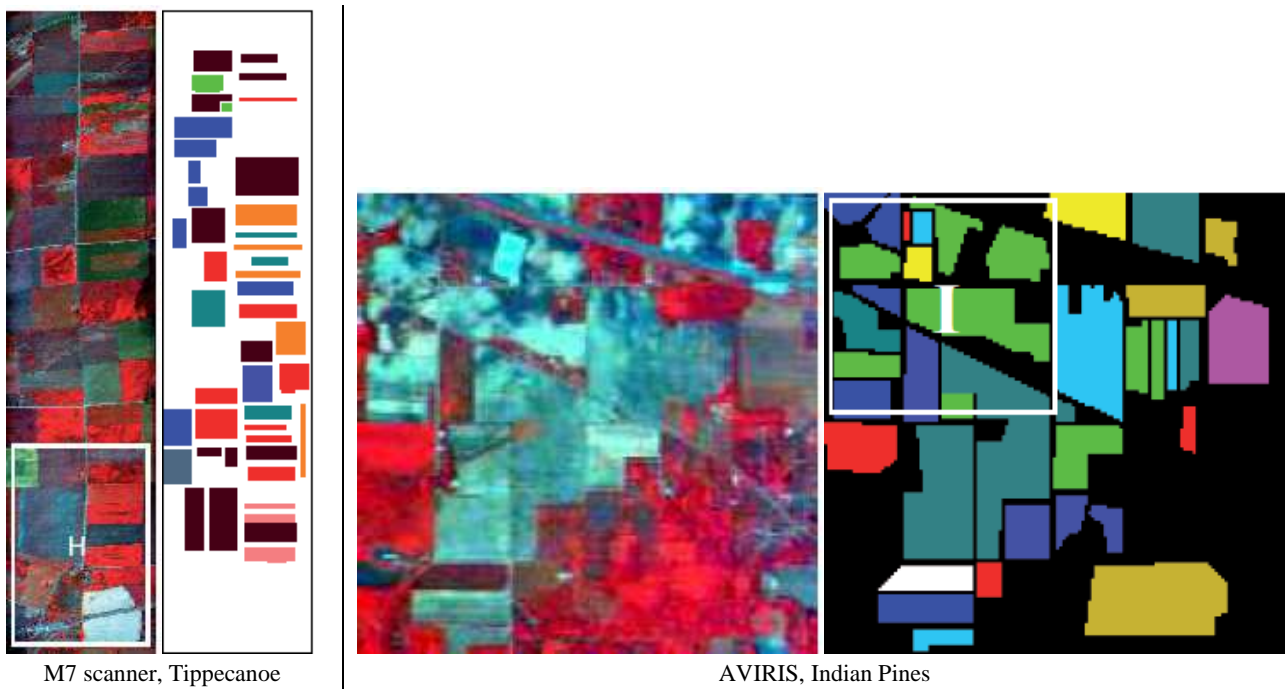


Fig. 5.5. Considered agricultural datasets and available ground truth pixels. Areas marked by 'H' and 'I' are the training areas of the experiments shown in Section 5.4.

### 5.3.2. Experimental Setup

Experiments on urban areas use four training areas, each providing areas with increasing complexity in landcover.

A. This area covers all the classes present in the Pavia image. The shifts that need to be detected by the learning process are related to sampling in incomplete portions of the distribution. This first step can be considered as a classical active learning problem.

B. This area of the Pavia image lacks the class Water. In this example, we aim at discovering a major class (water covers a large part of the rest of the image) for a relatively easy classification problem. This experiment should reveal an inadequacy of traditional active learning since random sampling has a higher probability of finding this new class simply by chance.

E. This area of the Zurich image accounts for most of the classes except Water which for this image is a very marginal class. The aim of this experiment is to assess whether the cluster-based strategy is adapted to find small classes.

F. This experiment is the most complex for urban areas. The 'F' area of the Zurich image lacks two classes (Water and Bare Soil), one being major and the other marginal. In this case we want to assess the ability of the proposed approach to update the model to one with several new classes having different PDFs in the new image.

Regarding agricultural areas, we concentrate on the problem of discovering new classes. Two experiments with increasing complexity have been performed.

H. In this setting, the model is trained with a ground truth covering a small part of the image with reduced ground truth. Both major and marginal classes are missing. In particular, a major class is not reported in the initial ground survey ('Oats,' in blue in Fig. 5.5), thus implying very poor performance of the model without samples from the new distribution.

I. This experiment is designed to test the algorithms proposed to discover classes with strongly overlapping spectra. As it has been mentioned above, the image was taken shortly after the crops were planted, so that each signature is not pure, rather a mixture between soil and crop, resulting in strongly

overlapping classes. In this setting, three classes are unknown to the first model, ‘Soybean-clean,’ ‘Wheat’ and ‘Grass/pasture-mowed.’ By the strong degree of mixture of the classes of this image with the unknown classes, this problem seems not to be suited for standard active learning algorithms.

For all experiments, 1) first the LDA classifier is optimized using 1000 pixels from the Pavia image (300 for the Zurich image, 300 for the Tippecanoe image, and 300 for the Indian Pines image) from the training sub-area and tested on the available ground truth in the same area. This experiment assesses the performance of the model for the subarea the training samples are drawn from. Afterwards, four experiments are added as follows: 2) direct classification of the entire image with the same training data; 3) classification of the entire image using 1600 (1000, 600, and 2300) pixels randomly selected from the whole image; 4) starting with the 1000 (300, 300, and 300) pixels of the model locally optimal, sample 600 (700, 300, and 2000) pixels randomly; and 5) with the same initial set, actively sample 600 (700, 300, and 2000) pixels. Finally, 6) active sampling of 600 (700, 300, and 2000) pixels is applied after the clustering-based initial selection.

For active learning, BT active learning is implemented in MATLAB. Thirty (70, 30, and 100) iterations with 20 (10, 10, and 20) samples per iteration have been carried out. The differences in number of pixels per iteration and in the number of iterations are dictated by the different resolutions of the images and by the differences in complexity between the data sets respectively. Ten independent runs have been conducted to study stability of the solution with respect to initialization. Performance was evaluated in terms of overall accuracy (OA), Kappa statistic and standard deviations.

## 5.4. Results and Discussion

This section presents and discusses the experimental results obtained by the proposed method on both the urban and the agricultural data sets.

### 5.4.1. Urban Data

The first rows of Table 5.I report the performance of the different strategies considered for the Pavia data set by considering the patch ‘A’ as initial training area. When trained solely on the patch ‘A,’ LDA performs perfectly when classifying that patch (OA=98.42%), but fails on the entire image, where a decrease of about 12% in accuracy is observed (to 87.23%). A classifier trained on 1600 pixels randomly selected from the entire image can improve this result by approximately 2% as does a random-based strategy sampling from the 1000 initial samples. On the contrary, selecting the new pixels with active learning leads to an increase in performance of about 5% relative to the base classifier and 3% with respect to the experiment using 1600 random pixels. This approach reaches the best accuracy observed at 93.03% and 0.906 in terms of Kappa statistic. This is because the sampling is focused on the boundaries between classes where the shifts among distributions are more likely to occur. The curves of Fig. 5.6(a) show performance of the proposed methods as a function of the number of training samples. We note that the active learning process is faster to converge than it is the random selection process. In particular, 40 additive samples are sufficient for the standard BT method to reach the value of accuracy obtained by adding 600 random samples to the initial training set. Comparing orange and green curves, which are related to active sampling with and without clustering-based initialization respectively, we observe that the clustering strategy is not useful for this particular scenario. In fact, all the classes are already included in the initial training set, and so the initialization step tends to select samples that are not really important for better discriminating the different classes. In any case, a good improvement with respect to the random selection is preserved.

The results of the second experiment, in which the patch ‘B’ has been used to select the initial training set, are presented in the second part of Table 5.I. Because water pixels are not present in this patch, results show a strong decrease of LDA performance when applied to the entire image (from 85.81% to 67.27%).

TABLE 5.1  
 OA AND KAPPA FOR THE PAVIA DATASET. ITERATIVE STRATEGIES ARE GIVEN AT CONVERGENCE.  
 IN BOLD, BEST RESULTS AMONG THE EXPERIMENTS FOR THE TRAINING AREA

Training patch	Prediction area	# train		Sampling strategy	OA		Kappa	
		Base	Added		$\mu$	$\sigma$	$\mu$	$\sigma$
A	A*	1000	-	-	98.42	0.12	0.965	0.003
	All image	1000	-	-	87.23	0.70	0.827	0.009
	All image	1600	-	-	89.81	0.25	0.864	0.003
	All image	1000	600	RS	89.31	0.26	0.857	0.003
	All image	1000	600	BT	<b>93.03</b>	<b>0.20</b>	<b>0.906</b>	<b>0.003</b>
	All image	1000	600	Cluster + BT	92.97	<b>0.17</b>	<b>0.905</b>	<b>0.002</b>
B	B*	1000	-	-	85.81	0.74	0.767	0.012
	All image	1000	-	-	67.27	0.30	0.572	0.007
	All image	1600	-	-	89.78	0.28	0.863	0.004
	All image	1000	600	RS	88.83	0.46	0.850	0.006
	All image	1000	600	BT	<b>91.98</b>	<b>0.25</b>	<b>0.892</b>	<b>0.003</b>
	All image	1000	600	Cluster + BT	91.89	0.20	0.891	0.003

\* Not comparable with the results of the other rows, different test sets.

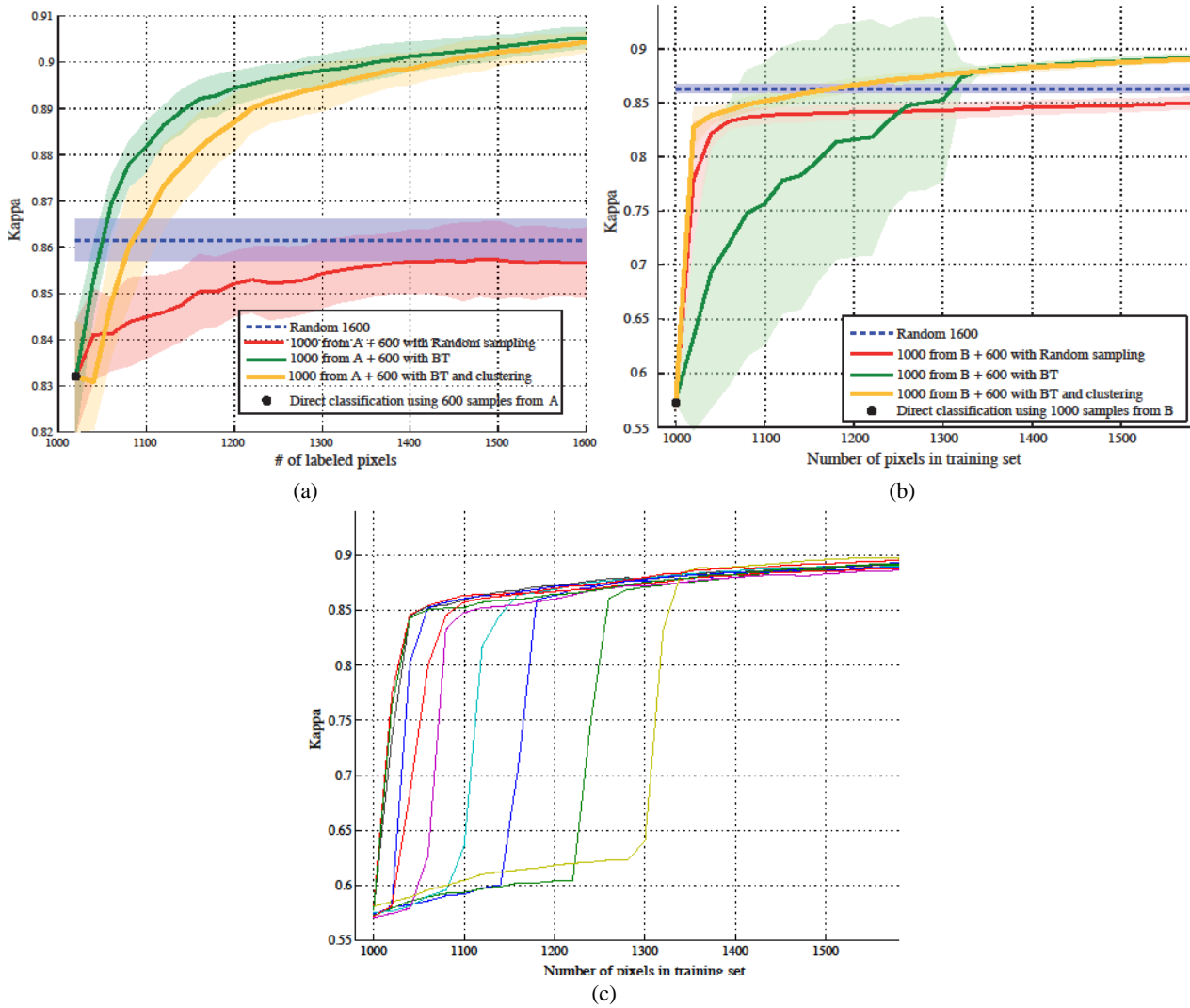


Fig. 5.6. Learning curves for the Pavia dataset. a) when using image patch 'A' for training set; b) when using image patch 'B' for training. c) Single runs composing the BT active learning curve (green curve in panel b).



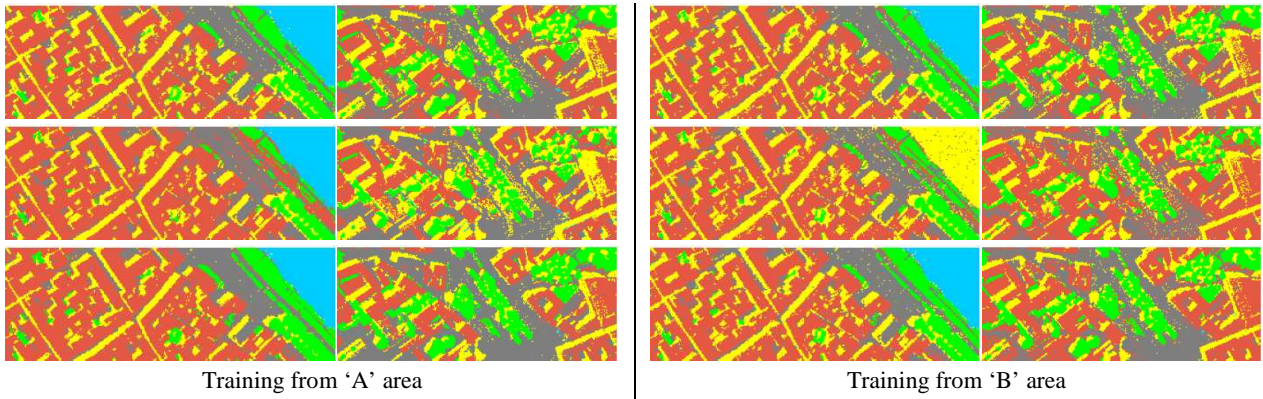


Fig. 5.7. Classification maps for the Pavia dataset of regions ‘C’ and ‘D’ obtained when training LDA using pixels from regions (left) ‘A’ and (right) ‘B.’ Top row illustrates the upper bound, where 1600 pixels randomly selected from the entire image. The middle row shows the experiment using the 1000 pixels only. The bottom row illustrates the results obtained by adding to these 1000 pixels 600 actively selected pixels from the rest of the image.

Sampling randomly from the entire image solves this problem, since the water class is well represented in the rest of the image and it is relatively easy to find by arbitrary sampling. Again, the active learning algorithm outperforms the others by 2%–3% by focusing on the uncertain areas, resulting in an accuracy of 91.98% and 0.892 in Kappa. Regarding the curves in Fig. 5.6(b), the active learning strategy is slower than the others to converge. The green curve in Fig. 5.6(b) is even worse than random selection in the first iterations. This can be explained by the plots of Fig. 5.1. If the water class is not found no area of uncertainty will be present for the class water and as a consequence such a class will never be sampled (unless by chance). The single runs generating the green curve in Fig. 5.6(b) are shown in Fig. 5.6(c). The steep increase in accuracy for each run corresponds to the iteration where the water class is discovered. Applying the active learning after the clustering-based initialization, we have a fast convergence to optimal results avoiding overfitting, as illustrated by the orange curve in Fig. 5.6(b). In this case, 180 additive samples are necessary to exceed the value of accuracy associated with the random selection.

These observations are confirmed by the maps shown in Fig. 5.7, in which a decrease of noisy classification patterns is obtained using the active learning strategy. Active strategies avoid sampling in already solved areas and thus reduce noisy classification results induced by sampling outliers.

Results obtained for the Zurich data set confirm the considerations given for the Pavia image. For both patch ‘E’ and ‘F’ as initial training areas, active learning outperforms by about 5% the random selection method as described in Table 5.II. Once again the plots in Fig. 5.8. highlight the necessity of performing the initial selection with the clustering based strategy when classes are missing in the initial training set. In particular, while this aspect is not crucial for the patch ‘E,’ in which a single marginal class is not present initially, it becomes fundamental for the patch ‘F,’ which lacks two classes, one major and the other marginal. Starting from the patch ‘E,’ both strategies need 100 additional samples to reach the random sampling accuracy. For patch ‘F’ only 80 instead of 220 samples are needed with clustering initialization relative to the traditional BT method. In the graph of Fig. 5.9, we report the number of iterations necessary to discover the classes missing in patch ‘F’ in the ten experiments performed. For the class Bare Soil, shown in Fig. 5.9(a), the initialization process is able to find it at the first iteration for all the ten runs considered. An identical behavior is obtained for the class Water [see Fig. 5.9(b)], although the number of pixels of this class is very limited. A high probability of detection is verified for the random selection in the Bare Soil case, given the fact that it is easy to find this class by chance, while poor performance is obtained for class Water. Finally, the traditional active sampling fails for both cases, where 30 iterations are needed to discover pixels of these classes in some runs. The final maps obtained for the Zurich image for the different proposed solutions are shown in Fig. 5.10.

TABLE 5.II  
 OA AND KAPPA FOR THE ZURICH DATASET. ITERATIVE STRATEGIES ARE GIVEN AT CONVERGENCE.  
 IN BOLD, BEST RESULTS AMONG THE EXPERIMENTS FOR THE TRAINING AREA

Training patch	Prediction area	# train		Sampling strategy	OA		Kappa	
		Base	Added		$\mu$	$\sigma$	$\mu$	$\sigma$
E	E*	300	-	-	92.25	0.521	0.902	0.006
	All image	300	-	-	68.62	2.60	0.614	0.029
	All image	1000	-	-	79.48	1.23	0.743	0.014
	All image	300	700	RS	80.19	1.19	0.751	0.014
	All image	300	700	BT	85.07	<b>0.58</b>	0.809	<b>0.007</b>
	All image	300	700	Cluster + BT	<b>85.35</b>	0.68	<b>0.813</b>	0.008
F	F*	300	-	-	83.62	1.24	0.785	0.016
	All image	300	-	-	67.54	1.03	0.596	0.012
	All image	1000	-	-	78.87	1.49	0.736	0.017
	All image	300	700	RS	80.08	1.24	0.750	0.014
	All image	300	700	BT	<b>85.25</b>	<b>0.67</b>	<b>0.812</b>	<b>0.008</b>
	All image	300	700	Cluster + BT	<b>85.25</b>	<b>0.67</b>	<b>0.812</b>	<b>0.008</b>

\* Not comparable with the results of the other rows, different test sets.

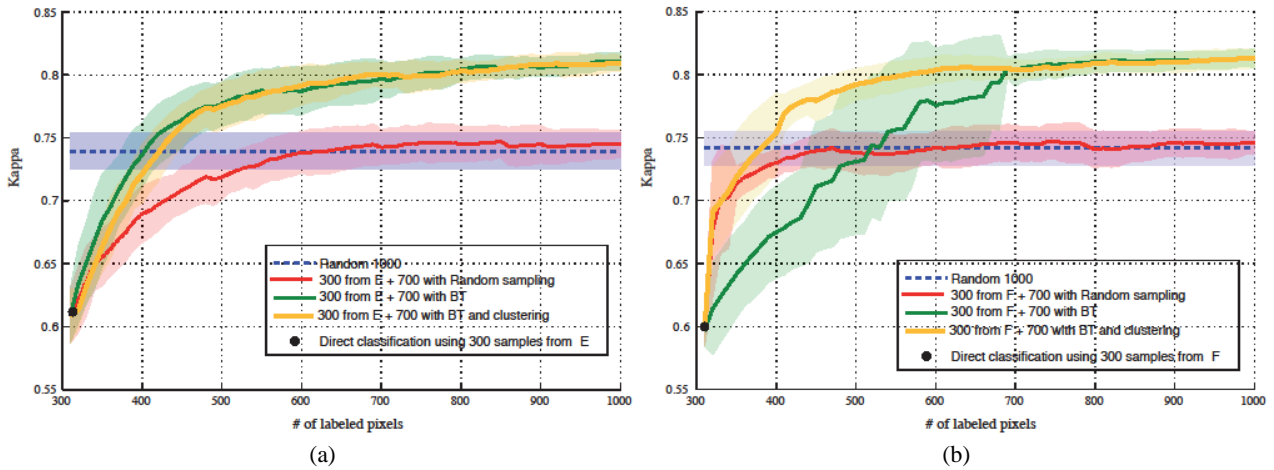


Fig. 5.8. Learning curves for the Zurich dataset. a) when using image patch ‘E’ for training set; b) when using image patch ‘F’ for training.

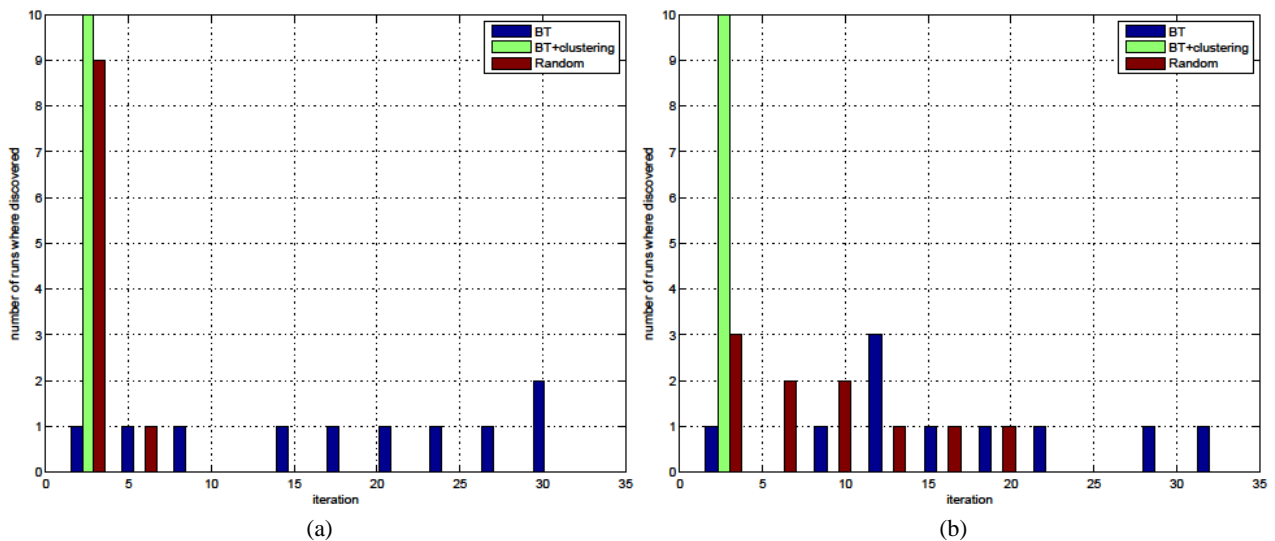


Fig. 5.9. Number of runs for the Zurich dataset where classes missing in the image patch ‘F’ are discovered at each iteration. a) for training on patch ‘E’; b) for training on patch ‘F’.



Fig. 5.10. Classification maps for the Zurich dataset of the region ‘G’ obtained when training LDA using pixels from regions (left) ‘E’ and (right) ‘F’. Top row illustrates the upper bound, classifying 1000 pixels randomly selected from the entire image. The middle row shows the experiment using the 300 pixels only. The bottom row illustrates the results obtained by adding to these 300 pixels 700 actively selected pixels from the rest of the image.

#### 5.4.2. Agricultural Data

Results obtained for the agricultural data sets are illustrated in Table 5.III and corresponding Figs. 5.11–5.13.

At convergence, the results for the Tippecanoe image (training patch ‘H’) show an improvement with respect to random sampling by approximately 2% and 0.02 in terms of accuracy and Kappa respectively, which is less spectacular than in the previous experiments. However, the learning rates show a strong divergence between the random and the active curves starting from iteration 3, when 360 samples are used for training (left side of Fig. 5.11). The similar behavior in the first two iterations is observed because the initial training set obviates most of the classes and then all the strategies perform well. Once the classification problem has become clearer, the active learning strategies can make difference, as shown in the figure. This behavior was already encountered and documented in [24]. As for the classification maps of Fig. 5.12, the active learning strategy returns a more desirable description of the class ‘Rye’ (in red), whose confusion with the class ‘Soil’ (in pink) is strongly diminished.

The last experiment considers the Indian Pines image. For this complex data set, consisting classes showing strongly mixed signatures, the same behavior as in the urban data set is observed (right side of Fig. 5.11): the traditional active learning strategy does not converge efficiently in the first iterations and is outperformed by random sampling. This again is due to the incapability of this strategy to discover new classes in highly overlapping problems. On the contrary, the proposed strategy considering pre-clustering performs efficiently, learns the global structure as efficiently as random sampling and outperforms it after 200 queries, reaching at convergence results higher by 3% in accuracy and 0.04 in Kappa. The classification



TABLE 5.III

OA AND KAPPA FOR THE (TOP) TIPPECANOE AND (BOTTOM) INDIAN PINES DATASETS. ITERATIVE STRATEGIES ARE GIVEN AT CONVERGENCE. IN BOLD, BEST RESULTS AMONG THE EXPERIMENTS FOR THE TRAINING AREA

Training patch	Prediction area	# train		Sampling strategy	OA		Kappa	
		Base	Added		$\mu$	$\sigma$	$\mu$	$\sigma$
H	H*	300	-	-	99.26	0.20	0.988	0.003
	All image	300	-	-	82.78	1.48	0.800	0.021
	All image	600	-	-	96.06	0.74	0.951	0.009
	All image	300	300	RS	96.04	0.53	0.951	0.006
	All image	300	300	BT	97.62	0.72	0.970	0.009
	All image	300	300	Cluster + BT	<b>97.79</b>	<b>0.33</b>	<b>0.972</b>	<b>0.004</b>
I	I*	300	-	-	72.52	2.21	0.671	0.026
	All image	300	-	-	43.70	0.80	0.365	0.009
	All image	2300	-	-	71.25	0.66	0.673	0.007
	All image	300	2000	RS	71.78	<b>0.44</b>	0.679	<b>0.005</b>
	All image	300	2000	BT	74.37	0.71	0.709	0.008
	All image	300	2000	Cluster + BT	<b>74.69</b>	1.07	<b>0.713</b>	0.012

\* Not comparable with the results of the other rows, different test sets.

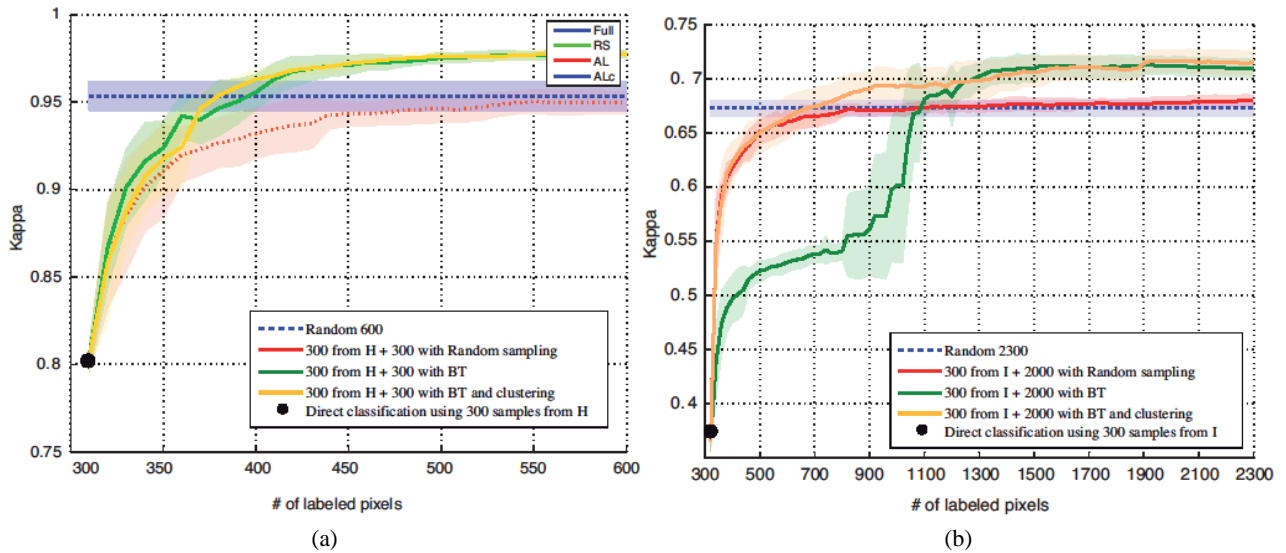


Fig. 5.11. Learning curves for the agricultural datasets. (a) Tippecanoe; (b) Indian Pines.

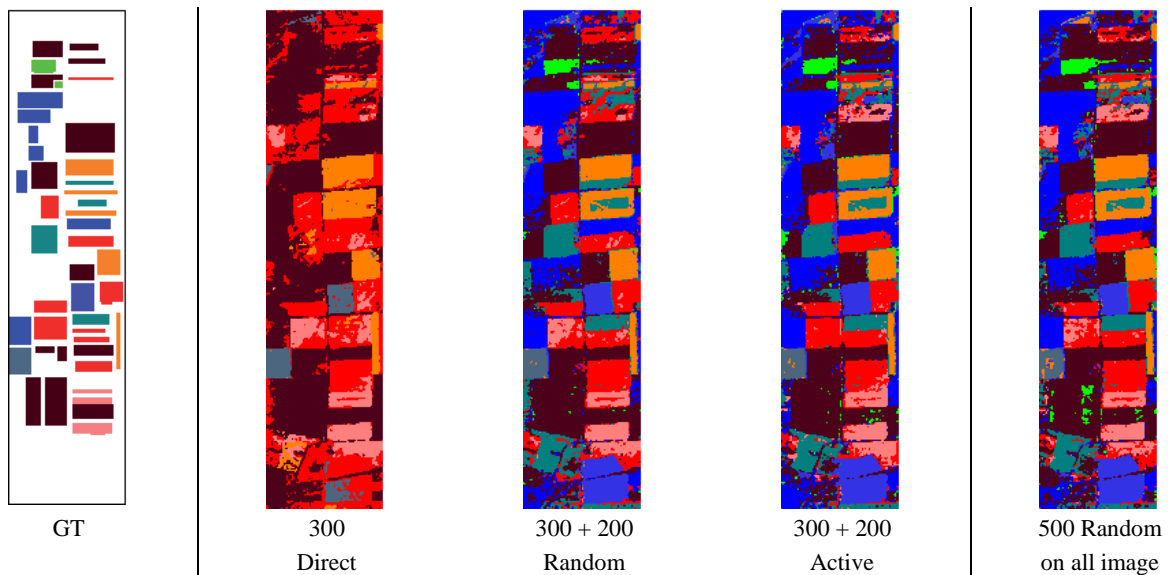


Fig. 5.12. Classification maps for the Tippecanoe dataset using training information coming from the 'H' area after 10 iterations.



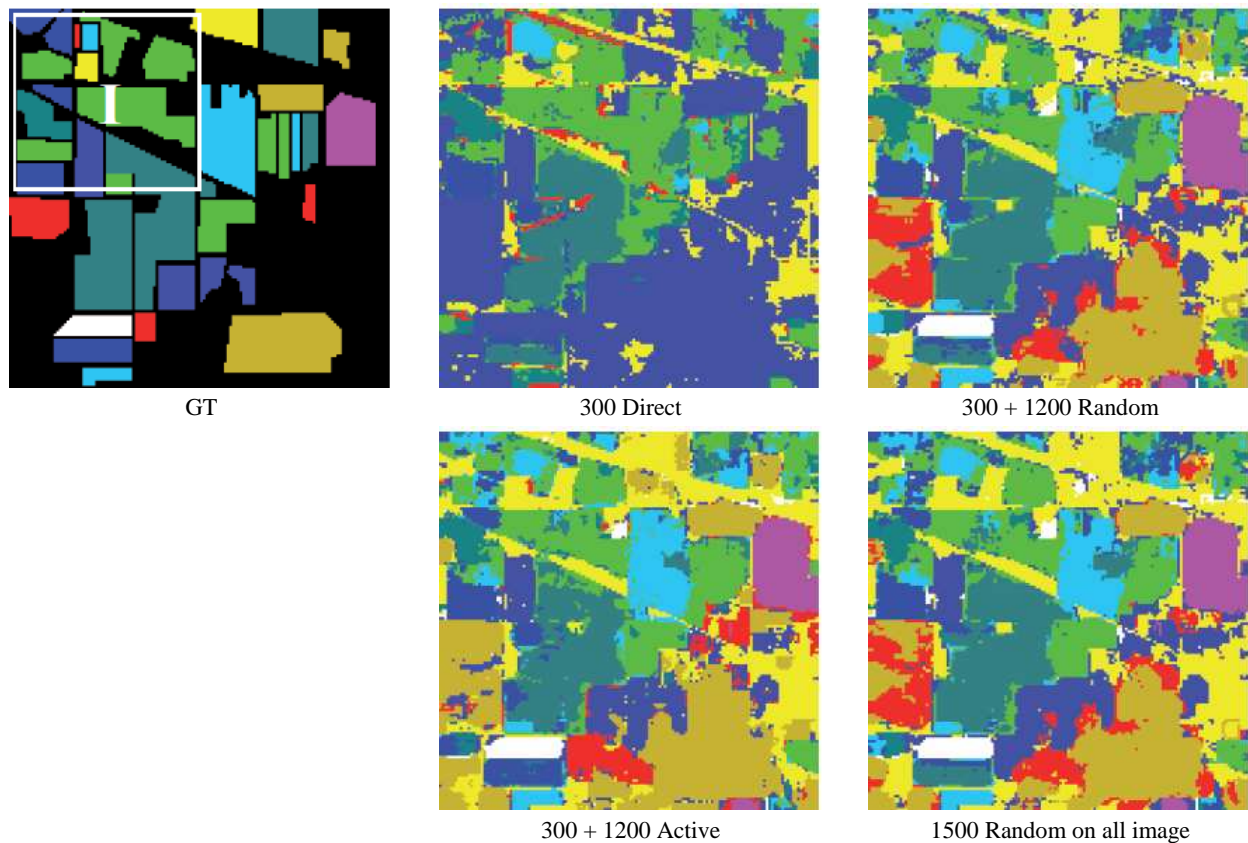


Fig. 5.13. Classification maps for the Indian Pines dataset using training information coming from the ‘I’ area after 60 iterations.

maps obtained by this strategy, illustrated in Fig. 5.13, show a more homogeneous result than the one obtained by random sampling.

## 5.5. Conclusion

In this chapter, we have proposed a simple, but effective way to use active learning to solve the problem of data set shift, which may occur when a classifier trained on a portion of the image is applied to the rest of the image. The experimental results obtained on hyperspectral and VHR data sets demonstrate good capability of the proposed method for selecting pixels that allow rapid convergence to an optimal solution. Moreover, the use of a clustering-based selection strategy allows us to discover new classes in case they have been omitted in the initial training set. Such strategies for optimal sampling guarantee signature extension and can be extended to a large variety of applications dealing with spectral data, as it is not dependent on the image characteristics of the data. Future research will explore these kinds of applications. An example could be the classification of electrocardiographic signals, that has recently been tackled in [39] using active learning techniques, but without considering issues related to covariate shift.

## 5.6. Acknowledgment

The authors would like to acknowledge Prof. Paolo Gamba from the University of Pavia for providing the ROSIS data, Prof. Mikhail Kanevski from the University of Lausanne for providing the QuickBird data and Prof. M. M. Crawford for providing the AVIRIS and M7 data.

## 5.7. References cited in Chapter 5

- [1] S. Rajan, J. Ghosh, and M. Crawford, “Exploiting class hierarchy for knowledge transfer in hyperspectral data,” *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 11, pp. 3408–3417, Nov. 2006.

- [2] M. D. Fleming, J. S. Berkebile, and R. M. Hoffer, "Computer-aided analysis of LANDSAT-I MSS data: A comparison of three approaches, including a "Modified clustering" approach", *LARS information note 072475 Purdue University*, 1975.
- [3] C. E. Woodcock, S. A. Macomber, M. P. Pax-Lenney, and W. B. Cohen, "Monitoring large areas for forest change using landsat: Generalization across space, time and landsat sensors," *Remote Sens. Environ.*, vol. 78, no. 1–2, pp. 194–203, Oct. 2001.
- [4] M. Pax-Lenney, C. E. Woodcock, S. A. Macomber, S. Gopal, and C. Song, "Forest mapping with a generalized classifier and landsat TM data," *Remote Sens. Environ.*, vol. 77, no. 3, pp. 241–250, Sep. 2001.
- [5] G. M. Foody, D. S. Boyd, and M. E. J. Cutler, "Predictive relations of tropical forest biomass from landsat TM data and their transferability between regions," *Remote Sens. Environ.*, vol. 85, no. 4, pp. 463–474, Jun. 2003.
- [6] I. Olthof, C. Butson, and R. Fraser, "Signature extension through space for northern landcover classification: A comparison of radiometric correction methods," *Remote Sens. Environ.*, vol. 95, no. 3, pp. 290–302, Apr. 2005.
- [7] X. Jia and J. A. Richards, "Cluster-space representation for hyperspectral data classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 40, no. 3, pp. 593–598, Mar. 2002.
- [8] Q. Jackson and D. Landgrebe, "An adaptive classifier design for high-dimensional data analysis with a limited training data set," *IEEE Trans. Geosci. Remote Sens.*, vol. 39, no. 12, pp. 2664–2679, Dec. 2001.
- [9] L. Gómez-Chova, G. Camps-Valls, J. Muñoz-Marí, and J. Calpe, "Semi-supervised image classification with laplacian support vector machines," *IEEE Geosci. Remote Sens. Lett.*, vol. 5, no. 3, pp. 336–340, Jul. 2008.
- [10] G. Camps-Valls and L. Bruzzone, *Kernel methods for remote sensing data analysis*. NJ, USA: J. Wiley & Sons, 2009.
- [11] D. Tuia and G. Camps-Valls, G, "Semi-supervised remote sensing image classification with cluster kernels," *IEEE Geosci. Remote Sens. Lett.*, vol. 6, no. 2, pp. 224–228, Apr. 2009.
- [12] J. Quiñero-Candela, M. Sugiyama, M., A. Schwaighofer, and N. D. Lawrence, *Dataset shift in machine learning*. MIT Press, 2009.
- [13] M. Sugiyama, M. Krauledat, and K. R. Müller, "Covariate shift adaptation by importance weighted cross validation," *J. Mach. Learn. Res.*, vol. 8, pp. 985–1005, May 2007.
- [14] S. Bickel, M., Brückner, and T. Scheffer, "Discriminative learning under covariate shift," *J. Mach. Learn. Res.*, vol. 10, pp. 2137–2155, Sep. 2009.
- [15] J. Yang, R. Yan, and A. G. Hauptmann, "Cross-domain video concept detection using adaptive SVMs", in *Proc. 15th International Conference on Multimedia*, Augsburg, Germany, 2007, pp. 188–197.
- [16] Y. Li, H. Kambara, Y. Koike, and M. Sugiyama, "Application of covariate shift adaptation techniques in brain computer interfaces," *IEEE Trans. Biomed. Eng.*, vol. 57, no. 6, pp. 1318–1324, Jun. 2010.
- [17] G. Schweikert, C. Widmer, B. Schölkopf, and G. Rätsch, "An empirical analysis of domain adaptation algorithms for genomic sequence analysis," in *Proc. Advances in Neural Information Processing Systems*, Vancouver, Canada, 2008.
- [18] L. Bruzzone and D. Fernandez-Prieto, "Unsupervised retraining of a maximum likelihood classifier for the analysis of multitemporal remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 39, no. 2, pp. 456–460, Feb. 2001.
- [19] L. Bruzzone and M. Marconcini, "Toward the automatic updating of land-cover maps by a domain-adaptation SVM classifier and a circular validation strategy," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 4, pp. 1108–1122, Apr. 2009.
- [20] L. Gómez-Chova, G. Camps-Valls, L. Bruzzone, and J. Calpe-Maravilla, "Mean map kernel methods for semisupervised cloud classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 1, pp. 207–220, Jan. 2010.
- [21] P. Mitra, B. Uma Shankar, and S. Pal, "Segmentation of multispectral remote sensing images using active support vector machines," *Pattern Recogn. Lett.*, vol. 25, no. 9, pp. 1067–1074, Jul. 2004.
- [22] Q. Liu, X. Liao, and L. Carin, "Detection of unexploded ordnance via efficient semisupervised and active learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 9, pp. 2558–2567, Sep. 2008.
- [23] S. Rajan, J. Ghosh, and M. Crawford, "An active learning approach to hyperspectral data classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 4, pp. 1231–1242, Apr. 2008.
- [24] D. Tuia, F. Ratle, F. Pacifici, M. Kanevski, and W. J. Emery, "Active learning methods for remote sensing image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 7, pp. 2218–2232, Jul. 2009.

- [25] J. Li, J. Bioucas-Dias, and A. Plaza, "Semisupervised hyperspectral image segmentation using multinomial logistic regression with active learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 11, pp. 4085–4098, Nov. 2010.
- [26] E. Pasolli, F. Melgani, and Y. Bazi, "Support vector machine active learning through significance space construction," *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 3, pp. 431–435, May 2011.
- [27] S. Patra and L. Bruzzone, "A fast cluster-assumption based active-learning technique for classification of remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 5, pp. 1617–1626, May 2011.
- [28] T. Luo, K. Kramer, D. B. Goldgof, L. O. Hall, S. Samson, A. Remsen, et al., "Active learning to recognize multiple types of plankton," *J. Mach. Learn. Res.*, vol. 6, pp. 589–613, Apr. 2005.
- [29] D. Cohn, L. Atlas, and R. Ladner, "Improving generalization with active learning," *Mach. Learn.*, vol. 15, no. 2, pp. 201–221, 1994.
- [30] L. Copa, D. Tuia, M. Volpi, and M. Kanevski, "Unbiased query-by-bagging active learning for VHR image classification," in *Proc. SPIE Remote Sensing Conference*, Toulouse, France, 2010.
- [31] M. Ferecatu and N. Boujemaa, "Interactive remote sensing image retrieval using active relevance feedback," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 4, pp. 818–826, Apr. 2007.
- [32] Z. Xu, K. Yu, V. Tresp, X. Xu, and J. Wang, "Representative sampling for text classification using support vector machines," in *Proc. 25th European Conf. on Information Retrieval Research*, 2003, pp. 393–407.
- [33] H. T. Nguyen and A. Smeulders, "Active learning using pre-clustering," in *Proc. International Conference on Machine Learning*, vol. 69, Banff, Canada, 2004, pp. 623–630.
- [34] S. Dasgupta and D. Hsu, "Hierarchical sampling for active learning," in *Proc. International Conference on Machine Learning*, vol. 307, Helsinki, Finland, 2008, pp. 208–215.
- [35] G. Licciardi, F. Pacifici, D. Tuia, S. Prasad, T. West, F. Giacco, et al., "Decision fusion for the classification of hyperspectral data: Outcome of the 2008 GRS-S data fusion contest," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 11, pp. 3857–3865, Nov. 2009.
- [36] P. Soille, *Morphological image analysis*. Berlin-Heidelberg: Springer-Verlag, 2004.
- [37] M. Fauvel, J. A. Benediktsson, J. Chanussot, and J. R. Sveinsson, "Spectral and spatial classification of hyperspectral data using SVMs and morphological profiles," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 11, pp. 3804–3814, Nov. 2008.
- [38] D. Tuia, F. Pacifici, M. Kanevski, and W. J. Emery, "Classification of very high spatial resolution imagery using mathematical morphology and support vector machines," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 11, pp. 3866–3879, Nov. 2009.
- [39] E. Pasolli and F. Melgani, "Active learning methods for electrocardiographic signal classification," *IEEE Trans. Inf. Technol. Biomed.*, vol. 14, no. 6, pp. 1405–1416, Nov. 2010.



## 6. Active Learning Methods for Biophysical Parameter Estimation

*Abstract* – In this chapter, we face the problem of collecting training samples for regression problems under an active learning perspective. In particular, we propose various active learning strategies specifically developed for regression approaches based on Gaussian processes (GP) and support vector machines (SVMs). For GP regression, the first two strategies are based on the idea of adding samples that are distant from the current training samples in the kernel space, while the third one uses a pool of regressors in order to select the samples with the greater disagreements between the different regressors. Finally, the last strategy exploits an intrinsic GP regression outcome to pick up the most difficult and hence interesting samples to label. For SVM regression, the method based on the pool of regressors and two additional strategies based on the selection of the samples distant from the current support vectors are proposed. The experimental results obtained on real data sets show that the proposed strategies exhibit a good capability to select samples that are significant for the regression process, thus opening the way to the active learning approach for remote sensing regression problems.

The work presented in this chapter has been submitted to *IEEE Trans. Geosci. Remote Sens.*; Co-authors: F. Melgani, N. Alajlan, Y. Bazi.

## 6.1. Introduction

Among the most challenging problems faced by the remote sensing community, one can find the estimation of biophysical parameters from remote sensing data. This issue spans different application domains such as estimation of biomass concentration in forest areas [1], assessment of ozone concentration in the atmosphere [2], and analysis of water quality for monitoring oceans and coastal areas through estimation of chlorophyll concentration [3].

From a methodological point of view, this problem can be viewed as an inverse modelling issue in which it is necessary to define a model that relates the acquired observations to the parameter of interest. The estimation of the model can be done by adopting supervised regression techniques, which require the availability of a set of training samples. By training samples, we mean pairs of radiances acquired by the sensor and measurements of the biophysical parameter to estimate. In the literature, two main approaches of regression have been proposed. The first one is based on parametric models (e.g., polynomial and exponential models), in which it is necessary to estimate the values of a predefined set of parameters. The second one makes use of nonparametric models, which depend completely from data. In general, because of the strong nonlinearity between the acquired radiances and the biophysical parameters to estimate, nonparametric methods have been preferred to parametric ones despite their greater computational complexity [4]. In particular, different approaches have been proposed, such as artificial neural networks (ANNs) [2], [5], [6], support vector machines (SVMs) [3], [7]-[9], and Gaussian processes (GPs) [10].

In the aforementioned works, the regression process is done by assuming that the training set is composed of a sufficient number of samples in order to obtain reliable and accurate estimations. However, from a practical point of view, the process of collecting training samples is not trivial, because the parameter measurements associated with the acquired radiances have to be performed manually by human experts and thus are subject to errors and costs in terms of time and money. For this reason, the number of available training samples is typically limited and performances can be consequently affected due to data scarcity. A solution to this problem is given by semisupervised approaches, in which the unlabeled samples are exploited during the design of the regression model in order to compensate the deficit in labeled samples. By unlabeled samples, we mean samples whose radiance values are known, but for which the corresponding biophysical parameter values are unknown. Such samples exhibit the advantage that they are available at zero cost from the data under analysis. In the literature, few works have been proposed for regression problems in general [11] and in the remote sensing field too [12], [13].

In the data classification context, a solution to the problem of training sample collection is given by the active learning approach. Starting from a small training set, additional samples are selected from a large amount of unlabeled data. These samples are labeled manually and added to the training set. The process is iterated until a stopping criterion is reached. Active learning strategies have been applied successfully in different fields [14], [15] and for remote sensing problems too, such as segmentation [16], detection of buried objects [17], [18], classification of hyperspectral images [19], [20], and classification of very high spatial resolution images [21], [22]. Similarly, the active learning approach has been studied for regression problems by the machine learning and statistics communities, in which it is also known as *optimal experimental design*. After the seminal paper by Cohn et al. [23], in which active learning has been applied to two statistically-based learning architectures, such as mixtures of Gaussians and locally weighted regression, several works have appeared in the last few years. For instance, in [24], the authors focus on the problem of local minima in active learning for neural networks, and two probabilistic solutions are proposed. In [25], after introducing the fundamental limits in a minimax sense of active and passive learning for various function classes, some strategies based on a tree-structured partition of the data are presented. In [26], considering linear regression scenarios, a method using the weighted least-squares learning based on the conditional expectation of the generalization error is proposed. In [27], the authors apply the query by

committee approach in the regression context. The main idea is to train a committee of learners and query the labels of the samples where the committee's prediction differ, thus minimizing the variance of the learner by training on samples where variance is largest. In [28], it is suggested to solve the problems of active learning and model selection at the same time in order to improve further the generalization performance. In [29], a solution to the problem of pool-based active learning in linear regression is proposed. In [30], the authors develop a strategy for kernel-based linear regression, in which the proposed greedy algorithm employs a minimum-entropy criterion derived using a Bayesian interpretation of ridge regression. Despite the promising performance given by the active learning approach in the regression field, nothing similar has been proposed in the remote sensing literature.

The objective of this chapter is to introduce the active learning approach for regression problems for the estimation of biophysical parameters from remote sensing data. In particular, we propose different active learning strategies specifically developed for two state-of-the-art regression approaches, namely GPs [31] and SVMs [32]. For GP regression, the first two methods are based on adding samples that are distant from the current training samples in the kernel space, while the third one uses a pool of regressors in order to select the samples with the greater disagreements between the different regressors. Finally, the last strategy exploits an intrinsic GP regression outcome to pick up the most difficult samples, and thus the most interesting ones. For SVM regression, the method based on the pool of regressors and two additional strategies based on the selection of the samples distant from the current support vectors are proposed. In order to assess the proposed strategies, we conducted an experimental study based on simulated and real data sets. The obtained experimental results show that interesting performances can be achieved.

The remaining part of the chapter is organized as follows. In Section 6.2, the basic mathematical formulation of GPs and SVMs are recalled. In Section 6.3, the active learning strategies proposed for regression problems are described. Section 6.4 presents the data sets used in the experimental analysis and the related results. Finally, conclusions are drawn in Section 6.5.

## 6.2. Gaussian Process and Support Vector Machine Regression

### 6.2.1. Gaussian Process Regression

Let us consider a set of labeled samples  $L = \{\mathbf{x}_i, y_i\}_{i=1}^n$ , where  $\mathbf{x}_i = [x_{i1}, \dots, x_{id}] \in \mathfrak{R}^d$  represents a vector of  $d$  remote observations and/or processed features and  $y_i \in \mathfrak{R}$  is the associated target value, i.e., the *in situ* measurement of the biophysical parameter of interest. Let us aggregate all  $\mathbf{x}_i$ 's ( $i=1, \dots, n$ ) into a feature matrix  $X$  and all  $y_i$ 's ( $i=1, \dots, n$ ) into a target vector  $\mathbf{y}$  so that  $L = \{X, \mathbf{y}\}$ . The goal is to infer from the set of labeled samples  $L$  the function  $f(\cdot)$  so that  $y = f(\mathbf{x})$ .

The underlying idea of the GP regression can be described in different ways. One of them consists in formulating the Bayesian estimation problem directly in the function space (the so-called function-space view). To understand such a formulation, let us first assume that the observed values  $\mathbf{y}$  of the function to model are the sum of a latent function  $\mathbf{f}$  and a noise component  $\boldsymbol{\varepsilon}$ , where

$$\mathbf{f} \sim GP(\mathbf{0}, K(X, X)) \quad (6.1)$$

$$\boldsymbol{\varepsilon} \sim N(\mathbf{0}, \sigma_n^2 I) \quad (6.2)$$

where  $GP(\cdot)$ , ' $\sim$ ',  $N(\cdot)$  and  $I$  stand for Gaussian Process, "follows", normal distribution and identity matrix, respectively. Equation (6.1) means that a GP is assumed over the latent function  $\mathbf{f}$ , i.e., this last is a collection of random variables, any finite number of which follow a joint Gaussian distribution with mean  $\mathbf{0}$  and covariance matrix  $K(X, X)$ . This matrix is built by means of a covariance (kernel) function  $k(\mathbf{x}, \mathbf{x}')$  computed on all the training sample pairs. Equation (6.2) states that a Gaussian distribution with mean  $\mathbf{0}$  and

variance  $\sigma_n^2$  is supposed for the entries of the noise vector  $\boldsymbol{\varepsilon}$  with each entry drawn independently from the others. Accordingly, we have

$$p(\mathbf{f} | X) = N(\mathbf{0}, K(X, X)) \quad (6.3)$$

$$p(\boldsymbol{\varepsilon}) = N(\mathbf{0}, \sigma_n^2 I) \quad (6.4)$$

where  $p(\cdot)$  stands for a probability density function.

Because of the statistical independence between the latent function  $\mathbf{f}$  and the noise component  $\boldsymbol{\varepsilon}$ , we can write

$$p_{\mathbf{y}}(z) = p_{\mathbf{f}}(z) * p_{\boldsymbol{\varepsilon}}(z) \quad (6.5)$$

where ‘\*’ is the convolution operator and consequently

$$p(\mathbf{y} | X) = N(\mathbf{0}, K(X, X) + \sigma_n^2 I). \quad (6.6)$$

Equation (6.6) means that the noisy observations  $\mathbf{y}$  are also modeled with a GP.

Now, let us focus the attention on the inference process. The best statistical estimation of the output value  $f_*$  associated with an unknown sample  $\mathbf{x}_*$  and given the set of labeled samples  $L$  is

$$\hat{f}_* | X, \mathbf{y}, \mathbf{x}_* \sim E\{f_* | X, \mathbf{y}, \mathbf{x}_*\} = \int f_* p(f_* | X, \mathbf{y}, \mathbf{x}_*) df_*. \quad (6.7)$$

It is clear that, for finding the output value estimate, we need the knowledge of the predictive distribution  $p(f_* | X, \mathbf{y}, \mathbf{x}_*)$ . For this purpose, we will first consider the joint distribution of the known observations  $\mathbf{y}$  and the desired function value  $f_*$ . Owing to the marginalization property of GPs, we can write the following expression:

$$p(\mathbf{y}, f_* | X, \mathbf{x}_*) = N\left(\mathbf{0}, \begin{bmatrix} K(X, X) + \sigma_n^2 I & \mathbf{k}_* \\ \mathbf{k}_*^T & k(\mathbf{x}_*, \mathbf{x}_*) \end{bmatrix}\right). \quad (6.8)$$

The vector  $\mathbf{k}_*$  denotes the covariance values between the training samples  $X$  and the sample  $\mathbf{x}_*$  whose prediction is desired. Since the joint distribution of  $\mathbf{y}$  and  $f_*$  is Gaussian, it can be shown that the conditional (or predictive) distribution is also Gaussian and takes the following expression

$$p(f_* | X, \mathbf{y}, \mathbf{x}_*) = N(\mu_*, \sigma_*^2) \quad (6.9)$$

where

$$\mu_* = \mathbf{k}_*^T [K(X, X) + \sigma_n^2 I]^{-1} \mathbf{y} \quad (6.10)$$

$$\sigma_*^2 = k(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{k}_*^T [K(X, X) + \sigma_n^2 I]^{-1} \mathbf{k}_*. \quad (6.11)$$

These are the key equations in the GP regression method. Two important information can be retrieved from them: 1) the mean  $\mu_*$ , which represents the best output-value estimate for the considered sample according to (6.7) and depends on the covariance matrix  $K(X, X)$ , the kernel distances between training and test samples  $\mathbf{k}_*$ , the noise variance  $\sigma_n^2$ , and the training observations  $\mathbf{y}$ ; and 2) the variance  $\sigma_*^2$ , which expresses a confidence measure associated by the model to the output estimate.

A central role in the GP regression model is played by the covariance function  $k(\mathbf{x}, \mathbf{x}')$  as it embeds the geometrical structure of the training samples. Through it, it is possible to define our prior knowledge about the smoothness of the output function  $f(\cdot)$ . A typical choice for the covariance function is the squared exponential function:

$$k_{SE}(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp\left(-\frac{|\mathbf{x} - \mathbf{x}'|^2}{2l^2}\right) \mathbf{k}_*. \quad (6.12)$$



The two hyperparameters  $\sigma_f^2$  and  $l$  are called process (signal) variance and length scale, respectively.

The tuning of the hyperparameters, better known as model selection issue, is a critical problem since it has a direct impact on the prediction accuracy. We can deal with this issue in different ways. In this work, we adopt the Bayesian model selection, which formulates the model selection issue within a Bayesian framework. It relies on the idea to maximize the posterior probability distribution defined over the vector of parameters  $\boldsymbol{\theta} = [\sigma_n^2, \sigma_f^2, l]$

$$p(\boldsymbol{\theta} | X, \mathbf{y}) = \frac{p(\mathbf{y} | X, \boldsymbol{\theta}) \cdot p(\boldsymbol{\theta})}{p(\mathbf{y} | X)}. \quad (6.13)$$

Often, the evaluation of the denominator in (6.13) is analytically intractable. As a solution, one may resort to the maximum-likelihood estimation procedure. It consists in the maximization of the marginal likelihood (evidence), i.e., the integral of the likelihood times the prior

$$p(\mathbf{y} | X, \boldsymbol{\theta}) = \int p(\mathbf{y} | \mathbf{f}, X, \boldsymbol{\theta}) \cdot p(\mathbf{f} | X, \boldsymbol{\theta}) d\mathbf{f} \quad (6.14)$$

with the marginalization over the latent function  $f$ . Under a GP regression modeling, both the prior and the likelihood follow Gaussian distributions. After some calculations, it is possible to show that the log marginal likelihood can be written as

$$\begin{aligned} \log p(\mathbf{y} | X, \boldsymbol{\theta}) = & -\frac{1}{2} \mathbf{y}^T \cdot (K(X, X) + \sigma_n^2 I)^{-1} \mathbf{y} + \\ & -\frac{1}{2} \log |K(X, X) + \sigma_n^2 I| + \\ & -\frac{n}{2} \log(2\pi) \end{aligned} \quad (6.15)$$

Equation (6.15) is the sum of three terms. The first is the only that involves the target observations and represents the capability of the model to fit the data. The second one is the model complexity penalty, while the third term is a normalization constant. Maximization of the marginal likelihood leads automatically to the best trade-off between model complexity and data fit. Moreover, no validation procedure with independent samples is needed. From an implementation viewpoint, this maximization problem can easily be solved by a gradient-based search routine. For more details, we refer the reader to [31].

### 6.2.2. Support Vector Machine Regression

The  $\varepsilon$ -insensitive SVM regression technique is based on the idea to find an estimate  $\hat{f}(\mathbf{x})$  of the true and unknown relationship  $y = f(\mathbf{x})$  between the vector of observations  $\mathbf{x}$  and the desired biophysical parameter  $y$  from the given set of training samples  $L$  such that: 1)  $\hat{f}(\mathbf{x})$  has, at most,  $\varepsilon$  deviation from the desired targets  $y_i$  ( $i=1, \dots, n$ ) and 2) it is as smooth as possible. This is usually performed by mapping the data from the original  $d$ -dimensional feature space to a higher dimensional transformed feature space, i.e.,  $\Phi(\mathbf{x}) \in \mathfrak{R}^{d'}$  ( $d' > d$ ), to increase the linearity of the function and, accordingly, to approximate it by the following linear model

$$\hat{f}(\mathbf{x}) = \boldsymbol{\omega}^* \cdot \Phi(\mathbf{x}) + b. \quad (6.16)$$

The optimal linear function in the higher dimensional transformed feature space is the one that minimizes a cost function, which expresses a combination of two criteria: Euclidean norm minimization and error minimization. The cost function is defined as

$$\Psi(\boldsymbol{\omega}, \boldsymbol{\xi}) = \frac{1}{2} \|\boldsymbol{\omega}\|^2 + C \sum_{i=1}^n (\xi_i + \xi_i^*) \quad (6.17)$$

subject to the following constraints

$$\begin{cases} y_i - (\mathbf{w} \cdot \Phi(\mathbf{x}_i) + b) \leq \varepsilon + \xi_i \\ (\mathbf{w} \cdot \Phi(\mathbf{x}_i) + b) - y_i \leq \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \geq 0 \end{cases}, \quad i=1,2,\dots,n \quad (6.18)$$

where  $\xi_i$ 's and  $\xi_i^*$ 's are the slack variables that are introduced to account for samples that do not lie in the  $\varepsilon$ -deviation tube. Constant  $C$  represents a regularization parameter that allows tuning the trade-off between the complexity of the function  $\hat{f}(\mathbf{x})$  and the tolerance of deviations larger than  $\varepsilon$ . The formulation of the error function is equivalent to dealing with the  $\varepsilon$ -insensitive loss function  $|\xi|_\varepsilon$  typically defined as

$$|\xi|_\varepsilon = \begin{cases} 0 & \text{if } |\delta| \leq \varepsilon \\ |\delta| - \varepsilon & \text{otherwise} \end{cases} \quad (6.19)$$

where  $\delta$  represents the deviation with respect to the desired target. This means that the differences between the targets and the estimated values are tolerated inside the  $\varepsilon$ -tube (error smallest than  $\varepsilon$ ), while a linear penalty is assigned to estimates lying outside the  $\varepsilon$ -insensitive tube.

The reformulation of the aforementioned optimization problem through a Lagrange functional into a dual optimization problem leads to a solution that is a function of the data conveniently expressed in the original dimensional feature space as

$$\hat{f}(\mathbf{x}) = \sum_{i \in S} (\alpha_i - \alpha_i^*) k(\mathbf{x}_i, \mathbf{x}) + b \quad (6.20)$$

where  $k(\cdot, \cdot)$  is a kernel function defined as

$$k(\mathbf{x}_i, \mathbf{x}) = \langle \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}) \rangle \quad (6.21)$$

and  $S$  is the subset of indices ( $i=1,2,\dots,n$ ) corresponding to the nonzero Lagrange multipliers  $\alpha_i$ 's or  $\alpha_i^*$ 's. The Lagrange multipliers weight each training sample according to its importance in determining a solution. The training samples associated to nonzero weights are called support vectors (SVs). In  $S$ , margin support vectors that lie on the  $\varepsilon$ -insensitive tube and nonmargin support vectors that correspond to errors coexist. The kernel  $k(\cdot, \cdot)$  should be chosen such that it satisfies the condition imposed by the Mercer's theorem. A common example of nonlinear kernel that fulfils Mercer's condition is the Gaussian kernel function.

### 6.3. Proposed Active Learning Methods

Let us consider a training set composed initially of  $n$  labeled samples  $L = \{\mathbf{x}_i, y_i\}_{i=1}^n$  and an additional learning set composed of  $m$  unlabeled samples  $U = \{\mathbf{x}_j\}_{j=n+1}^{n+m}$ , with  $m \gg n$ . In order to increase the training set  $L$  with a series of samples chosen from the learning set  $U$  and labeled manually by the expert, an active learning algorithm has the task of choosing them properly so as to minimize the error of the regression process while minimizing the number of learning samples to label.

In Fig. 6.1, we show the generic flow chart of the active learning approach for regression problems proposed in this chapter. Starting from the initial and small training set  $L$ , the unlabeled samples of the learning set  $U$  are evaluated and sorted using an opportune criterion  $h$ . In particular, we suppose for convention that the criterion  $h$  has to be minimized. At this point, from the sorted samples  $U_s$ , the first  $N_s$  samples are selected, where  $N_s$  is the number of samples to be added in the training set  $L$ . Finally, the selected samples  $U'_s$  are labeled by the human expert and added to the training set  $L$ . The entire process is iterated until the predefined convergence condition is satisfied (e.g., the total number of samples to add to the training set is reached, or the accuracy improvement on an independent calibration/validation set over the last iterations becomes insignificant).

Algorithm 6.1. summarizes the active learning approach for regression problems.

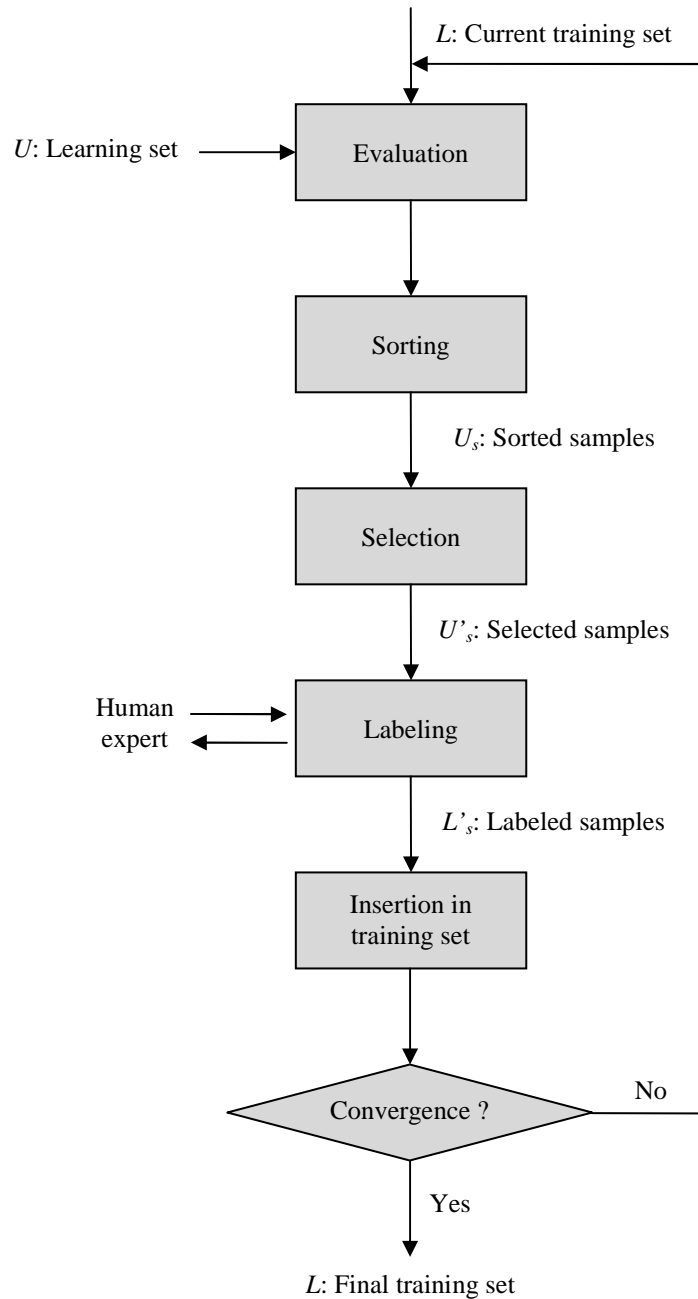


Fig. 6.1. Flow chart of the proposed active learning approach for regression problems.

---

**Algorithm 6.1.:** Active Learning Approach

---

**Inputs:**

$L$ : initial training set, composed of  $n$  labeled samples.

$U$ : learning set, composed of  $m$  ( $m \gg n$ ) unlabeled samples.

$N_s$ : number of samples to add at each iteration of the active learning process.

---

**Output:**

$L$ : final training set.

---

**Repeat**

1. Considering the current training set  $L$ , evaluate each sample  $\mathbf{x}_j$  ( $j = n+1, n+2, \dots, n+m$ ) of the learning set  $U$  using the criterion  $h$ .
  2. Sort the learning set  $U$  in function of the criterion  $h$  in order to obtain the set  $U_s$ .
  3. Select the first  $N_s$  samples from  $U_s$ .
-

- 
4. Label the selected samples  $U'_s$ .
  5. Add the labeled samples  $L'_s$  to the training set  $L$  and remove them from  $U$ .
- Until** the predefined convergence condition is not satisfied.
- 

In the next subsections, we present the different active learning strategies proposed in this chapter. First, we focus on solutions for GP regression and then we consider SVM regression.

### 6.3.1. Active Learning Strategies for GP Regression

#### 6.3.1.1. Distance from the Closest Training Sample

The first strategy, named TRd in the rest of the paper, consists to calculate for each sample  $\mathbf{x}_j$  ( $j = n+1, n+2, \dots, n+m$ ) the kernel distances (covariance similarities)  $\mathbf{d}_j \in \mathbb{R}^n = [d_{j,1}, d_{j,2}, \dots, d_{j,n}]$  from the samples  $\mathbf{x}_i$  ( $i = 1, 2, \dots, n$ ) already composing the current training set:

$$d_{j,i} = k_{SE}(\mathbf{x}_j, \mathbf{x}_i) \quad (6.22)$$

where  $k_{SE}(\mathbf{x}_j, \mathbf{x}_i)$  is the squared exponential function defined in (6.12). The distance values are thus calculated by means of the same kernel operator used by the GP regressor.

After that, the closest training sample  $t_{MIN,j}$  is identified and the corresponding distance value  $d_{MIN,j}$  is considered as criterion. In this way, we select samples placed in areas of the kernel space not covered by training samples and avoid to choose samples similar to those already present in the current training set.

Algorithm 6.2. encodes the proposed strategy based on the distance from the closest training sample.

---

**Algorithm 6.2.:** GP Active Learning based on Distance from the Closest Training Sample

---

1. Compute the kernel distances  $\mathbf{d}_j \in \mathbb{R}^n = [d_{j,1}, d_{j,2}, \dots, d_{j,n}]$  from the  $n$  different training samples for each sample  $\mathbf{x}_j$  ( $j = n+1, n+2, \dots, n+m$ ) of the learning set  $U$ .
  2. Identify the training sample  $t_{MIN,j}$  closest to the sample.
  3. Consider the distance value  $d_{MIN,j}$  associated with the training sample  $t_{MIN,j}$ .
  4. Set  $h(j)=d_{MIN,j}$ .
- 

#### 6.3.1.2. Weighted Distance from the Training Samples

In the second method (TRwd), after calculating the distances from the training samples, we do not only consider the closest training sample as done in the strategy TRd, but we weigh opportunely all the distance values. The criterion of selection for the sample  $\mathbf{x}_j$  ( $j = n+1, n+2, \dots, n+m$ ) is given by the following formulation:

$$h_{TRwd,j} = \sum_{i=1}^n k_{SE}(\mathbf{x}_j, \mathbf{x}_i) \quad (6.23)$$

The farther the considered sample with respect to the training samples, the smaller the value of the function  $h_{TRwd}$ . Therefore, the samples characterized by the lower values of  $h_{TRwd}$  are selected.

Algorithm 6.3. summarizes the proposed method based on the weighted distance from the training samples.

---

**Algorithm 6.3.:** GP Active Learning based on Weighted Distance from the Training Samples

---

1. Compute the kernel distances  $\mathbf{d}_j \in \mathbb{R}^n = [d_{j,1}, d_{j,2}, \dots, d_{j,n}]$  from the  $n$  different training samples for each sample  $\mathbf{x}_j$  ( $j = n+1, n+2, \dots, n+m$ ) of the learning set  $U$ .
  2. Compute the weighted distance  $h_{TRwd,j}$  using (6.23).
  3. Set  $h(j)=h_{TRwd,j}$ .
-

### 6.3.1.3. Variance on Predictions

The third strategy (VoP) is based on the measure of variance on the predictions defined in (6.11). This value expresses a confidence measure associated by the model to the output and therefore provides an information on the reliability of the estimations. The selection criterion for the sample  $\mathbf{x}_j$  ( $j = n+1, n+2, \dots, n+m$ ) is given by:

$$h_{VoP,j} = \sigma_*^2(\mathbf{x}_j) \quad (6.24)$$

where  $\sigma_*^2(\mathbf{x}_j)$  is the variance measure defined in (6.11). The function  $h_{VoP}$  tends to zero when the confidences on the estimations are high. Since we desire to enrich the current training set with new and difficult samples, we choose the samples with the greater values of  $h_{VoP}$ .

Algorithm 6.4. resumes the proposed strategy based on the value of variance on predictions.

---

#### Algorithm 6.4.: GP Active Learning based on Variance on Predictions

---

1. Compute the value of variance on prediction  $h_{VoP,j}$  using (6.11) for each sample  $\mathbf{x}_j$  ( $j = n+1, n+2, \dots, n+m$ ) of the learning set  $U$ .
  2. Set  $h(j) = -h_{VoP,j}$ .
- 

### 6.3.1.4. Pool of Regressors

The last technique (PoR) is based on a pool of regressors constructed by bagging. Considering the original training set  $L$ ,  $n_s$  training subsets are constructed by randomly choosing a percentage  $p_s$  of samples from  $L$ . Each training subset is considered independently from each other and used to train a different regressor. In this way,  $n_s$  parallel regressors are designed. In particular, the target value  $\mu_{j,k}$  is predicted for each sample  $\mathbf{x}_j$  ( $j = n+1, n+2, \dots, n+m$ ) and for each regressor  $r_k$  ( $k=1, 2, \dots, n_s$ ). Therefore,  $n_s$  different estimations are obtained for each sample. Finally, the different estimations are combined opportunely by calculating the variance value on them:

$$h_{PoR,j} = \frac{1}{n_s} \sum_{k=1}^{n_s} (\mu_{j,k} - \bar{\mu}_j)^2 \quad (6.25)$$

where

$$\bar{\mu}_j = \frac{1}{n_s} \sum_{k=1}^{n_s} \mu_{j,k} . \quad (6.26)$$

The samples characterized by the greater disagreements between the different regressors, i.e. the greater values of variance, are selected. Indeed, a high disagreement means that the corresponding sample has been estimated with high uncertainty, and thus adding it to the training set could be useful to improve the regression process.

Algorithm 6.5. summarizes the proposed methodology based on the pool of regressors.

---

#### Algorithm 6.5.: GP Active Learning based on Pool of Regressors

---

1. Construct  $n_s$  different training subsets  $L_g$  ( $g=1, 2, \dots, n_s$ ) by randomly selecting a percentage  $p_s$  of samples from  $L$ .
  2. Predict the target value of each sample  $\mathbf{x}_j$  ( $j = n+1, n+2, \dots, n+m$ ) of the learning set  $U$  for each regressor  $r_k$  ( $k=1, 2, \dots, n_s$ ).
  3. Compute the variance on the predictions  $h_{PoR,j}$  given by the different regressors using (6.25).
  4. Set  $h(j) = -h_{PoR,j}$ .
-

### 6.3.2. Active Learning Strategies for SVM Regression

#### 6.3.2.1. Distance from the Closest Support Vector

The first strategy (SVd) proposed for the SVM regression is very similar to the TRd strategy presented previously for GP regression. However, while for TRd we calculate the distances with respect to all training samples, in this case we consider only the training samples identified as SVs after the regressor training on  $L$ . This is motivated by the fact that while for GP regression all training samples contribute to describe the regression model, for SVM only the SVs are necessary to define the regression function.

Algorithm 6.6. describes the proposed strategy based on the distance from the closest SV.

---

#### Algorithm 6.6.: SVM Active Learning based on Distance from the Closest Support Vector

---

1. Identify the  $S_n$  support vectors of the regressor on the training set  $L$ .
  2. Compute the kernel distances  $\mathbf{d}_j \in R^{S_n} = [d_{j,1}, d_{j,2}, \dots, d_{j,S_n}]$  from the  $S_n$  different support vectors for each sample  $\mathbf{x}_j$  ( $j = n+1, n+2, \dots, n+m$ ) of the learning set  $U$ .
  3. Identify the support vector  $s_{MIN,j}$  closest to the sample.
  4. Consider the distance value  $d_{MIN,j}$  associated with the support vector  $s_{MIN,j}$ .
  5. Set  $h(j)=d_{MIN,j}$ .
- 

#### 6.3.2.2. Distance from the Support Vectors

The second method (SVd2) is based as the previous one on the distance values from the support vectors. However, more complex sorting and selection strategies are performed in order to take into account the sample distribution in the feature space. First, for each sample  $\mathbf{x}_j$  ( $j = n+1, n+2, \dots, n+m$ ) the index  $s_{MIN,j}$  of the closest support vector is identified and the corresponding distance value  $d_{MIN,j}$  is calculated. Then, for each sample  $\mathbf{x}_j$  ( $j = n+1, n+2, \dots, n+m$ ) we define as  $\tilde{\alpha}_j$  the absolute value of the Lagrange multiplier associated with the closest support vector. We recall that the Lagrange multipliers weigh each training sample according to its importance in determining a solution. The most important training samples are those for which the corresponding Lagrange multipliers are in absolute terms equal to the regularization parameter  $C$ . At this point, the samples of the learning set  $U$  are ordered first in function of the value  $\tilde{\alpha}_j$  and then in function of the distance value  $d_{MIN,j}$ . The final selection is obtained from this sorted set after including an additional selection constraint. In particular, if the new sample to select shares the same closest support vector with a sample already selected at that iteration, it is discarded. In this way, we limit the selection of similar and redundant samples and select samples distributed as most as possible over the feature space.

Algorithm 6.7. encodes the proposed method based on the distance from the SVs.

---

#### Algorithm 6.7.: SVM Active Learning based on Distance from the Support Vectors

---

1. Identify the  $S_n$  support vectors of the regressor on the training set  $L$ .
  2. Compute the kernel distances  $\mathbf{d}_j \in R^{S_n} = [d_{j,1}, d_{j,2}, \dots, d_{j,S_n}]$  from the  $S_n$  different support vectors for each sample  $\mathbf{x}_j$  ( $j = n+1, n+2, \dots, n+m$ ) of the learning set  $U$ .
  3. Identify the support vector  $s_{MIN,j}$  closest to the sample.
  4. Consider the distance value  $d_{MIN,j}$  associated with the support vector  $s_{MIN,j}$ .
  5. Consider the absolute value  $\tilde{\alpha}_j$  of the Lagrange multiplier associated with the support vector  $s_{MIN,j}$ .
  6. Set  $h_1(j)=-\tilde{\alpha}_j$ ,  $h_2(j)=d_{MIN,j}$ .
  7. Select first in function of  $h_1$  and then in function of  $h_2$ , but if the new sample to select shares the same closest support vector with a sample already selected at that iteration, it is discarded.
-

### 6.3.2.3. Pool of Regressors

The last strategy (PoR) is identical to that presented previously for GP regression. We refer the reader to Section 6.3.1.4. for more details.

## 6.4. Experiments

### 6.4.1. Data Set Description and Experimental Setup

In order to validate the proposed active learning methods, we have conducted an experimental study on simulated and real data sets.

The first data set refers to multispectral data that simulates the spectral behavior of the chlorophyll concentration in subsurface case I + case II (open and coastal) waters, through the first eight channels (412-618 nm) of the multispectral Medium Resolution Imaging Spectrometer (MERIS) satellite sensor. These channels are the most useful for sea color applications and, in particular, for the analysis of chlorophyll concentration. We refer the reader to [5] for a more detailed description on the simulation procedure adopted to generate these data. The range of variation of the chlorophyll concentration is from 0.02 to 54 mg/m<sup>3</sup>.

The second data set is the SeaWiFS Bio-optical Algorithm Mini-Workshop (SeaBAM) [33] one. It represents real measurements of chlorophyll concentration, mostly in case I (open) waters, around the U.S. and Europe related to five different Sea-viewing Wide Field-of-view Sensor (SeaWiFS) wavelengths (412, 443, 490, 510, and 555 nm). The chlorophyll concentration values span an interval between 0.02 and 32.79 mg/m<sup>3</sup>.

The radiance values related to both MERIS and SeaBAM data sets were converted to the logarithmic domain. A statistical motivation of this preprocessing step is that biooptical quantities are assumed log-normally distributed [34].

In all the following experiments, for both data sets, all the available samples were split in two sets, corresponding to learning set  $U$  and test set. In particular, for the MERIS data set, 1000 and 4000 samples were considered for learning and test sets, respectively. Analogously, 460 and 459 samples were used for the SeaBAM data set. The initial training samples were selected randomly from the learning set  $U$ . For the MERIS data set, starting from 50 samples, the active learning algorithms were run until all the learning samples were included in the training set, adding 50 samples at each iteration. Similarly, for the SeaBAM data set, 25 samples were added at each iteration by starting from 60 samples.

GP and SVM regressors were also trained on the entire learning set (i.e., all the 1000 and 460 training samples for the MERIS and the SeaBAM data sets, respectively) in order to have a reference-training scenario, called "full" training. On one hand, the regression results obtained in this way represent a lower bound for the errors. On the other hand, we expect that the upper error bound will be given by the random sample selection strategy (Ran). We recall that the purpose of any active learning strategy is to converge to the performance of the "full" training scenario faster than the Ran method.

Regression performances were evaluated in terms of two measures: 1) the mean squared error (MSE) and 2) the squared correlation coefficient (R2)

$$MSE = \frac{1}{t} \sum_{i=1}^t (\hat{y}_i - y_i)^2 \quad (6.27)$$

$$R2 = \frac{\sum_{i=1}^t (\hat{y}_i - \bar{\hat{y}})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^t (\hat{y}_i - \bar{\hat{y}})^2 \sum_{i=1}^t (y_i - \bar{y})^2}} \quad (6.28)$$

where  $t$  is the number of test samples.

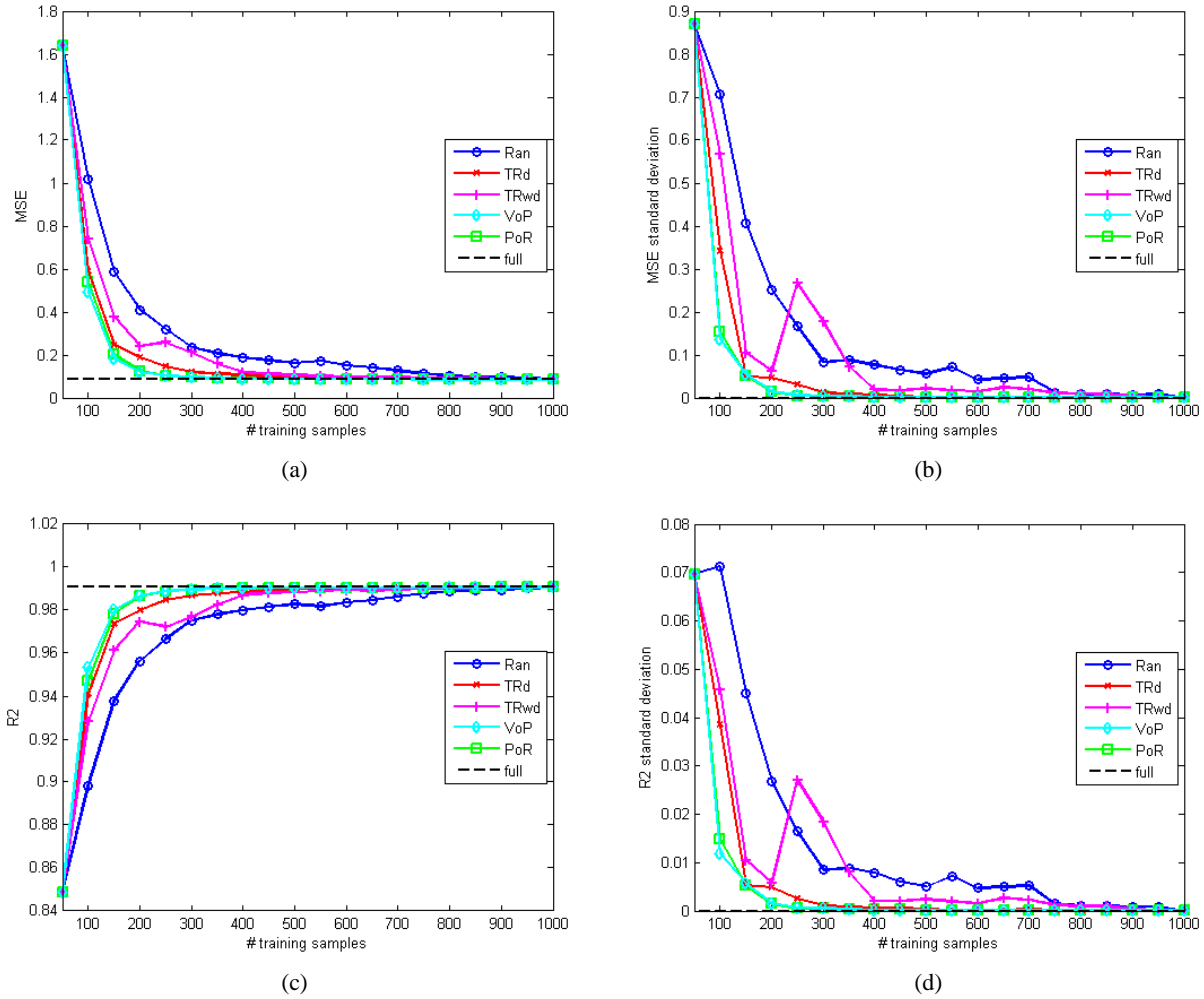


Fig. 6.2. Performances achieved on the MERIS data set for GP regression in terms of (a) MSE, (b) MSE standard deviation, (c) R2, and (d) R2 standard deviation.

Concerning the parameter setting, in the case of GP regression, we considered a squared exponential covariance function. The hyperparameters were tuned using the Bayesian model selection method. The noise variance, the length scale, and the signal variance were varied in the ranges  $[0.0001, 1]$ ,  $[0.01, 10]$ , and  $[0.001, 10]$ , respectively. For the SVM regression, we adopted a Gaussian kernel. This choice is motivated by the good prediction accuracy and the limited computational complexity associated to this kernel. The regularization and kernel width parameters were tuned empirically at each iteration by cross validation (CV) in the ranges  $[2^{-9}, 2^9]$  and  $[2^{-11}, 2^3]$ , respectively. Regarding the PoR approach, for both GP and SVM regression, the number of training subsets  $n_s$  and the percentage  $p_s$  of samples selected randomly from  $L$  were set empirically to 5 and 0.8, respectively. In order to assess the influence of these parameters, we performed an empirical analysis which showed a scarce sensitivity of PoR to them.

The entire active learning process was run for each method ten times, each time with a different initial training set to yield statistically reliable results. At each run, the initial training samples were chosen in a completely random way. Moreover, in order to take into account possible intrinsic variation of results because of the CV procedure, we ran ten times GP and SVM regressors on the entire learning set, each time by randomly reordering the samples.

Note that, for Figs. 6.2-6.6 that will be introduced in the next subsection, each graph shows the results in function of the number of interactions. All results are averaged over ten runs of the approaches. For GP regression, acronyms are as follows: Ran = random, TRd = distance from the closest training sample, TRkd



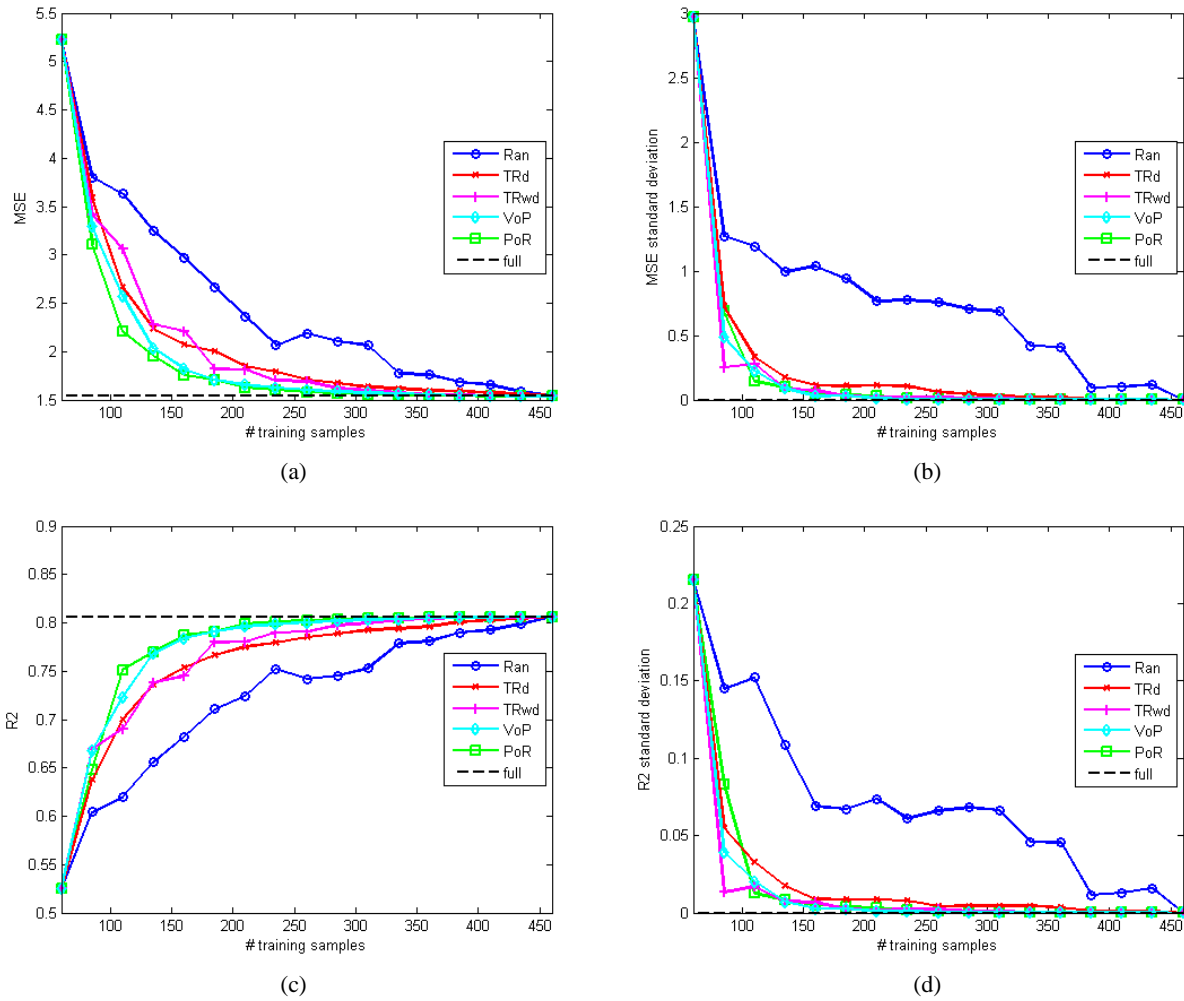


Fig. 6.3. Performances achieved on the SeaBAM data set for GP regression in terms of (a) MSE, (b) MSE standard deviation, (c) R2, and (d) R2 standard deviation.

= kernel distance from the training samples, VoP = variance on predictions, PoR = pool of regressors, full = full GP. For SVM regression, they are: Ran = random, SVd = distance from the closest support vector, SVd2 = distance from the support vectors, PoR = pool of regressors, full = full SVM.

#### 6.4.2. Experimental Results

Considering the GP regression and the MERIS data set, the performances for the “full” regressor in terms of MSE and R2 are equal to 0.086 and 0.991, respectively. In Figs. 6.2(a)-(d), we show the results in function of the number of training samples for the proposed active learning strategies and the random one. First, we note that in general the active selection of the training samples allows a faster convergence to the “full” result with respect to the random strategy, both in terms of accuracies and standard deviations (and thus stability). The active selection allows to converge to the “full” result using about 300 training samples, which represent 30% of the entire learning set. Instead, the entire set of training samples is necessary for the Ran method to converge. Moreover, before convergence, all the proposed active learning strategies give an improvement with respect to the Ran one. In particular, the VoP and PoR methods exhibit the best performances. This means that similar values of accuracies can be obtained using a minor quantity of training samples, which implies a reduction of the manual labeling work and a decreasing of the computational time necessary to train the regressor. We note how for the TRkd method an anomalous peak is

TABLE 6.I  
MSE, R2, AND STANDARD DEVIATIONS ( $\sigma$ ) ACHIEVED FOR THE GP REGRESSION ON  
(A) THE MERIS AND (B) THE SEABAM DATA SETS

(a)

Method	# training samples	MSE	$\sigma_{MSE}$	R2	$\sigma_{R2}$
Full	1000	0.086	0.000	0.991	0.000
Initial	50	1.638	0.869	0.849	0.070
Ran	150	0.585	0.406	0.938	0.045
TRd		0.247	<b>0.049</b>	0.974	<b>0.005</b>
TRwd		0.378	0.105	0.961	0.010
VoP		<b>0.184</b>	0.054	<b>0.980</b>	0.006
PoR		0.201	0.051	0.978	<b>0.005</b>
Ran	300	0.237	0.084	0.975	0.008
TRd		0.121	0.012	0.987	0.001
TRwd		0.212	0.177	0.977	0.018
VoP		<b>0.095</b>	<b>0.005</b>	<b>0.990</b>	<b>0.000</b>
PoR		0.097	<b>0.005</b>	0.989	<b>0.000</b>

(b)

Method	# training samples	MSE	$\sigma_{MSE}$	R2	$\sigma_{R2}$
Full	460	1.536	0.000	0.806	0.000
Initial	60	5.221	2.968	0.526	0.215
Ran	160	2.972	1.038	0.682	0.069
TRd		2.073	0.113	0.754	0.009
TRwd		2.210	0.074	0.745	0.007
VoP		1.818	<b>0.029</b>	0.784	<b>0.003</b>
PoR		<b>1.753</b>	0.047	<b>0.787</b>	0.005
Ran	310	2.062	0.687	0.753	0.066
TRd		1.632	0.028	0.793	0.004
TRwd		1.601	0.010	0.800	0.001
VoP		1.573	<b>0.003</b>	0.803	<b>0.000</b>
PoR		<b>1.557</b>	0.005	<b>0.805</b>	<b>0.000</b>

verified around 250 training samples. This is due to a bad estimation of the hyperparameters by the Bayesian model selection method in one run of the experiments.

The obtained results are shown in greater detail in Table 6.I(a). In particular, we considered the performances obtained after 3 and 6 iterations of the iterative process, which corresponds to 150 and 300 samples used to train the regressor, respectively. We report the values of MSE, R2, and standard deviations associated with the accuracies ( $\sigma_{MSE}$ ,  $\sigma_{R2}$ ). The best results are highlighted in bold font. As it can be seen, the proposed strategies are characterized by better performances with respect to the Ran method from different points of view. First, better values of accuracies are obtained using the same number of training samples. Then, better values of standard deviations associated with the accuracies are verified. Indeed, minor values of standard deviation mean that the proposed strategies exhibit a greater level of stability with respect to the random selection of the initial training set.

Concerning the SeaBAM data set, the results confirm the observations drawn for the MERIS one. The graphs with the accuracies in function of the number of training samples are illustrated in Fig. 6.3(a)-(d). For the “full” regressor the performances in terms of MSE and R2 are equal to 1.536 and 0.806, respectively. Also for this set of experiments, the proposed active learning strategies give a faster convergence to the “full accuracy” and better performances before convergence with respect to the Ran method. In particular, the methods VoP and PoR confirm the best results. In this case, they converge to the “full” accuracy using about 310 training samples. The results obtained after 5 and 11 iterations of the active learning process, which correspond to 160 and 310 training samples respectively, are summarized in Table 6.I(b).

The effectiveness of the active learning approach is also confirmed for the SVM regression. The graphs with the accuracies for the MERIS data set are shown in Fig. 6.4(a)-(d). In this case, the performances of the

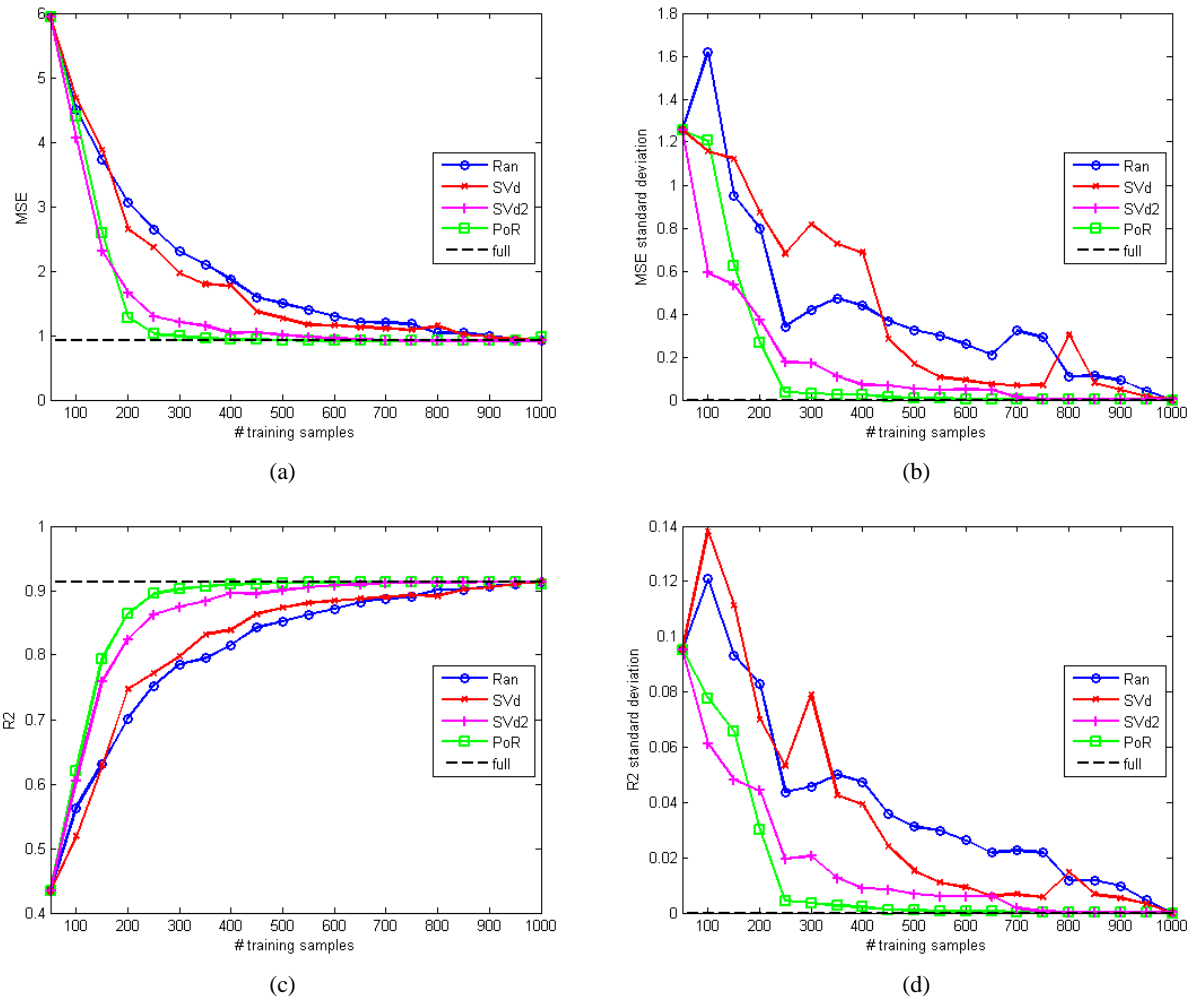


Fig. 6.4. Performances achieved on the MERIS data set for SVM regression in terms of (a) MSE, (b) MSE standard deviation, (c) R2, and (d) R2 standard deviation.

“full” regressor are equal to 0.916 and 0.913 in terms of MSE and R2, respectively. The method SVd, in which the samples more distant from the current SVs are selected, exhibits poor performances, which are very similar to those obtained by the Ran selection. Instead, good improvements are verified using the SVd2 and the PoR strategies. In these two cases, the convergence to the “full” results is obtained using about 400 training samples. The results corresponding to 150 and 300 training samples are detailed in Table 6.II(a).

Finally, in Fig. 6.5(a)-(d) we show the results using the SVM regression for the SeaBAM data set, for which the “full” accuracies correspond to 1.305 and 0.834 in terms of MSE and R2, respectively. We note that small accuracy variations are observed at convergence. This is due to the CV procedure, which may lead to different best parameters for the different runs, depending on the order of the samples. As for the MERIS data set, bad performances are obtained for the SVd strategy, while very good results are achieved for the SVd2 and PoR ones, for which the convergence is verified using about 150 training samples. On an average, it is noteworthy that at convergence the SVd2 and PoR methods give values of accuracies slightly better than the “full” regressor. This can be explained by the way the SVM model selection is carried out. Indeed, as can be seen in the figure, in the case all the 460 samples are collected, the CV procedure applied to the different strategies does not reproduce the “full” accuracy since the training sets accumulated at the last iteration result with different sample ordering. The results obtained using 160 and 310 training samples are illustrated in Table 6.II(b).

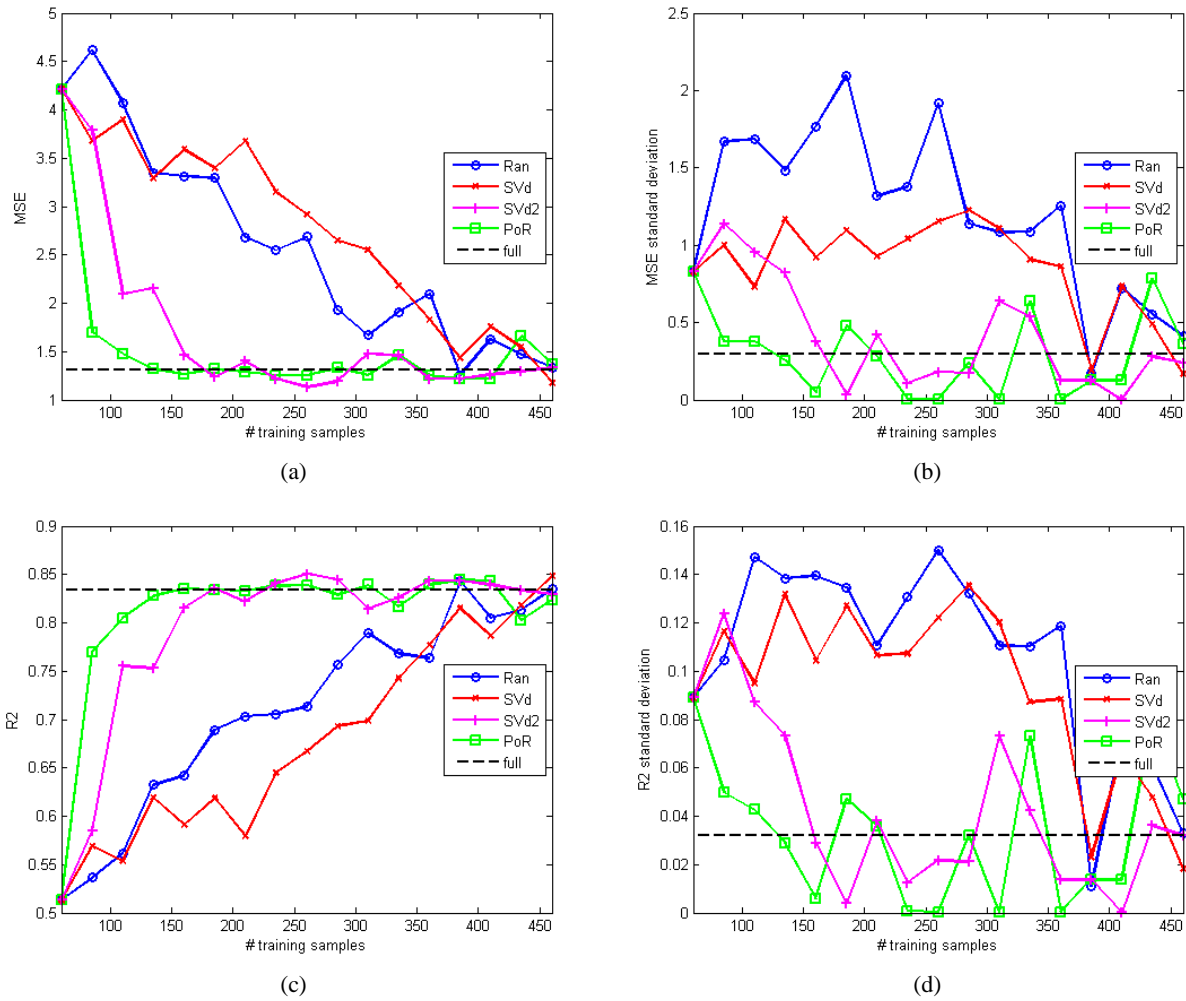


Fig. 6.5. Performances achieved on the SeaBAM data set for SVM regression in terms of (a) MSE, (b) MSE standard deviation, (c) R2, and (d) R2 standard deviation.

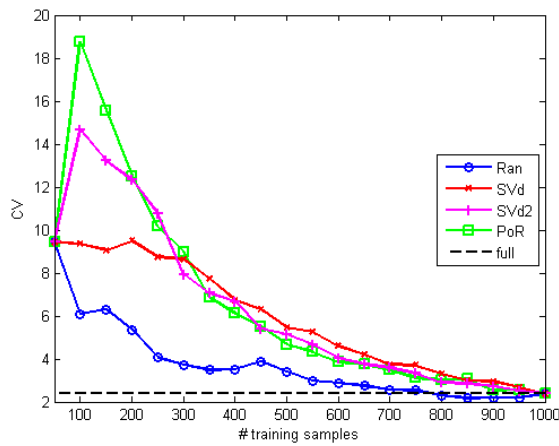
TABLE 6.II  
MSE, R2, AND STANDARD DEVIATIONS ( $\sigma$ ) ACHIEVED FOR THE SVM REGRESSION ON  
(A) THE MERIS AND (B) THE SEABAM DATA SETS

(a)

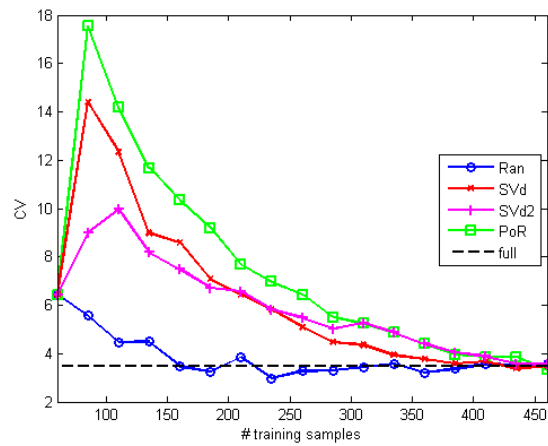
Method	# training samples	MSE	$\sigma_{\text{MSE}}$	R2	$\sigma_{\text{R2}}$	CV	#SV
Full	1000	0.916	0.000	0.913	0.000	2.41	240
Initial	50	5.936	1.256	0.434	0.095	9.43	38.8
Ran	150	3.725	0.949	0.631	0.093	6.32	87.6
SVd		3.880	1.122	0.629	0.112	9.07	92.6
SVd2		<b>2.303</b>	<b>0.537</b>	0.758	<b>0.048</b>	13.22	97.2
PoR		2.583	0.625	<b>0.795</b>	0.066	15.57	118.3
Ran	300	2.301	0.419	0.786	0.046	3.72	129.6
SVd		1.967	0.817	0.797	0.079	8.65	130.8
SVd2		1.207	0.173	0.875	0.021	7.94	165.9
PoR		<b>0.983</b>	<b>0.031</b>	<b>0.902</b>	<b>0.004</b>	8.98	184.7

(b)

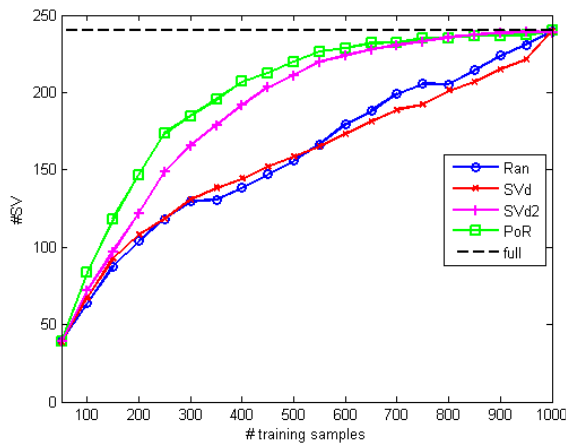
Method	# training samples	MSE	$\sigma_{MSE}$	R2	$\sigma_{R2}$	CV	#SV
Full	460	1.305	0.294	0.834	0.032	3.48	156.6
Initial	60	4.214	0.828	0.513	0.089	6.44	30.7
Ran	160	3.314	1.762	0.642	0.140	3.47	68.3
SVd		3.589	0.919	0.591	0.104	8.58	65.7
SVd2		1.463	0.373	0.816	0.029	7.49	97.1
PoR		<b>1.267</b>	<b>0.050</b>	<b>0.836</b>	<b>0.006</b>	10.3	105.3
Ran	310	1.667	1.084	0.789	0.111	3.31	111.4
SVd		2.550	1.107	0.698	0.120	4.35	88.9
SVd2		1.477	0.639	0.814	0.073	5.27	147.2
PoR		<b>1.256</b>	<b>0.002</b>	<b>0.839</b>	<b>0.001</b>	5.24	141.4



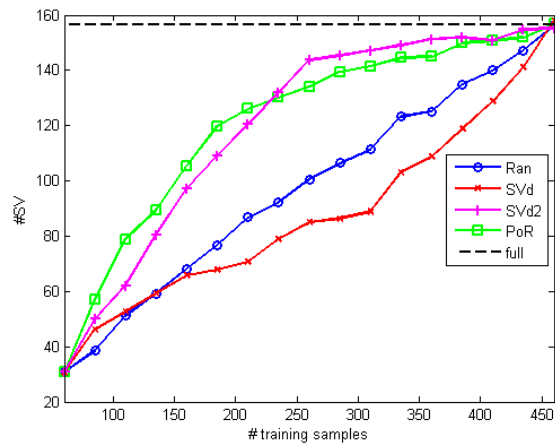
(a)



(b)



(c)



(d)

Fig. 6.6. Performances achieved for SVM regression on (a), (c) the MERIS and (b), (d) the SeaBAM data sets in terms of (a), (b) CV accuracy and (c), (d) #SV.

To better understand the behaviours of the active learning strategies proposed for SVM regression, we show in Fig. 6.6(a)-(d) the evolution at each iteration of the CV accuracy and the number of SVs (#SV) for (a), (c) the MERIS and (b), (d) the SeaBAM data set, respectively. It is interesting to observe that the value of CV tends to increase in the first iterations, while we have a decrease of the CV value only when a sufficient number of samples have been added to the training set. The increase of the CV value means that samples difficult to estimate are added to the training set. However, these new samples are highly informative and thus allow improving the generalization performance (i.e., the accuracy on the test samples). A completely different behavior is obtained for the Ran strategy, for which the CV value tends to decrease

from the beginning. Analogously, we note that in the SVd2 and PoR methods the #SV value tends to increase faster than the Ran method. The fast increment of the #SV for the active learning strategies shows clearly that the samples added to the training set are really important for the regression process. The obtained results in terms of CV and #SV are detailed in Table 6.II(a),(b). Finally, we note that similar observations cannot be done for the GP regression, for which the hyperparameters have not been estimated by CV technique, but using the Bayesian model selection method. Moreover, while for SVM regression only the SVs describe the regression function, for GP regression all training samples contribute to define the regression model.

## 6.5. Conclusion

In this chapter, the active learning approach has been introduced to deal with the problem of training sample collection for regression problems related to the estimation of biophysical parameters from remote sensing data. Starting from an initial training set, an iterative process selects from an unlabeled data set the samples more significant for the regression process, i.e., those able to give small prediction errors while minimizing the number of training samples and the computational costs required by the regressor. In particular, we have proposed several strategies specifically developed for two state-of-the-art regression approaches, namely GP and SVM. For GP regression, the first two methods (TRd and TRkd) are based on adding samples that are distant from the current training samples in the kernel space, while the third one (PoR) uses a pool of regressors in order to select the samples with the greater disagreements between the regressors of the pool. Finally, the last strategy (VoP) exploits an intrinsic GP regression outcome to pick up the most difficult samples. For SVM regression, the method based on the pool of regressors (PoR) and two additional strategies (SVd and SVd2) based on the selection of the samples distant from the current support vectors are proposed.

The experimental results obtained on simulated MERIS and real SeaBAM data sets show good capabilities of the proposed strategies for selecting significant samples. In general, the proposed methods are characterized by higher performances in terms of both accuracy and stability with respect to a completely random selection strategy. Comparing them, the best methodologies seem PoR and VoP for GP regression and SVd2 and PoR for SVM regression.

In this chapter, though we focused on GP and SVM regression, the active selection of the training samples could be used in combination with other supervised regression approaches. Moreover, while in this work the initial training set was chosen in a random way, a more sophisticated initialization strategy could be envisioned in order to improve further the performances of the active learning approach.

## 6.6. Acknowledgment

The authors would like to thank Prof. G. Corsini (University of Pavia, Italy) and the SeaBAM group for providing the data used in the experiments.

## 6.7. References cited in Chapter 6

- [1] D. G. Goodenough, A. S. Bhogall, H. Chen, and A. Dyk, "Comparison on methods for estimation of Kyoto protocol products of forests from multitemporal Landsat," in *Proc. IGARSS*, Sidney, AUS, Jul. 2001, vol. 2, pp. 764–767.
- [2] D. Del Frate, A. Ortenzi, S. Casadio, and C. Zehner, "Application of neural algorithms for a real-time estimation of ozone profiles from GOME measurements," *IEEE Trans. Geosci. Remote Sens.*, vol. 40, no. 10, pp. 2263–2270, Oct. 2002.
- [3] L. Bruzzone and F. Melgani, "Robust multiple estimator systems for the analysis of biophysical parameters from remotely sensed data," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 1, pp. 159–174, Jan. 2005.

- [4] D. S. Kimes, Y. Knyazikhin, J. L. Privette, A. A. Abuelgasim, and F. Gao, "Inversion methods for physically-based models," *Remote Sens. Rev.*, vol. 18, no. 2–4, pp. 381–439, Sep. 2000.
- [5] P. Cipollini, G. Corsini, M. Diani, and R. Grasso, "Retrieval of sea water optically active parameters from hyperspectral data by means of generalized radial basis function neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 39, no. 7, pp. 1508–1524, Jul. 2001.
- [6] D. D'Alimonte and G. Zibordi, "Phytoplankton determination in an optically complex coastal region using a multilayer perceptron neural network," *IEEE Trans. Geosci. Remote Sens.*, vol. 41, no. 12, pp. 2861–2868, Dec. 2003.
- [7] H. Zhan, P. Shi, and C. Chen, "Retrieval of oceanic chlorophyll concentration using support vector machines," *IEEE Trans. Geosci. Remote Sens.*, vol. 41, no. 12, pp. 2947–2951, Dec. 2003.
- [8] G. Camps-Valls, L. Bruzzone, J. L. Rojo-Alvarez, and F. Melgani, "Robust support vector regression for biophysical variable estimation from remotely sensed images," *IEEE Geosci. Remote Sens. Lett.*, vol. 3, no. 3, pp. 339–343, Jul. 2006.
- [9] D. Tuia, J. Verrelst, L. Alonso, F. Pérez-Cruz, and G. Camps-Valls, "Multioutput support vector regression for remote sensing biophysical parameter estimation," *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 4, pp. 804–808, Jul. 2011.
- [10] L. Pasolli, F. Melgani, and E. Blanzieri, "Gaussian process regression for estimating chlorophyll concentration in subsurface waters from remote sensing data," *IEEE Geosci. Remote Sens. Lett.*, vol. 7, no. 3, pp. 464–468, Jul. 2010.
- [11] Z.-H. Zhou and M. Li, "Semi-supervised regression with co-training," in *Proc. Int. Joint Con. Artif. Intell.*, 2005, pp. 908–913.
- [12] Y. Bazi and F. Melgani, "Semisupervised PSO-SVM regression for biophysical parameter estimation," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 6, pp. 1887–1895, Jun. 2007.
- [13] Y. Bazi and F. Melgani, "Semisupervised Gaussian process regression for biophysical parameter estimation," in *Proc. IGARSS*, Honolulu, HI, Jul. 2010, vol. 1, pp. 4248–4251.
- [14] P. Mitra, C. A. Murthy, and S. K. Pal, "A probabilistic active support vector learning algorithm," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 3, pp. 413–418, Mar. 2004.
- [15] E. Pasolli and F. Melgani, "Active learning methods for electrocardiographic signal classification," *IEEE Trans. Inf. Technol. Biomed.*, vol. 14, no. 6, pp. 1405–1416, Nov. 2010.
- [16] P. Mitra, B. Uma Shankar, and S. Pal, "Segmentation of multispectral remote sensing images using active support vector machines," *Pattern Recogn. Lett.*, vol. 25, no. 9, pp. 1067–1074, Jul. 2004.
- [17] Y. Zhang, X. Liao, and L. Carin, "Detection of buried targets via active selection of labeled data: application to sensing subsurface UXO," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 11, pp. 2535–2543, Nov. 2004.
- [18] Q. Liu, X. Liao, and L. Carin, "Detection of unexploded ordnance via efficient semisupervised and active learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 9, pp. 2558–2567, Sep. 2008.
- [19] S. Rajan, J. Ghosh, and M. M. Crawford, "An active learning approach to hyperspectral data classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 4, pp. 1231–1242, Apr. 2008.
- [20] J. Li, J. Bioucas-Dias, and A. Plaza, "Semisupervised hyperspectral image segmentation using multinomial logistic regression with active learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 11, pp. 4085–4098, Nov. 2010.
- [21] D. Tuia, F. Ratle, F. Pacifici, M. F. Kanevski, and W. J. Emery, "Active learning methods for remote sensing image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 7, pp. 2218–2232, Jul. 2009.
- [22] E. Pasolli, F. Melgani, and Y. Bazi, "Support vector machine active learning through significance space construction," *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 3, pp. 431–435, May 2011.
- [23] D. Cohn, Z. Ghahramani, and M. Jordan, "Active learning with statistical models," *J. Artif. Intell. Res.*, vol. 4, pp. 129–145, Mar. 1996.
- [24] K. Fukumizu, "Statistical active learning in multilayer perceptrons," *IEEE Trans. Neural Netw.*, vol. 11, no. 1, pp. 17–26, Jan. 2000.
- [25] R. Castro, R. Willett, and R. Nowak, "Faster rates in regression via active learning," *Adv. Neural Inf. Process. Syst.*, vol. 18, pp. 179–186, 2006.
- [26] M. Sugiyama, "Active learning in approximately linear regression based on conditional expectation of generalization error," *The Journal of Machine Learning Research*, vol. 7, pp. 141–166, Jan. 2006.

- [27] R. Burbidge, J. J. Rowland, and R. D. King, “Active learning for regression based on query by committee,” *Intelligent Data Engineering and Automated Learning*, pp. 209–218, 2007.
- [28] M. Sugiyama and N. Rubens, “A batch ensemble approach to active learning with model selection,” *Neural Networks*, vol. 21, no. 9, pp. 1278–1286, Nov. 2008.
- [29] M. Sugiyama and S. Nakajima, “Pool-based active learning in approximate linear regression,” *Mach. Learn.*, vol. 75, no. 3, pp. 249–274, Jan. 2009.
- [30] J. Paisley, X. Liao, and L. Carin, “Active learning and basis selection for kernel-based linear models: a Bayesian perspective,” *IEEE Trans. Signal Process.*, vol. 58, no. 5, pp. 2686–2700, May 2010.
- [31] C. E. Rasmussen and C. K. I. Williams, *Gaussian Process for Machine Learning*. Cambridge, MA: MIT Press, 2006.
- [32] V. Vapnik, *Statistical Learning Theory*. New York: Wiley, 1998.
- [33] J. E. O’Reilly and S. Maritorena, “SeaBAM evaluation data set,” *The SeaWiFS Bio-Optical Algorithm Mini-Workshop (SeaBAM)*, 1997, Santa Barbara, CA: Univ. California. [Online]. Available: <http://seabass.gsfc.nasa.gov/seabam>.
- [34] J. W. Campbell, “The lognormal distribution as a model for the bio-optical variability in the sea,” *J. Geophys. Res.*, vol. 100, no. C7, pp. 13237–13254, 1995.



## 7. Active Learning for Spectroscopic Data Regression

*Abstract – In this chapter, we introduce an active learning approach for the estimation of chemical concentrations from spectroscopic data. Its main objective is to opportunely collect training samples in such a way to minimize the error of the regression process while minimizing the number of training samples to use, and thus to reduce the costs related to the training sample collection. In particular, we propose different active learning strategies specifically developed for regression approaches based on partial least squares regression (PLSR) and support vector machine (SVM). For PLSR, the first method is based on adding samples that are distant from the current training samples in the feature space, while the second one uses a pool of regressors in order to select the samples with the greater disagreements between the different regressors of the pool. For SVM, the method based on the pool of regressors and an additional strategy based on the selection of the samples distant from the support vectors are proposed. Experimental results on three different real data sets are reported and discussed.*

The work presented in this chapter has been submitted to *J. Chemometr.*; Co-authors: F. Douak, F. Melgani, N. Alajlan, Y. Bazi, N. Benoudjit.

## 7.1. Introduction

Spectroscopy is an important technology for product analysis and quality control in different chemical fields. For example, it has been applied successfully in pharmaceutical [1], [2], food [3] and textile industries [4]. Chemical analysis by spectroscopy results interesting since it allows a fast acquisition of a large number of spectral data, which can be analyzed in order to yield accurate estimations of the concentration of the chemical component of interest in a given product.

From a methodological point of view, the problem of concentration estimation can be viewed as an inverse modelling issue in which it is necessary to define a model that relates the acquired observations to the concentration of interest. The estimation of the model is typically done by adopting supervised regression techniques, which require the availability of a set of training samples. By training samples, we mean pairs of spectral data acquired by the spectrometer and measurements of the concentration to estimate. In the literature, two main approaches of regression have been proposed. The first one is based on linear models, appreciated for their simplicity, such as multiple linear regression, principal component regression and partial least squares regression (PLSR) [5]. The second approach makes use of nonlinear models. They are characterized by greater computational complexity, but they can give better performances when a strong nonlinearity between the acquired spectral data and the concentrations to estimate is present. In this context, two state-of-the-art methods are radial basis functions neural network (RBFN) and support vector machine (SVM) [6], [7].

In general, the regression process is done by assuming that the training set is composed by a sufficient number of samples in order to obtain a reliable model and accurate estimations. However, from a practical point of view, the process of collection of training samples is not trivial, because the concentration measurements associated with the acquired spectral data have to be performed by human experts and thus are subject to costs in terms of time and money. For this reason, the number of available training samples is typically limited and performances can be affected consequently due to training sample scarcity.

A solution to the problem of training sample collection is given by the active learning approach. Starting from a small training set, additional samples are selected from a large amount of unlabeled data. These samples are labeled by the expert and added to the training set. The process is iterated until a stopping criterion is reached. In particular, active learning strategies have been applied successfully in the classification context [8] in different fields [9]-[11]. Similarly, the active learning approach has been studied for regression problems by the machine learning and statistics communities, in which it is also known as *optimal experimental design*. After the seminal paper by Cohn et al. [12], in which active learning has been applied to two statistically-based learning architectures, such as mixtures of Gaussians and locally weighted regression, several works have appeared in the last few years. For instance, in [13], the authors focus on the problem of local minima in active learning for neural networks, and two probabilistic solutions are proposed. In [14], after introducing the fundamental limits in a minimax sense of active and passive learning for various function classes, some strategies based on a tree-structured partition of the data are presented. In [15], considering linear regression scenarios, a method using the weighted least-squares learning based on the conditional expectation of the generalization error is proposed. In [16], the authors apply the query by committee approach in the regression context. The main idea is to train a committee of learners and query the labels of the samples where the committee's prediction differ, thus minimizing the variance of the learner by training on samples where variance is largest. In [17], it is suggested to solve the problems of active learning and model selection at the same time in order to improve further the generalization performance. In [18], a solution to the problem of pool-based active learning in linear regression is proposed. In [19], the authors develop a strategy for kernel-based linear regression, in which the proposed greedy algorithm employs a minimum-entropy criterion derived using a Bayesian interpretation of ridge regression. Despite

the promising performance given by the active learning approach in the regression context, nothing similar has been proposed in the chemometrics literature.

The objective of this chapter is to introduce the active learning approach for regression problems for the estimation of concentrations from spectroscopic data. In particular, we propose different active learning strategies specifically developed for two state-of-the-art regression approaches, namely PLSR and SVM. For PLSR, the first method is based on adding samples that are distant from the current training samples in the feature space, while the second one uses a pool of regressors in order to select the samples with the greater disagreements between the different regressors. For SVM, the method based on the pool of regressors and an additional strategy based on the selection of the samples distant from the current support vectors are proposed. To illustrate the capabilities of the proposed strategies, we conducted an experimental study based on three different real data sets: 1) a diesel data set for estimating the cetane number by near-infrared spectroscopy; 2) an orange juice data set where near-infrared reflectance spectroscopy is used to estimate the saccharose concentration; 3) a Tecator data set for the estimation of fat content in meat by mid-infrared spectroscopy. The obtained results show that interesting performances can be achieved.

The remaining part of the chapter is organized as follows. In Section 7.2 the basic mathematical formulations of PLSR are recalled, while for SVR we refer the reader to subsection 6.2.2. In Section 7.3, the active learning strategies proposed for regression problems are described. Section 7.4 presents the data sets used in the experimental analysis and the related results. Finally, conclusions are drawn in Section 7.5.

## 7.2. Partial Least Squares Regression

Let us consider a set of labeled samples  $L = \{\mathbf{x}_i, y_i\}_{i=1}^n$ , where  $\mathbf{x}_i = [x_{i1}, \dots, x_{id}] \in \mathfrak{R}^d$  represents a vector of  $d$  spectral acquisitions and/or processed features and  $y_i \in \mathfrak{R}$  is the associated target value, i.e., the measurement of the concentration value of interest. Let us aggregate all  $\mathbf{x}_i$ 's ( $i=1, \dots, n$ ) into a feature matrix  $X$  and all  $y_i$ 's ( $i=1, \dots, n$ ) into a target vector  $\mathbf{y}$  so that  $L = \{X, \mathbf{y}\}$ . The goal is to infer from the set of labeled samples  $L$  the function  $f(\cdot)$  so that  $y = f(\mathbf{x})$ .

The PLSR aims at finding a linear regression model by projecting data to a new space [20]. In particular, it tries to find the multidimensional direction in the space  $X$  that explains the maximum multidimensional variance direction in the space  $\mathbf{y}$ . The user has to supply the number  $l$  of latent factors in the regression. If it equals the rank of the matrix  $X$ , the method yields simply the least squares regression estimates.

After centering the input  $X$  and  $\mathbf{y}$ , the following steps are performed for each latent factor  $k$  ( $k=1, \dots, l$ ):

*Step 1:* find the weight vector  $\mathbf{w}_k$  by maximizing the covariance between the linear combination

$X_{k-1}\mathbf{w}_k$  and  $\mathbf{y}$  under the constraint that  $\mathbf{w}_k'\mathbf{w}_k = 1$ . This corresponds to find the unit vector  $\mathbf{w}_k$  that maximizes  $\mathbf{w}_k'X_{k-1}'\mathbf{y}_{k-1}$ , i.e., the scaled covariance between  $X_{k-1}$  and  $\mathbf{y}_{k-1}$

$$w_k = \frac{X_{k-1}'\mathbf{y}_{k-1}}{\|X_{k-1}'\mathbf{y}_{k-1}\|}. \quad (7.1)$$

*Step 2:* find the factor score  $\mathbf{t}_k$  as the projection of  $X_{k-1}$  on  $\mathbf{w}_k$ , so that the  $X$ -residuals  $E$

$$X_{k-1} = \mathbf{t}_k\mathbf{w}_k + E. \quad (7.2)$$

Since  $\mathbf{w}_k'\mathbf{w}_k = 1$ , the solution is

$$\mathbf{t}_k = X_{k-1}\mathbf{w}_k. \quad (7.3)$$

*Step 3:* regress  $X_{k-1}$  on  $\mathbf{t}_k$  to find the loadings  $\mathbf{p}_k'$

$$X_{k-1} = \mathbf{t}_k\mathbf{p}_k' + E. \quad (7.4)$$

The least square solution is given by

$$\mathbf{p}_k = X_{k-1}' \mathbf{t}_k / \mathbf{t}_k' \mathbf{t}_k. \quad (7.5)$$

Step 4: regress  $\mathbf{y}_{k-1}$  on  $\mathbf{t}_k$  to find  $\mathbf{q}_k$ , so that the  $\mathbf{y}$ -residuals  $F$

$$\mathbf{y}_{k-1} = \mathbf{t}_k \mathbf{q}_k + F. \quad (7.6)$$

The solution is given by

$$\mathbf{q}_k = \mathbf{y}_{k-1}' \mathbf{t}_k / \mathbf{t}_k' \mathbf{t}_k. \quad (7.7)$$

Step 5: subtract  $\mathbf{t}_k \mathbf{p}_k'$  from  $X_{k-1}$  in order to obtain  $X_k$ . Similarly,  $\mathbf{y}_k$  is obtained by subtracting  $\mathbf{t}_k \mathbf{q}_k'$  from  $\mathbf{y}_{k-1}$ .

After the computation of the latent factors, the matrix  $X$  is deflated by subtracting  $\mathbf{t}_k \mathbf{q}_k'$  from  $X$ . In this way, the model refers to the residuals after previous dimension  $E$  instead of relating to the variables  $X$  themselves

$$E = X_{k-1} - \mathbf{t}_k \mathbf{p}_k' \quad (7.8)$$

$$F = \mathbf{y}_{k-1} - \mathbf{t}_k \mathbf{q}_k. \quad (7.9)$$

Replacing  $X_{k-1}$  and  $\mathbf{y}_{k-1}$  by the residuals  $E$  and  $F$  and increasing  $k$  of one, we obtain

$$X_{k-1} = E \quad (7.10)$$

$$\mathbf{y}_{k-1} = F \quad (7.11)$$

$$k = k + 1. \quad (7.12)$$

The regression coefficients  $\mathbf{b}$  are given by

$$\mathbf{b} = W(P'W)^{-1} \mathbf{q} \quad (7.13)$$

where  $W = (\mathbf{w}_1 | \mathbf{w}_2 | \dots | \mathbf{w}_l)$ ,  $P = (\mathbf{p}_1 | \mathbf{p}_2 | \dots | \mathbf{p}_l)$ ,  $\mathbf{q}' = (q_1, q_2, \dots, q_l)$ .

Finally, the prediction of a generic sample  $\mathbf{x}^*$  is given by

$$y^* = \mathbf{x}^* \mathbf{b}. \quad (7.14)$$

### 7.3. Proposed Active Learning Methods

Let us consider a training set composed initially of  $n$  labeled samples  $L = \{\mathbf{x}_i, y_i\}_{i=1}^n$  and an additional learning set composed of  $m$  unlabeled samples  $U = \{\mathbf{x}_j\}_{j=n+1}^{n+m}$ , with  $m \gg n$ . In order to increase the training set  $L$  with a series of samples chosen from the learning set  $U$  and labeled manually by the expert, an active learning algorithm has the task of choosing them properly so as to minimize the error of the regression process while minimizing the number of learning samples to label, and thus to reduce the costs related to the training sample collection.

In Fig. 7.1, we show the generic flow chart of the active learning approach for regression problems proposed in this chapter. Starting from the initial and small training set  $L$ , the unlabeled samples of the learning set  $U$  are evaluated and sorted using an opportune criterion  $h$ . In particular, we suppose for convention that the criterion  $h$  has to be minimized. At this point, from the sorted samples  $U_s$ , the first  $N_s$  samples are selected, where  $N_s$  is the number of samples to be added in the training set  $L$ . Finally, the selected samples  $U'_s$  are labeled by the human expert and added to the training set  $L$ . The entire process is iterated until the predefined convergence condition is satisfied (e.g., the total number of samples to add to the training set is reached, or the accuracy improvement on an independent calibration/validation set over the last iterations becomes insignificant).

Algorithm 7.1. summarizes the active learning approach for regression problems.

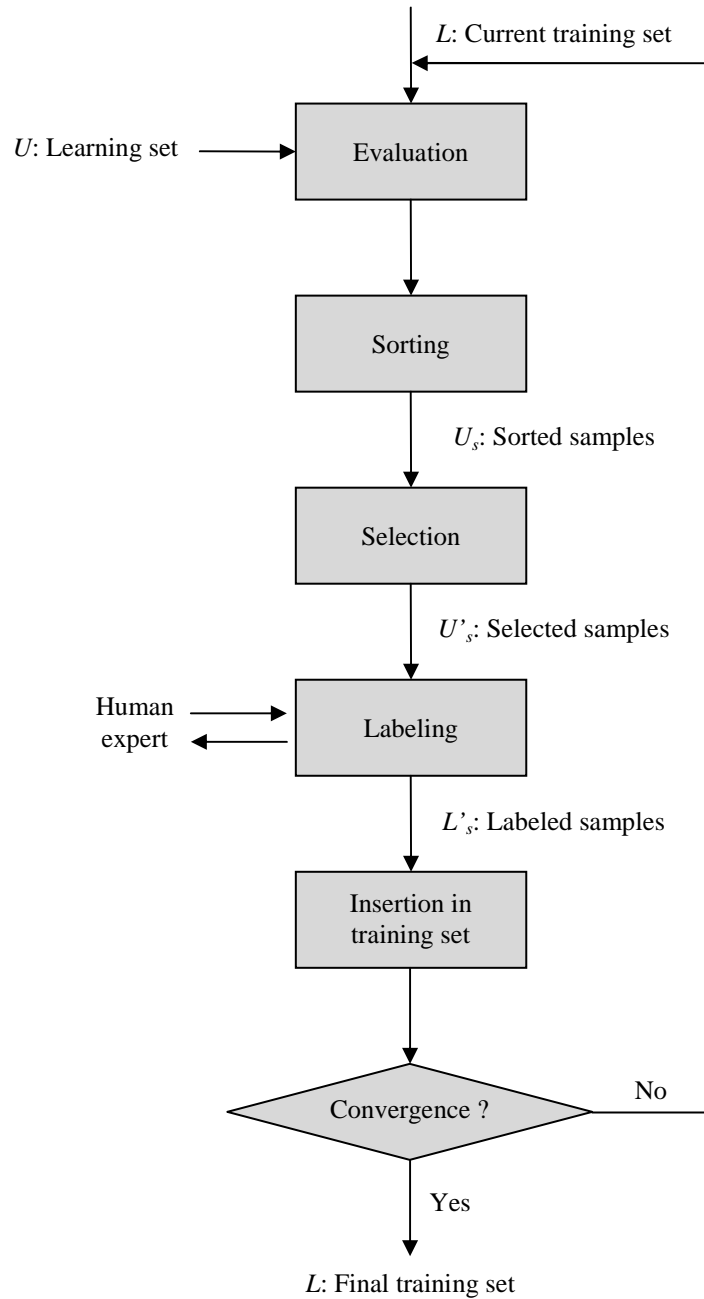


Fig. 7.1. Flow chart of the proposed active learning approach for regression problems.

**Algorithm 7.1.:** Active Learning Approach**Inputs:**

$L$ : initial training set, composed of  $n$  labeled samples.

$U$ : learning set, composed of  $m$  ( $m \gg n$ ) unlabeled samples.

$N_s$ : number of samples to add at each iteration of the active learning process.

**Output:**

$L$ : final training set.

**Repeat**

1. Considering the current training set  $L$ , evaluate each sample  $\mathbf{x}_j$  ( $j = n+1, n+2, \dots, n+m$ ) of the learning set  $U$  using the criterion  $h$ .
2. Sort the learning set  $U$  in function of the criterion  $h$  in order to obtain the set  $U_s$ .
3. Select the first  $N_s$  samples from  $U_s$ .

- 
4. Label the selected samples  $U'_s$ .
  5. Add the labeled samples  $L'_s$  to the training set  $L$  and remove them from  $U$ .
- Until** the predefined convergence condition is not satisfied.
- 

In the next subsections, we present the different active learning strategies proposed in this chapter. First, we focus on solutions for PLSR and then we consider SVM regression.

### 7.3.1. Active Learning Strategies for PLSR

#### 7.3.1.1. Distance from the Closest Training Sample

The first strategy, named TRd in the rest of the chapter, consists to calculate for each sample  $\mathbf{x}_j$  ( $j = n+1, n+2, \dots, n+m$ ) the Euclidean distances  $\mathbf{d}_j \in R^n = [d_{j,1}, d_{j,2}, \dots, d_{j,n}]$  in the feature domain from the samples  $\mathbf{x}_i$  ( $i = 1, 2, \dots, n$ ) already composing the current training set:

$$d_{j,i} = \|\mathbf{x}_j - \mathbf{x}_i\|. \quad (7.15)$$

After that, the closest training sample  $t_{MIN,j}$  is identified and the corresponding distance value  $d_{MIN,j}$  is considered as criterion. In this way, we select samples placed in areas of the feature space not covered by training samples and avoid to choose samples similar to those already present in the current training set.

Algorithm 7.2. synthesizes the proposed strategy based on the distance from the closest training sample.

---

**Algorithm 7.2.:** PLSR Active Learning based on Distance from the Closest Training Sample

---

1. Compute the Euclidean distances  $\mathbf{d}_j \in R^n = [d_{j,1}, d_{j,2}, \dots, d_{j,n}]$  from the  $n$  different training samples for each sample  $\mathbf{x}_j$  ( $j = n+1, n+2, \dots, n+m$ ) of the learning set  $U$ .
  2. Identify the training sample  $t_{MIN,j}$  closest to the sample.
  3. Consider the distance value  $d_{MIN,j}$  associated with the training sample  $t_{MIN,j}$ .
  4. Set  $h(j) = -d_{MIN,j}$ .
- 

#### 7.3.1.2. Pool of Regressors

The second strategy (PoR) is based on a pool of regressors. Considering the original training set  $L$ ,  $n_s$  training subsets are constructed by sampling  $L$  in the spectral domain. Each training subset is considered independently from each other and used to train a different regressor. In this way,  $n_s$  parallel regressors are designed. In particular, the target value  $\mu_{j,k}$  is predicted for each sample  $\mathbf{x}_j$  ( $j = n+1, n+2, \dots, n+m$ ) and for each regressor  $r_k$  ( $k=1, 2, \dots, n_s$ ). Therefore,  $n_s$  different estimations are obtained for each sample. Finally, the different estimations are combined opportunely by calculating the variance value on them:

$$h_{PoR,j} = \frac{1}{n_s} \sum_{k=1}^{n_s} (\mu_{j,k} - \bar{\mu}_j)^2 \quad (7.16)$$

where

$$\bar{\mu}_j = \frac{1}{n_s} \sum_{k=1}^{n_s} \mu_{j,k}. \quad (7.17)$$

The samples characterized by the greater disagreements between the different regressors, i.e. the greater values of variance, are selected. Indeed, a high disagreement means that the corresponding sample has been estimated with high uncertainty, and thus adding it to the training set could be useful to improve the regression process.

Algorithm 7.3. summarizes the proposed methodology based on the pool of regressors.

**Algorithm 7.3.:** PLSR Active Learning based on Pool of Regressors

1. Construct  $n_s$  different training subsets  $L_g$  ( $g=1, 2, \dots, n_s$ ) by sampling the spectral domain.
2. Predict the target value of each sample  $\mathbf{x}_j$  ( $j = n+1, n+2, \dots, n+m$ ) of the learning set  $U$  for each regressor  $r_k$  ( $k=1, 2, \dots, n_s$ ).
3. Compute the variance on the predictions  $h_{PoR,j}$  given by the different regressors using (7.15).
4. Set  $h(j) = -h_{PoR,j}$ .

**7.3.2. Active Learning Strategies for SVM Regression****7.3.2.1. Distance from the Support Vectors**

The second method (SVd) proposed for SVM regression is similar to TRd presented previously for PLSR. However, while for TRd we calculate the distances with respect to all training samples, in this case we consider only the training samples identified as SVs after the regressor training on  $L$ . This is motivated by the fact that while for PLSR all training samples contribute to describe the regression model, for SVM only the SVs are necessary to define the regression function. Moreover, more complex sorting and selection strategies are performed in order to take into account the sample distribution in the feature space. First, for each sample  $\mathbf{x}_j$  ( $j = n+1, n+2, \dots, n+m$ ) the index  $s_{MIN,j}$  of the closest support vector is identified and the corresponding distance value  $d_{MIN,j}$  is calculated. Then, for each sample  $\mathbf{x}_j$  ( $j = n+1, n+2, \dots, n+m$ ) we define as  $\tilde{\alpha}_j$  the absolute value of the Lagrange multiplier associated with the closest support vector. We recall that the Lagrange multipliers weigh each training sample according to its importance in determining the final solution. The most important training samples are those for which the corresponding Lagrange multipliers are in absolute terms equal to the regularization parameter  $C$ . At this point, the samples of the learning set  $U$  are ordered first in function of the value  $\tilde{\alpha}_j$  and then in function of the distance value  $d_{MIN,j}$ . The final selection is obtained from this sorted set after including an additional selection constraint. In particular, if the new sample to select shares the same closest support vector with a sample already selected at that iteration, it is discarded. In this way, we limit the selection of similar and redundant samples and select samples distributed as most as possible over the feature space.

Algorithm 7.4. synthesizes the proposed method based on the distance from the SVs.

**Algorithm 7.4.:** SVM Active Learning based on Distance from the Support Vectors

1. Identify the  $S_n$  support vectors of the regressor on the training set  $L$ .
2. Compute the Euclidean distances  $\mathbf{d}_j \in R^{S_n} = [d_{j,1}, d_{j,2}, \dots, d_{j,S_n}]$  from the  $S_n$  different support vectors for each sample  $\mathbf{x}_j$  ( $j = n+1, n+2, \dots, n+m$ ) of the learning set  $U$ .
3. Identify the support vector  $s_{MIN,j}$  closest to the sample.
4. Consider the distance value  $d_{MIN,j}$  associated with the support vector  $s_{MIN,j}$ .
5. Consider the absolute value  $\tilde{\alpha}_j$  of the Lagrange multiplier associated with the support vector  $s_{MIN,j}$ .
6. Set  $h_1(j) = -\tilde{\alpha}_j$ ,  $h_2(j) = -d_{MIN,j}$ .
7. Select first in function of  $h_1$  and then in function of  $h_2$ , but if the new sample to select shares the same closest support vector with a sample already selected at that iteration, it is discarded.

**7.3.2.2. Pool of Regressors**

The first strategy (PoR) is identical to that presented previously for PLSR. We refer the reader to Section 7.3.1.2. for more details.

TABLE 7.I  
DATASET INFORMATION AND EXPERIMENTAL SETUP FOR THE DIFFERENT DATA SETS

Dataset information				Experimental setup	
Name	# features	# learning samples	# test samples	# initial training samples	# samples added at each iteration
Diesel	401	133	112	33	20
Orange juice	700	149	67	49	20
Tecator	100	172	43	32	20

## 7.4. Experiments

### 7.4.1. Data Set Description and Experimental Setup

In order to validate the proposed active learning methods, we have conducted an experimental study on three real data sets.

The first data set refers to multispectral acquisitions of diesel fuels [21]. It was built by the Southwest Research Institute in order to develop instrumentation to evaluate fuel on battle fields. Along with the spectral acquisitions, different properties are available, such as boiling point at 50% recovery, cetane number, density, freezing temperature, total aromatics, viscosity. The data set contains only summer fuels, and outliers were removed. In our experiments, we consider one of the most difficult prediction tasks in this data set, i.e., the prediction of the cetane number of the fuel. All spectra range from 750 to 1550 nm, discretized into 401 wavelength values. The data set contains 20 high leverage spectra, shown in Fig. 7.2(a), and 225 low leverage spectra, the latter being separated into two subsets labeled a and b. As suggested by the providers of the data, we have built a learning set with the high leverage spectra and subset a of the low leverage spectra (thus yielding 133 spectra). The test set is made of the low leverage spectra of subset b (gathering 112 spectra).

The second data set deals with the problem of determining sugar (saccharose) concentration in orange juice samples by near-infrared reflectance spectroscopy [22]. The acquisitions consist of 700 spectral variables representing the absorbance ( $\log 1/R$ ) at different wavelengths between 1100 and 2500 nm. The absorbance is defined as  $\log (1/R)$ , where  $R$  is the light reflected by the sample surface. In this case, learning and test sets contain 149 and 67 samples, respectively. In Fig. 7.2(b), we show the near-infrared spectra of the orange juice learning set.

The last data set deals with the determination of the fat content in meat samples analyzed by near-infrared transmittance spectroscopy [23]. Each sample contains finely chopped pure meat with different moisture, fat and protein contents. Those contents, measured in percent, are determined by analytic chemistry. The spectra, acquired by the Tecator Infratec Food and Feed Analyzer, records light transmittance through the meat samples at 100 wavelengths in the range between 850 and 1050 nm. The corresponding 100 spectral variables are the absorbance defined by the measured transmittance values. The spectra are normalized according to the standard normal variance method, i.e., mean equal to zero and variance equal to one. For this data set, learning and test sets contain 172 and 43 spectra, respectively. The near-infrared spectra of the Tecator learning set is depicted in Fig. 7.2(c).

In all the following experiments, for all data sets, the initial training samples required by the active learning process were selected randomly from the learning set  $U$ . For the diesel data set, starting from 33 samples, the active learning algorithms were run until all the learning samples were added to the training set, adding 20 samples at each iteration. Similarly, 20 samples were added at each iteration by starting from 49 and 32 samples for the orange juice and Tecator data sets, respectively. The details of the experimental setup on the different data sets are summarized in Table 7.I. The entire active learning process was run ten times, each time with a different initial training set to yield statistically reliable results. At each run, the initial training samples were chosen in a completely random way.



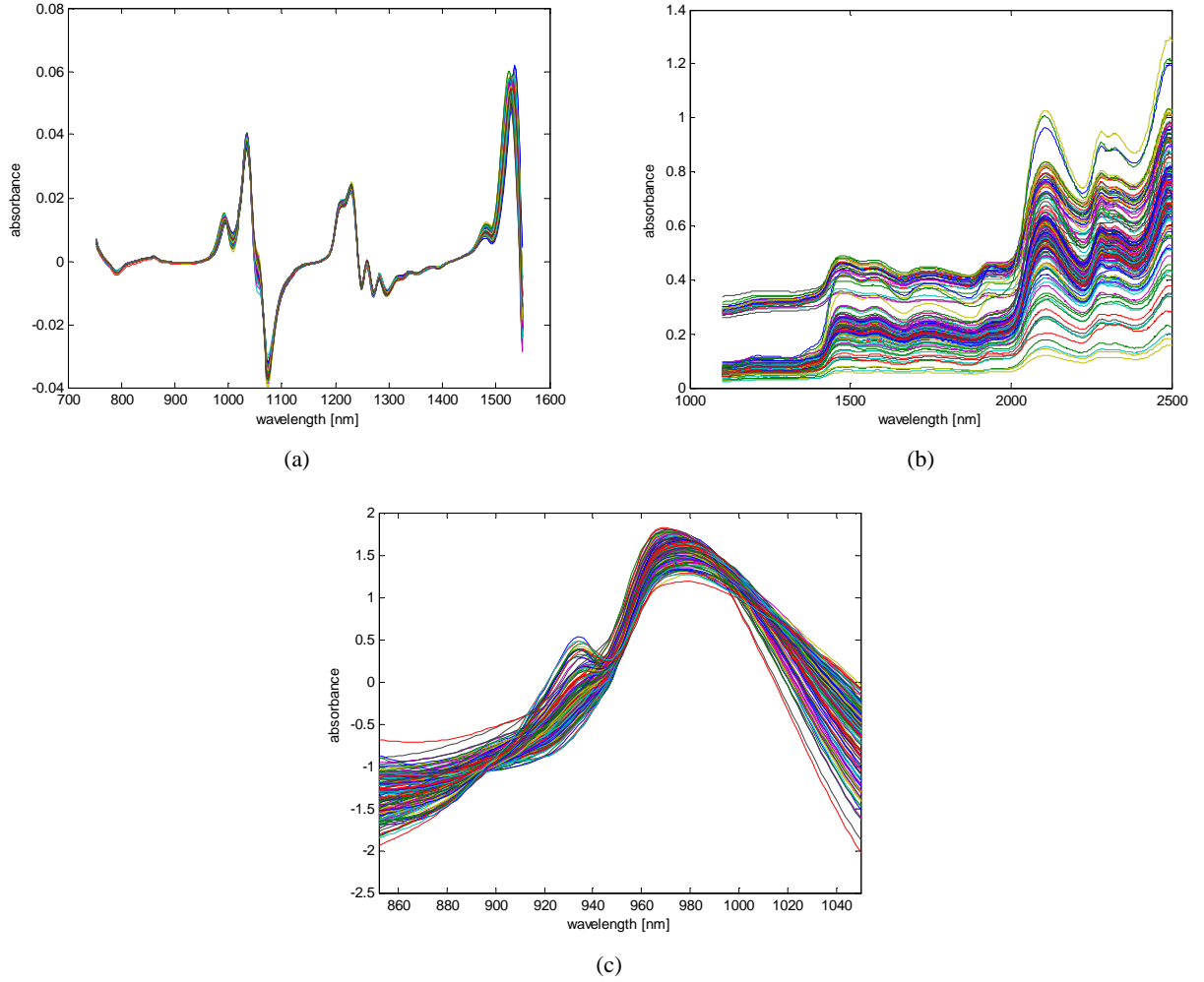


Fig. 7.2. Spectra of (a) the diesel, (b) the orange juice, and (c) the Tecator data sets.

PLSR and SVM regressors were also trained on the entire learning set in order to have a reference-training scenario, called "full" training. On the one hand, the regression results obtained in this way represent a lower bound for the errors. On the other hand, we expect that the upper error bound will be given by the completely random selection strategy. We recall that the purpose of any active learning strategy is to converge to the performance of the "full" training scenario faster than the random selection method.

Regression performances were evaluated on the test sets in terms of the standard error of estimate (EST)

$$EST = \sqrt{\frac{1}{t} \sum_{i=1}^t (\hat{y}_i - y_i)^2} \quad (7.18)$$

where  $t$  is the number of test samples.

Concerning the parameter setting, in the case of PLSR, the optimal number of latent variables was estimated by cross validation in the range  $[1, 20]$ . For SVM, we adopted a Gaussian kernel. This choice is motivated by the generally good prediction accuracy associated to this kernel. The regularization and kernel width parameters were tuned empirically through cross validation in the ranges  $[2^{-9}, 2^9]$  and  $[2^{-11}, 2^3]$ , respectively.

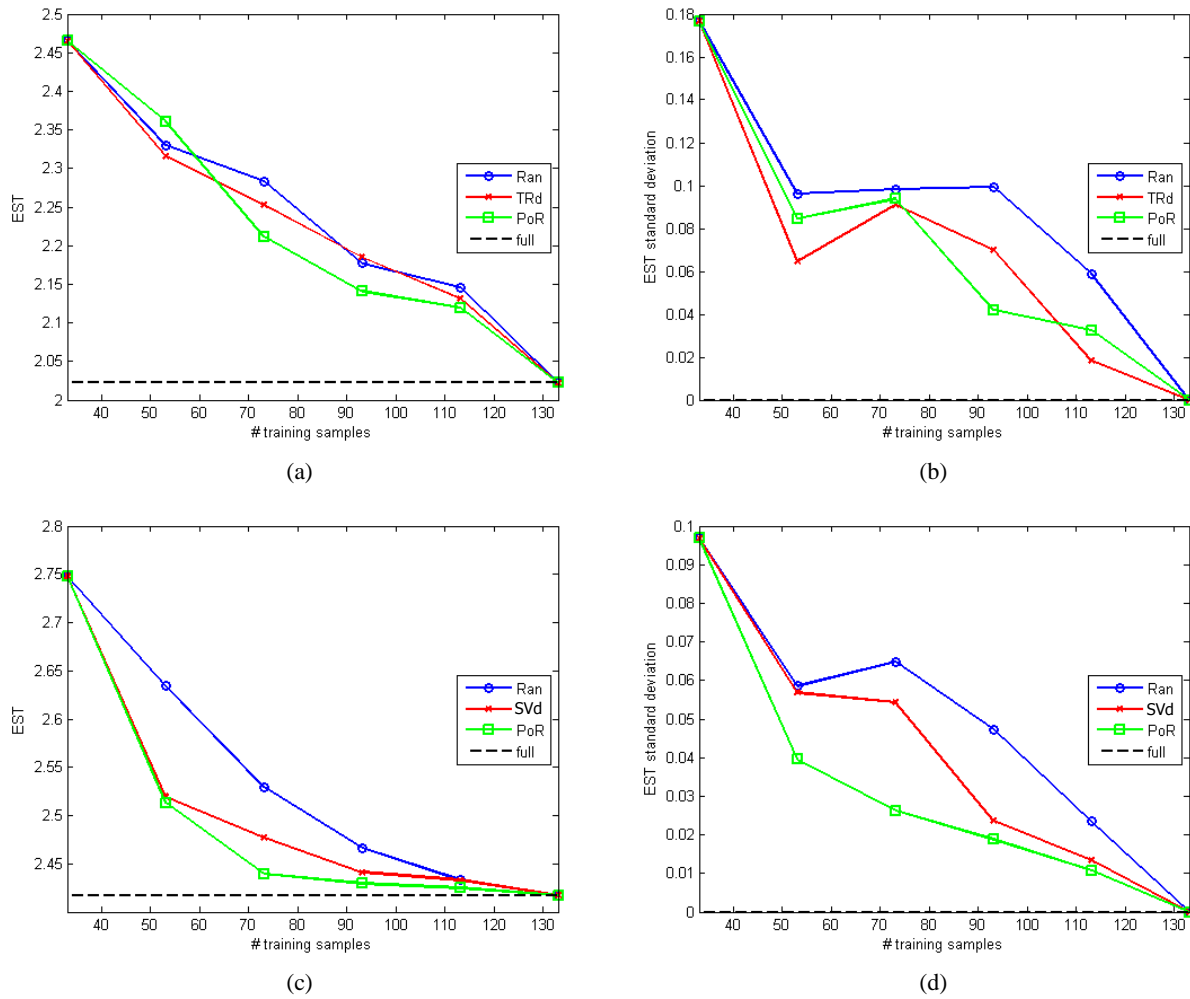


Fig. 7.3. Performances achieved on the diesel data set for (a), (b) PLSR and (c), (d) SVM in terms of (a), (c) EST, (b), (d) EST standard deviation.

## 7.4.2. Experimental Results

Figs. 7.3-7.5 report the results obtained for the diesel, orange juice, and Tecator data sets, respectively, by evolving the active learning process. In particular, the graphs refer to (a), (b) PLSR and (c), (d) SVM in terms of (a), (c) EST and (b), (d) EST standard deviation ( $\sigma_{\text{EST}}$ ). First, we note that at the starting points poor performances were obtained, both in terms of EST and related  $\sigma_{\text{EST}}$ . This result can be expected because of the small number of training samples used to train the regressors, which has also a direct impact on the regression model quality as shown by the strong variability in terms of  $\sigma_{\text{EST}}$  of the prediction errors. Another expected result is given by the improvement of performances when additional samples are inserted in the training set. This results in graphs with an approximately monotonous decreasing behavior of EST and  $\sigma_{\text{EST}}$ , which tend to converge to the results yielded by the “full” regressors, for which the entire learning set is exploited to train the model. Although such decreasing is verified for both active and random selection, we note that in general the active methods allow a faster convergence to the “full” result with respect to the random strategy, both in terms of EST and  $\sigma_{\text{EST}}$ . In particular, the improvements in terms of  $\sigma_{\text{EST}}$  indicate greater levels of stability in defining the regression model. While for the random selection the entire set of learning samples is necessary to converge for all experiments, in some cases the active learning process allows to converge completely using just a subset of the learning set. Moreover, before convergence, the proposed active learning strategies give in general an improvement with respect to the random one. This means that similar values of prediction errors can be obtained using a minor quantity of training samples,

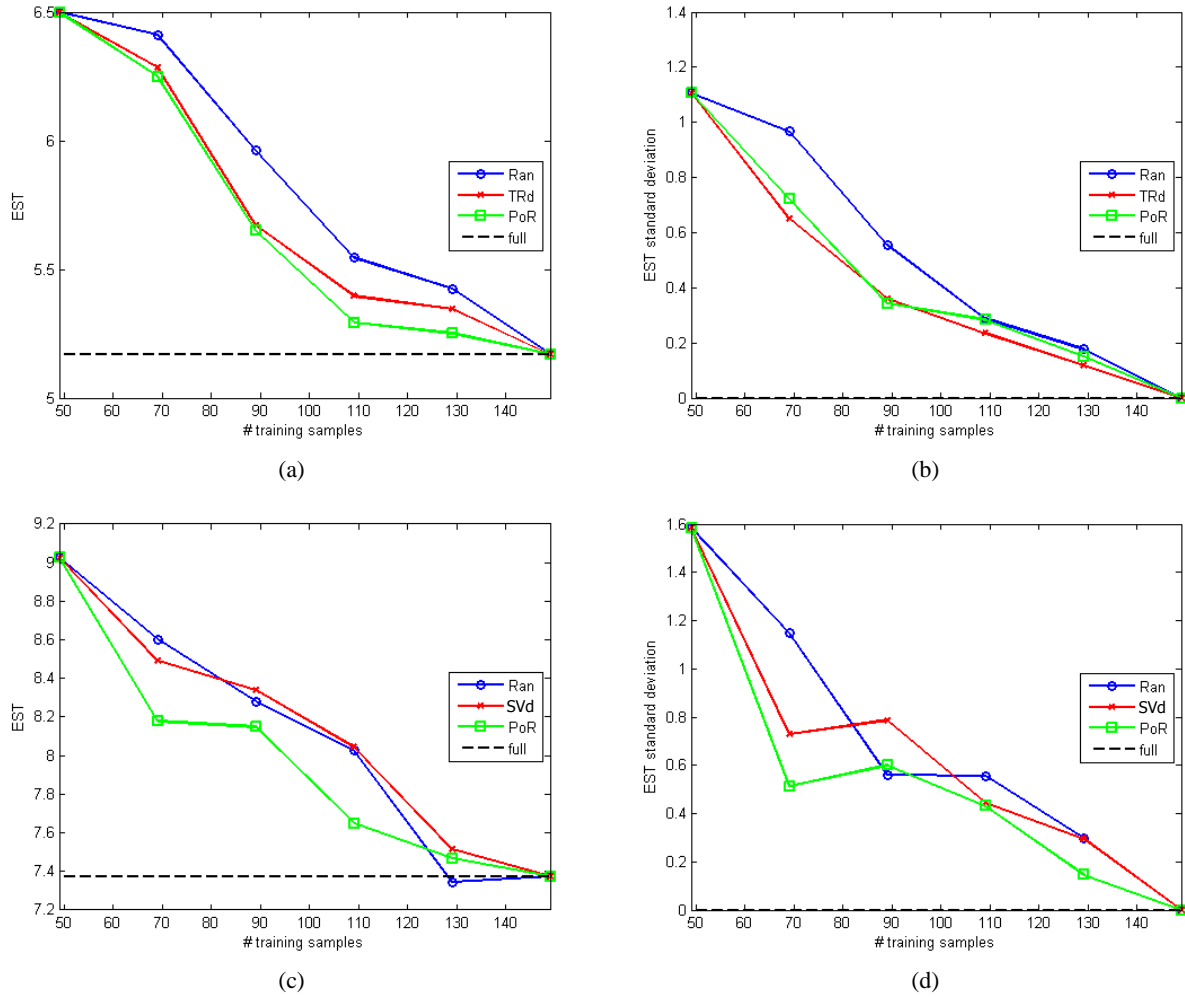


Fig. 7.4. Performances achieved on the orange juice data set for (a), (b) PLSR and (c), (d) SVM in terms of (a), (c) EST, (b), (d) EST standard deviation.

which implies a reduction of the expert work and a decreasing of the computational time necessary to train the regressor. Among the proposed strategies, the method PoR based on the pool of regressors yields in general better results with respect to those based on the distances in the feature space between labeled and unlabeled samples. This is verified for both PLSR and SVM.

The obtained results are shown in greater detail in Table 7.II(a)-(c), for the diesel, orange juice, and Tecator data sets, respectively. In particular, we considered the performances obtained when 40 additional samples are inserted in the training set. Therefore, number of training samples equal to 73, 89, 72 are considered for the different data sets, respectively. We report the values of EST and the corresponding  $\sigma_{EST}$ . The best results are highlighted in bold font. Moreover, for PLSR we indicate the number of optimal latent variables estimated automatically by cross validation, while for SVM we show the number of support vectors identified in the training process. The proposed strategies are characterized by better performances with respect to the random method from different points of view. First, better values of accuracies are obtained using the same number of training samples. Then, better values of standard deviations associated with the prediction errors are verified.

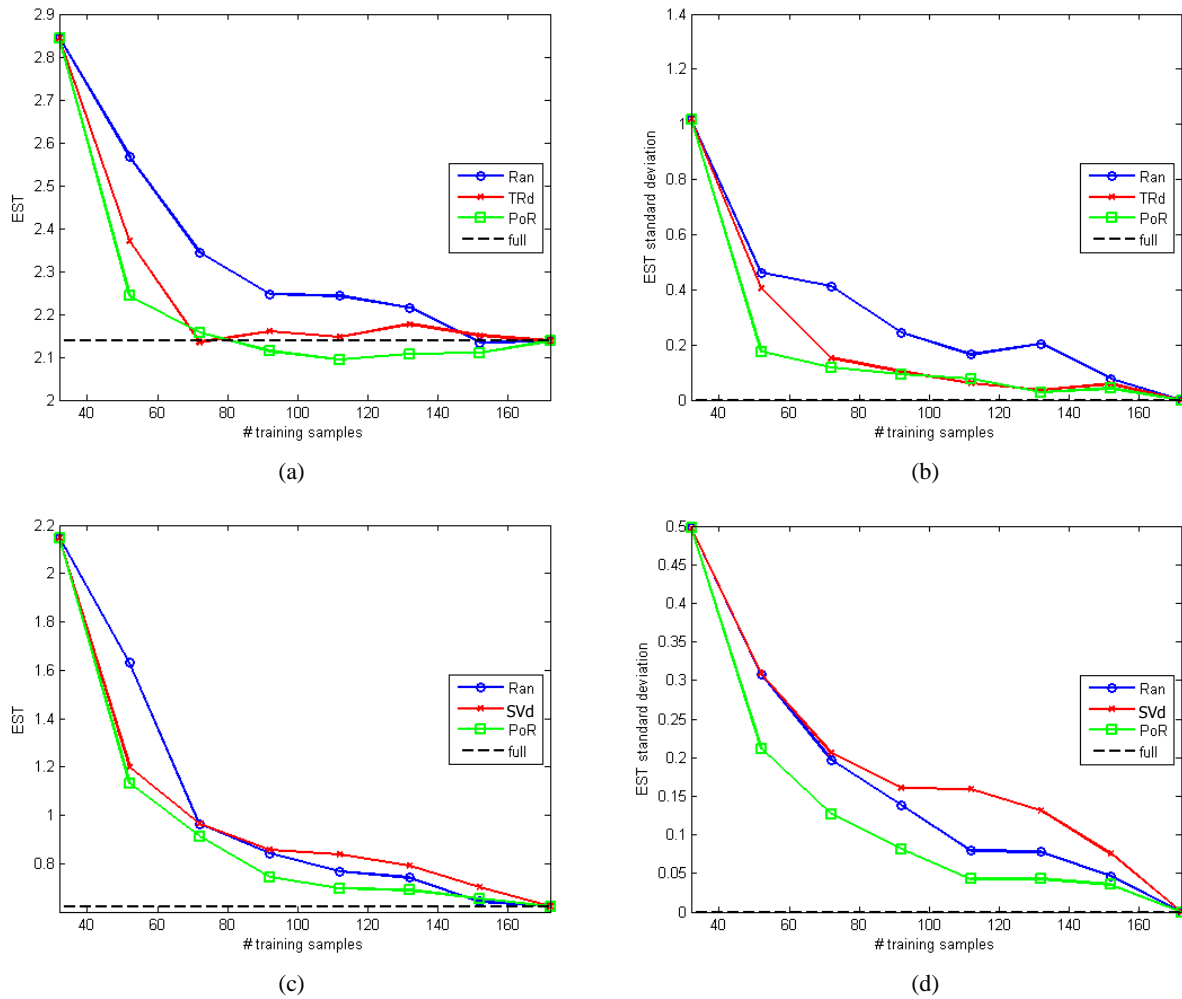


Fig. 7.5. Performances achieved on the Tecator data set for (a), (b) PLSR and (c), (d) SVM in terms of (a), (c) EST, (b), (d) EST standard deviation.

## 7.5. Conclusion

In this chapter, we have introduced the active learning approach to face the problem of training sample collection for regression problems related to the estimation of chemical concentrations from spectroscopic data. Starting from a small and suboptimal training set, an iterative process selects from a set of unlabeled data the samples considered more significant for the regression process, i.e., those able to give smaller prediction errors while minimizing the number of training samples and thus the expert efforts and costs for collecting the final training set. In particular, we have proposed some strategies specifically developed for two state-of-the-art regression approaches, namely PLSR and SVM. For PLSR, the first method is based on adding samples that are distant from the current training samples in the feature space, while the second one uses a pool of regressors in order to select the samples with the greater disagreements between the different regressors. For SVM, the method based on the pool of regressors and an additional strategy based on the selection of the samples distant from the current support vectors are proposed.

The experimental results on three different real data sets show good capabilities of the proposed strategies for selecting significant samples. In general, the proposed methods are characterized by higher performances in terms of both accuracy and stability with respect to a completely random selection strategy. Comparing them, the best active strategy appears the one based on the pool of regressors for both PLSR and SVM. It is however the most computational demanding since it needs the training of different regressors to build the pool.

TABLE 7.II  
EST, STANDARD DEVIATION ( $\sigma$ ), # LATENT VARIABLES, AND # SUPPORT VECTORS ACHIEVED FOR PLSR AND SVM ON  
(A) THE DIESEL, (B) THE ORANGE JUICE, AND (C) THE TECATOR DATA SETS

(a)

Method	# training samples	PLSR			SVM		
		EST	$\sigma_{EST}$	# latent variables	EST	$\sigma_{EST}$	# support vectors
Full	133	2.0222	0.0000	7.0	2.4171	0.0000	110.0
Initial	33	2.4651	0.1766	4.4	2.7478	0.0968	31.6
Ran	73	2.2835	0.0982	4.8	2.5296	0.0648	65.5
TRd/SVd		2.2528	<b>0.0911</b>	4.9	2.4768	0.0543	66.2
PoR		<b>2.2118</b>	0.0937	5.5	<b>2.4393</b>	<b>0.0262</b>	68.0

(b)

Method	# training samples	PLSR			SVM		
		EST	$\sigma_{EST}$	# latent variables	EST	$\sigma_{EST}$	# support vectors
Full	149	5.1688	0.0000	13.0	7.3729	0.0000	142.0
Initial	49	6.4987	1.1063	9.8	9.0254	1.5838	47.7
Ran	89	5.9608	0.5542	10.6	8.2810	<b>0.5606</b>	89.0
TRd/SVd		5.6693	0.3593	11.6	8.3395	0.7867	89.0
PoR		<b>5.6495</b>	<b>0.3431</b>	13.3	<b>8.1491</b>	0.5995	88.7

(c)

Method	# training samples	PLSR			SVM		
		EST	$\sigma_{EST}$	# latent variables	EST	$\sigma_{EST}$	# support vectors
Full	172	2.1377	0.0000	10.0	0.6214	0.0000	151.0
Initial	32	2.8436	1.0167	4.5	2.1485	0.4972	30.0
Ran	72	2.3443	0.4118	9.6	0.9646	0.1968	65.6
TRd/SVd		<b>2.1351</b>	0.1519	9.6	0.9649	0.2059	66.6
PoR		2.1572	<b>0.1186</b>	8.2	<b>0.9133</b>	<b>0.1271</b>	65.8

Though we focused on PLSR and SVM in this chapter, the active selection of the training samples could be used in combination with other supervised regression approaches. Moreover, while in this work the initial training set was chosen in a random way, more sophisticated initialization strategies could be envisioned in order to further improve the performances of the active learning process.

## 7.6. Acknowledgment

The authors would like to thank Prof. M. Meurens, Université Catholique de Louvain (Belgium), for providing the orange juice data set used in the experiments.

## 7.7. References cited in Chapter 7

- [1] S. Sekulic, H. W. Ward, D. Brannegan, E. Stanley, C. Evans, S. Sciavolino, P. Hailey, and P. Aldridge, "On-line monitoring of powder blend homogeneity by near-infrared spectroscopy," *Anal. Chem.*, vol. 68, no. 3, pp. 509–513, Feb. 1996.
- [2] M. Blanco, J. Coello, A. Eustaquio, H. Iturriaga, and S. MasPOCH, "Analytical control of pharmaceutical production steps by near infrared reflectance spectroscopy," *Anal. Chim. Acta*, vol. 392, no. 2-3, pp. 237–246, Jun. 1999.
- [3] Y. Ozaki, R. Cho, K. Ikegaya, S. Muraishi, and K. Kawachi, "Potential of near-infrared Fourier transform Raman spectroscopy in food analysis," *Appl. Spectrosc.*, vol. 46, no. 10, pp. 1503–1507, 1992.
- [4] M. Blanco, J. Coello, J. M. Garcia Fraga, H. Iturriaga, S. MasPOCH, and J. Pagès, "Determination of finishing oils in acrylic fibers by Near-Infrared Reflectance Spectroscopy," *Analyst*, vol. 122, pp. 777–781, 1999.

- [5] P. Geladi, "Some recent trends in the calibration literature," *Chemom. Intell. Lab. Syst.*, vol. 60, no. 1-2, pp. 211–224, Jan. 2002.
- [6] N. Benoudjit, F. Melgani, and H. Bouzgou, "Multiple regression systems for spectrophotometric data analysis," *Chemom. Intell. Lab. Syst.*, vol. 95, no. 2, pp. 144–149, Feb. 2009.
- [7] H. Li, Y. Liang, and Q. Xu, "Support vector machines and its applications in chemistry," *Chemom. Intell. Lab. Syst.*, vol. 95, no. 2, pp. 188–198, Feb. 2009.
- [8] B. Settles, "Active learning literature survey," *Tech. Rep.*, Univ. of Wisconsin-Madison, 2009.
- [9] P. Mitra, C. A. Murthy, and S. K. Pal, "A probabilistic active support vector learning algorithm," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 3, pp. 413–418, Mar. 2004.
- [10] E. Pasolli and F. Melgani, "Active learning methods for electrocardiographic signal classification," *IEEE Trans. Inform. Technol. Biomed.*, vol. 14, no. 6, pp. 1405–1416, Nov. 2010.
- [11] E. Pasolli, F. Melgani, and Y. Bazi, "Support vector machine active learning through significance space construction," *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 3, pp. 431–435, May 2011.
- [12] D. Cohn, Z. Ghahramani, and M. Jordan, "Active learning with statistical models," *J. Artif. Intell. Res.*, vol. 4, pp. 129–145, Mar. 1996.
- [13] K. Fukumizu, "Statistical active learning in multilayer perceptrons," *IEEE Trans. Neural Netw.*, vol. 11, no. 1, pp. 17–26, Jan. 2000.
- [14] R. Castro, R. Willett, and R. Nowak, "Faster rates in regression via active learning," *Adv. Neural Inf. Process. Syst.*, vol. 18, pp. 179–186, 2006.
- [15] M. Sugiyama, "Active learning in approximately linear regression based on conditional expectation of generalization error," *The Journal of Machine Learning Research*, vol. 7, pp. 141–166, Jan. 2006.
- [16] R. Burbidge, J. J. Rowland, and R. D. King, "Active learning for regression based on query by committee," *Intelligent Data Engineering and Automated Learning*, pp. 209–218, 2007.
- [17] M. Sugiyama and N. Rubens, "A batch ensemble approach to active learning with model selection," *Neural Networks*, vol. 21, no. 9, pp. 1278–1286, Nov. 2008.
- [18] M. Sugiyama and S. Nakajima, "Pool-based active learning in approximate linear regression," *Mach. Learn.*, vol. 75, no. 3, pp. 249–274, Jan. 2009.
- [19] J. Paisley, X. Liao, and L. Carin, "Active learning and basis selection for kernel-based linear models: a Bayesian perspective," *IEEE Trans. Signal Process.*, vol. 58, no. 5, pp. 2686–2700, May 2010.
- [20] S. Wold, H. Martens, and H. Wold, "The multivariate calibration problem in chemistry solved by the PLS method," *Matrix Pencils*, vol. 973, pp. 286–293, 1983.
- [21] Diesel data set available at <http://www.eigenvector.com/data/SWRI/>.
- [22] Orange juice data set available at <http://www.ucl.ac.be/mlg/>.
- [23] Tecator data set available at <http://lib.stat.cmu.edu/datasets/tecator>.

## 8. A Framework for Computer-Aided Ground-Truth Collection for Optical Image Classification

*Abstract* – Ground-truth design is a tricky problem and also critical since it has a direct impact on most of the subsequent image processing and analysis steps. In this chapter, a novel framework for assisting a human user in the design of a ground-truth for classifying a given optical remote sensing image is proposed. It is based on automatic unsupervised procedures of level set segmentation and clustering to make both spatial and spectral information contribute in the ground-truth design. In particular, it allows identifying the most significant areas of the image and facilitating the manual labeling operation. The resulting ground-truth is classifier-free and can be further improved by making it classifier-driven through an active learning process. Experimental results on very high spatial resolution and hyperspectral images show the usefulness and effectiveness of the proposed approach.

## 8.1. Introduction

In the remote sensing field, one of the most challenging problems is represented by the classification of images to create and update land-cover maps for different applications related to the monitoring of the Earth at local and global scales. From a methodological point of view, the classification process has been faced in the literature through two main approaches: unsupervised and supervised. In general, supervised methods have shown very promising performances, but they require a priori information about the considered classification task, and thus the intervention of human users. In the literature, most of the attention has been given on improving the accuracy of the classification process by acting mainly at the following three levels: 1) data representation; 2) discriminant function model; and 3) criterion on the basis of which the discriminant functions are optimized [1]. These works are based on an essential assumption that is the samples used to train the classifier are statistically representatives of the classification problem to solve. However, the process of collection of training samples is not trivial, because the human intervention is subject to errors and costs in terms of both time and money. Therefore, the quality and the quantity of such samples are very important, because they have a strong impact on the performances of the classifier.

Only in the last few years, in the literature there has been a growing interest in developing methods focused on the problem of the construction of the training sample set, also called ground-truth. In particular, the objective is to develop automatic strategies or semi-automatic procedures based on interactive processes with human users.

A first problem in ground-truth collection is given by the mislabeling issue due to errors in the process of sample labeling. Ground-truth collection can be done by following two main approaches: 1) in situ observation and 2) photo-interpretation [2]. Each of them has its own advantages and drawbacks, but both are subject to errors. In the first case, this may occur because of georeferencing problems, while in the second one, spectral mismatching errors by human users are the main source of problems. Since the presence of mislabeled training samples has a direct negative impact on the classification process, the development of automatic techniques for validating the collected samples is crucial. In the literature still few solutions for coping with this issue have been proposed [3]-[5]. They are based on two main approaches. The first one admits the presence of mislabeled samples, but aims at designing a classifier that is less influenced by this presence. The second one tends to identify and remove the mislabeled samples from the training set.

Another problem frequent in real application scenarios is represented by the scarcity of available training samples due to complexity and cost that characterize the ground-truth construction process. Accordingly, this constrains the classification process to be carried out with small numbers of training samples, thus leading to weak estimates of the classifier parameters and potentially high classification error rates, in particular if class distributions are overlapped. A possible solution to this issue consists to exploit the large number of unlabeled samples that are typically available at zero cost from the image under analysis. Indeed, the improvement of the classifier accuracy is obtained by combining automatically labeled and unlabeled samples. Methods dealing with this issue are termed as semisupervised methods, which are investigated in some recent works [6]-[9]. They are based either on inflation or transduction principles. The inflation principle relies on the idea of augmenting the original training set by exploiting a set of unlabeled samples, which covers a portion of the whole image to classify. For this purpose, the labels of the unlabeled samples need to be beforehand estimated. The transduction principle is conceptually completely different from the inflation one. This is due to two main reasons: 1) all samples of the image and not just part of it contribute to the learning process and 2) training and classification steps are fused into a unique step.

Focusing on the photointerpretation approach, a common procedure to design the ground-truth consists: 1) to select randomly single pixels or to define regions of interest (ROIs) and then 2) to label them. However, this approach depends strongly on the expertise of the human users and tends to select samples that are redundant for the process of classification. For these reasons, there has been a growing interest in



developing strategies for the semi-automatic selection of training samples in order to minimize the number of interactions with the human user and maintain high performance in terms of between-class discrimination. In this context, active learning represents an interesting solution. Starting from a small and suboptimal initial training set, it consists to select few additional samples from a large amount of unlabeled data (learning set) through an iterative process. The aim of active learning is to rank the learning set according to an opportune criterion that allows to select the most useful samples to improve the model. In the last few years, different solutions have been proposed and applied successfully in different research areas [10], [11] and in different remote sensing application fields, such as detection of buried objects [12], classification of hyperspectral images [13], [14], and classification of very high spatial resolution images [15], [16]. Finally, active learning approach has been proposed very recently to adapt classification models to new images [17]. Despite the promising performances obtained, active learning methods present some drawbacks critical in real applications: 1) the iterative process on which they are based on is time-consuming, limiting possibilities to interact in real-time with the human user; 2) the availability of an initial training set is supposed and can strongly influence the performance of the iterative active learning process; 3) the pixel-based labeling is awkward and difficult, since selected samples are generally placed on spatial boundaries between different classes.

The objective of this chapter is to propose an innovative framework for the design of the ground-truth that approaches this problem from a novel point of view with respect to what the literature presents. The proposed approach is (almost) completely automatic and comprehensive since it aims at assisting the human user from the first to the last steps of the design and in which active learning is just part of the framework. From a methodological point of view, the proposed strategy includes unsupervised procedures based on segmentation and clustering methods. In this way, both spatial and spectral information are considered in the process of ground-truth design. For this purpose, a new method of segmentation based on level sets is also introduced. To investigate the performance of the proposed approach, we conducted an experimental study based on two real images. In particular, we considered very high resolution (VHR) and hyperspectral images acquired by the IKONOS and the ROSIS sensors, respectively. The obtained results show promising capabilities of the proposed framework in terms of ground-truth design.

The remaining part of the chapter is organized as follows. In the next section, we present the proposed ground-truth design approach. Section 8.3. discusses experimental results obtained on two real remote sensing data sets. Finally, conclusions are drawn in Section 8.4.

## 8.2. Proposed Framework

The flow chart of the proposed ground-truth design framework is illustrated in Fig. 8.1. It is composed of different steps that can be summarized by Algorithm 8.1

---

### Algorithm 8.1.: Proposed Ground-Truth Design Framework

---

#### Inputs:

$X$ : original remote sensing image.

---

#### Output:

Ground-truth map.

---

1. Segment  $X$  by means of the hierarchical level set segmentation algorithm in order to obtain the segmented image  $X_s$ .
2. Select from the segmented image  $X_s$  the most representative segments  $S_s$ .
3. Label the selected segments  $S_s$ .
4. Sub-sample the labeled segments  $S'_s$  in order to obtain the classifier-free ground-truth.

#### Repeat

---

5. In case a classifier-driven ground-truth is desired, apply active learning.

**Until** the predefined convergence condition is not satisfied.

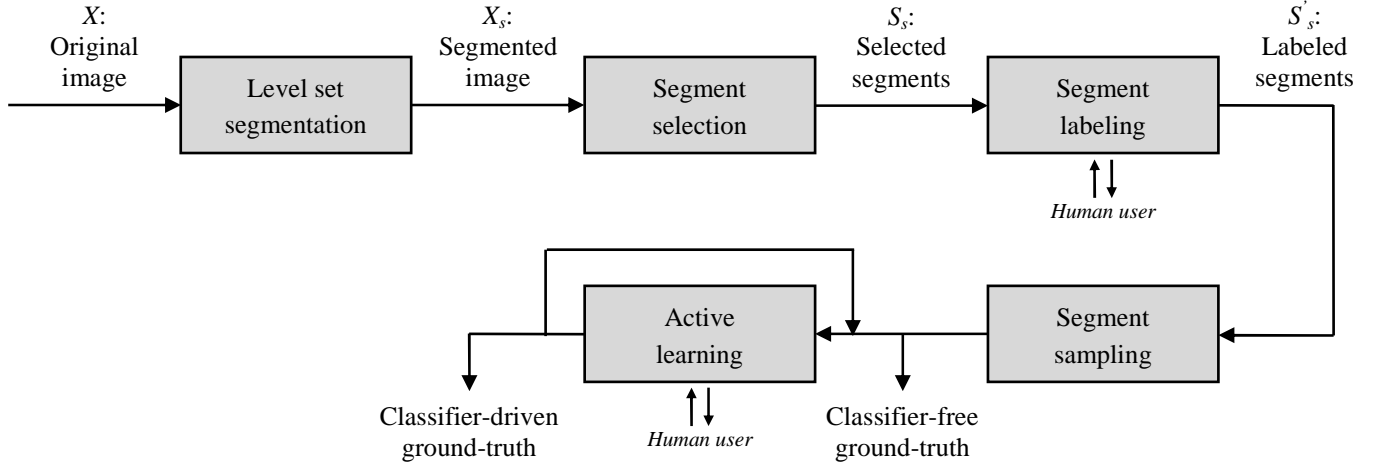


Fig. 8.1. Flow chart of the proposed ground-truth assisted design framework.

In the next subsections, we describe each step in more detail.

### 8.2.1. Level Set Segmentation

Given a remote sensing image  $X$ , the aim is to generate a corresponding segmented map  $X_s$ . We solve this issue of segmentation by means of a new level set method because methodologies currently proposed in the literature typically present two main drawbacks: 1) high computational burden; and 2) high number of free parameters to set.

A well-known segmentation model is the one proposed by Mumford and Shah [18] which aims at finding a contour  $C$  in order to segment  $X$  into nonoverlapping regions. The related energy functional is given by

$$F^{MS}(I, C) = \int_{\Omega} |X - I|^2 dx dy + \lambda \int_{\Omega \setminus C} |\nabla I|^2 dx dy + \mu |C| \quad (8.1)$$

where  $\Omega$  is the image domain,  $|C|$  is the length of the contour  $C$ , and  $\lambda$  and  $\mu$  are positive parameters. The minimization of the Mumford-Shah function results in an optimal contour that segments the image  $X$ , in addition to an image  $I$  formed from smooth regions within each of the connected components in the image domain separated by the optimal contour  $C$ .

The minimization of the aforementioned function is difficult in practice as it is a nonconvex problem. A possible solution is to consider the case where the image  $I$  in the functional (8.1) is a piecewise constant function [19]. In this case, the energy functional is given by

$$F^{MS}(C, c_1, c_2) = \int_{inside(C)} |X - c_1|^2 dx dy + \int_{outside(C)} |X - c_2|^2 dx dy + \mu |C| \quad (8.2)$$

and the related minimization problem is given as follows:

$$\min_{(C, c_1, c_2)} F^{MS}(C, c_1, c_2). \quad (8.3)$$

The functional (8.2) is known as the piecewise constant Mumford-Shah segmentation model or simply the Chan-Vese model [19]. We call the first two terms in (8.2) the global fitting energy, whereas the last term is a regularizing term that depends on the length of the curve. The parameter  $\mu$  controls the tradeoff between the goodness of fit and length of the curve  $C$ . The two constants  $c_1$  and  $c_2$  approximate the image  $X$  in the

segments inside( $C$ ) and outside( $C$ ), respectively. For a fixed contour  $C$ , the optimal values for these parameters are given by the average of  $X$  over the regions inside( $C$ ) and outside( $C$ ), respectively.

The minimization of the functional (8.2) can be performed within a level set framework. The framework introduced by Osher and Sethian became a popular tool in the field of image processing and computer vision [20]. The main idea behind the level set formulation is to represent the curve  $C$  by the zero level set of a Lipschitz function  $\phi : \Omega \rightarrow \mathfrak{R}$  such that

$$\begin{cases} C = \partial\omega = \{(x, y) \in \Omega : \phi(x, y) = 0\} \\ \text{inside}(C) = \omega = \{(x, y) \in \Omega : \phi(x, y) > 0\} \\ \text{outside}(C) = \Omega \setminus \omega = \{(x, y) \in \Omega : \phi(x, y) < 0\} \end{cases} . \quad (8.4)$$

The level set function is typically defined as the signed distance function of spatial points defined on  $\Omega$  to the curve  $C$ . By replacing the unknown variable  $C$  in (8.2) by the level set function  $\phi$ , the energy functional becomes [19]

$$\begin{aligned} F^{MS}(\phi, c_1, c_2) &= \int_{\Omega} |X - c_1|^2 H(\phi(x, y)) dx dy \\ &+ \int_{\Omega} |X - c_2|^2 (1 - H(\phi(x, y))) dx dy . \\ &+ \mu \int_{\Omega} \delta_0(\phi(x, y)) |\nabla \phi(x, y)| dx dy \end{aligned} \quad (8.5)$$

where  $H(z)$  is the Heaviside step function, i.e.,  $H(z)=1$  if  $z \geq 0$  and  $H(z)=0$  if otherwise.  $\delta_0(z)$  is the Dirac delta function  $\delta_0(z)=(d/dz)H(z)$ . In practice, these functions are replaced by the following regularized versions:

$$H_{\varepsilon}(z) = \frac{1}{2} \left( 1 + \frac{2}{\pi} \arctan\left(\frac{z}{\varepsilon}\right) \right) \quad (8.6)$$

$$\delta_{\varepsilon}(z) = \frac{d}{dz} H_{\varepsilon}(z). \quad (8.7)$$

The main advantage of such transformation is that the minimization of the functional (8.5) with respect to the level set function  $\phi$  can be handled more easily than the minimization of the functional (8.2) with respect to  $C$ . In addition the splitting and merging of the curve can be carried out by simply moving up and down the level set function  $\phi$ . To optimize  $F^{MS}(\phi, c_1, c_2)$  with respect to  $\phi$  as well as  $c_1$  and  $c_2$ , the two-step alternating approach proposed in [19] is iterated until convergence is reached.

In a first step,  $\phi$  is kept fixed and the functional is minimized with respect to  $c_1$  and  $c_2$ . The optimal values for these parameters are given by

$$c_1(\phi) = \frac{\int_{\Omega} X(x, y) H(\phi(x, y)) dx dy}{\int_{\Omega} H(\phi(x, y)) dx dy} \quad (8.8)$$

$$c_2(\phi) = \frac{\int_{\Omega} X(x, y) (1 - H(\phi(x, y))) dx dy}{\int_{\Omega} (1 - H(\phi(x, y))) dx dy} . \quad (8.9)$$

This simply means that  $c_1$  and  $c_2$  are the averages of  $X$  in  $\phi > 0$  and  $\phi \leq 0$ , respectively.

In a second step,  $c_1$  and  $c_2$  are, in turn, kept fixed and the energy functional is minimized with respect to  $\phi$ . The associated Euler-Lagrange equation is given by the following partial differential equation:

$$\frac{d\phi}{dt} = \delta_{\varepsilon}(\phi) \left[ \mu \operatorname{div} \left( \frac{\nabla \phi}{|\nabla \phi|} \right) - (X - c_1)^2 + (X - c_2)^2 \right]. \quad (8.10)$$

The solution of such evolution equation in  $\phi$  is made using finite differences. We refer the reader to [19] for the detailed numerical implementation. The segmentation procedure is summarized in Algorithm 8.2

**Algorithm 8.2.:** Chan-Vese Algorithm

---

1. Set  $k=0$  and initialize  $\phi$ , with  $\phi^k$  defined as the distance function from an initial curve  $C$ .

**Repeat**

2. For  $\phi=\phi^k$ , compute  $c_1^k$  and  $c_2^k$  as the averages of  $X$  in  $\phi > 0$  and  $\phi \leq 0$ , respectively.

3. Compute  $\phi^{k+1}$  by solving the following:

$$\phi^{k+1} = \phi^k - \Delta t \frac{\partial F^{MS}}{\partial \phi}(\phi^{k-1}, c_1^k, c_2^k). \quad (8.11)$$

where  $\Delta t$  is the time step.

4. Reinitialize  $\phi$  locally to the signed distance function to the curve.

**Until** the predefined convergence condition is not satisfied.

---

The formulation described above is of binary type and thus can be used only for connected problems, such as change detection problems [21]. The more general form is the multiphase segmentation for which a solution could be the multiphase level set implementation. This last is unfortunately computationally onerous. As an alternative, in this work, we propose a hierarchical binary implementation, summarized by Algorithm 8.3

**Algorithm 8.3.:** Hierarchical Level Set Segmentation**Input:**

$X$ : original remote sensing image.

**Output:**

$X_s$ : segmented image.

---

1. Run the binary level set algorithm on the original image  $X$  by setting  $\mu = \mu_1$  in order to obtain a root segmentation map  $X'_s$ .

2. Within each segment of  $X'_s$ , run again the algorithm on (masked)  $X$  by setting this time  $\mu = \mu_2$  (with  $\mu_2 < \mu_1$  to capture finer segments). The final result is a segmented image  $X_s$ .

---

An example of hierarchical level set segmentation result is shown in Fig. 8.2. In Fig. 8.2(a), we highlight in red a segment detected at the first iteration of the segmentation process, for which a problem of undersegmentation can be observed. In particular, the segment is large and includes different thematic classes, such as trees and grass. After the second step of the hierarchical segmentation, the segment is subdivided in several smaller segments (see Fig. 8.2(b)), which are less affected by the undersegmentation problem.

**8.2.2. Segment Selection**

In the second step of the framework, the task is to select from the segmented image  $X_s$  in an unsupervised way the  $\#_{int}$  most representative segments  $S_s$ , where  $\#_{int}$  is the desired number of interactions with the human user, i.e., the number of segments to label. For this purpose, three strategies are proposed.

The first ground-truth design strategy (Design-R) consists to select the segments in a completely random way, but excluding a priori the small segments with a number of pixels less than a threshold  $\#_{pix}$  fixed by the human user in order to reduce the impact of segmentation noise. The  $\#_{pix}$  is set according to the image resolution and the expected minimum object size in the considered image.

The second strategy (Design-M) selects the segments characterized by the maximum sizes in terms of number of pixels. In this way, big and homogeneous areas in the image are favored.

For the last strategy (Design-MC), we first represent each segment through its mean vector in the feature space. Then, just the segments with size greater than  $\#_{pix}$  are exploited and subdivided in  $\#_{int}$  different

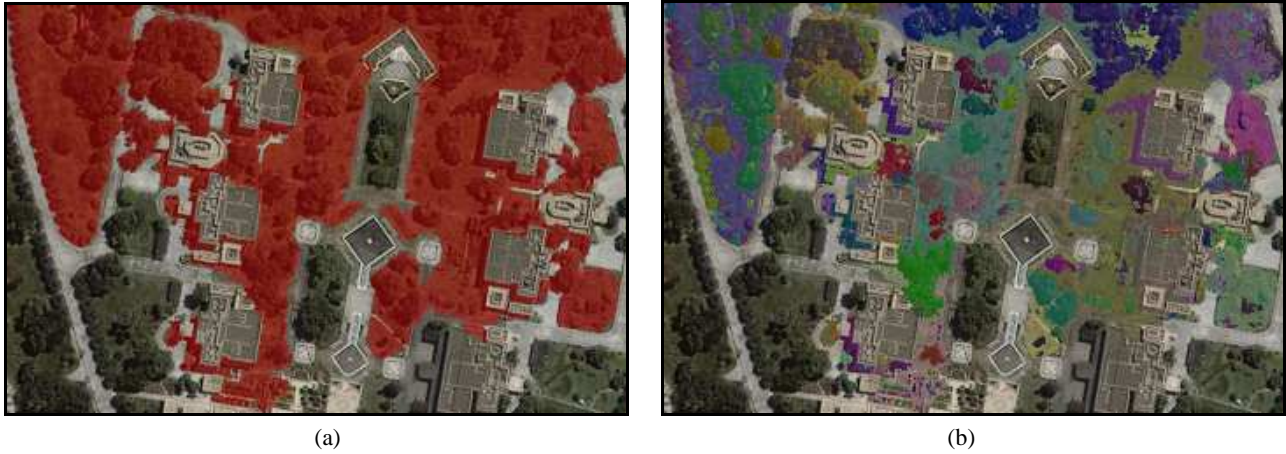


Fig. 8.2. Example of hierarchical level set segmentation. Each segment detected at the first iteration (a) is segmented another time (b).

groups through a clustering method. In the literature, several clustering techniques have been proposed [22]. In this work, in order to limit drastically the computational burden, we will opt for the simple  $K$ -means algorithm with  $K$  equal to the number of segment to select, i.e.,  $\#_{int}$ . Finally, from each cluster the segment with the maximum size is selected. In this way, big segments and exhibiting diversity in the feature domain are selected.

We note that all the proposed strategies select the segments in a single iteration, thus minimizing the computational time, and do not require any initialization phase.

Algorithm 8.4. resumes the three different strategies of segment selection proposed in this chapter.

---

**Algorithm 8.4.:** Segment Selection

---

**Input:**

$X_s$ : segmented image.

---

**Output:**

$S_s$ : selected segments.

---

1. Exclude the segments with a number of pixels less than the threshold  $\#_{pix}$ .

**Repeat**

[2. Design-R] Choose one segment in a completely random way.

[2. Design-M] Choose the segment with the greater size.

**Until** a number of segments equal to  $\#_{int}$  is selected.

[2. Design-MC] Cluster the segments in  $\#_{int}$  groups with the  $K$ -means algorithm, setting  $K=\#_{int}$ .

[3. Design-MC] Choose the segment with the greater size from each cluster.

---

### 8.2.3. Segment Labeling and Sampling

After selecting segments in an automatic and unsupervised way, in the successive step the human user has to interact with the system in order to label the selected segments. We observe that in the proposed framework the base element is not represented by single pixels, but by segments. In this way the process of labeling is facilitated with respect to the pixel-based approach, in particular when pixels are located on spatial boundaries between different classes.

Finally, the labeled segments  $S'_s$  may be sub-sampled in order to reduce the number of samples to train the classifier. For instance, if one desires to have a number of training samples equal to  $\#_{tr}$ , where  $\#_{tr} > \#_{int}$ , we may extract randomly from each segment a number of pixels proportional to its size.

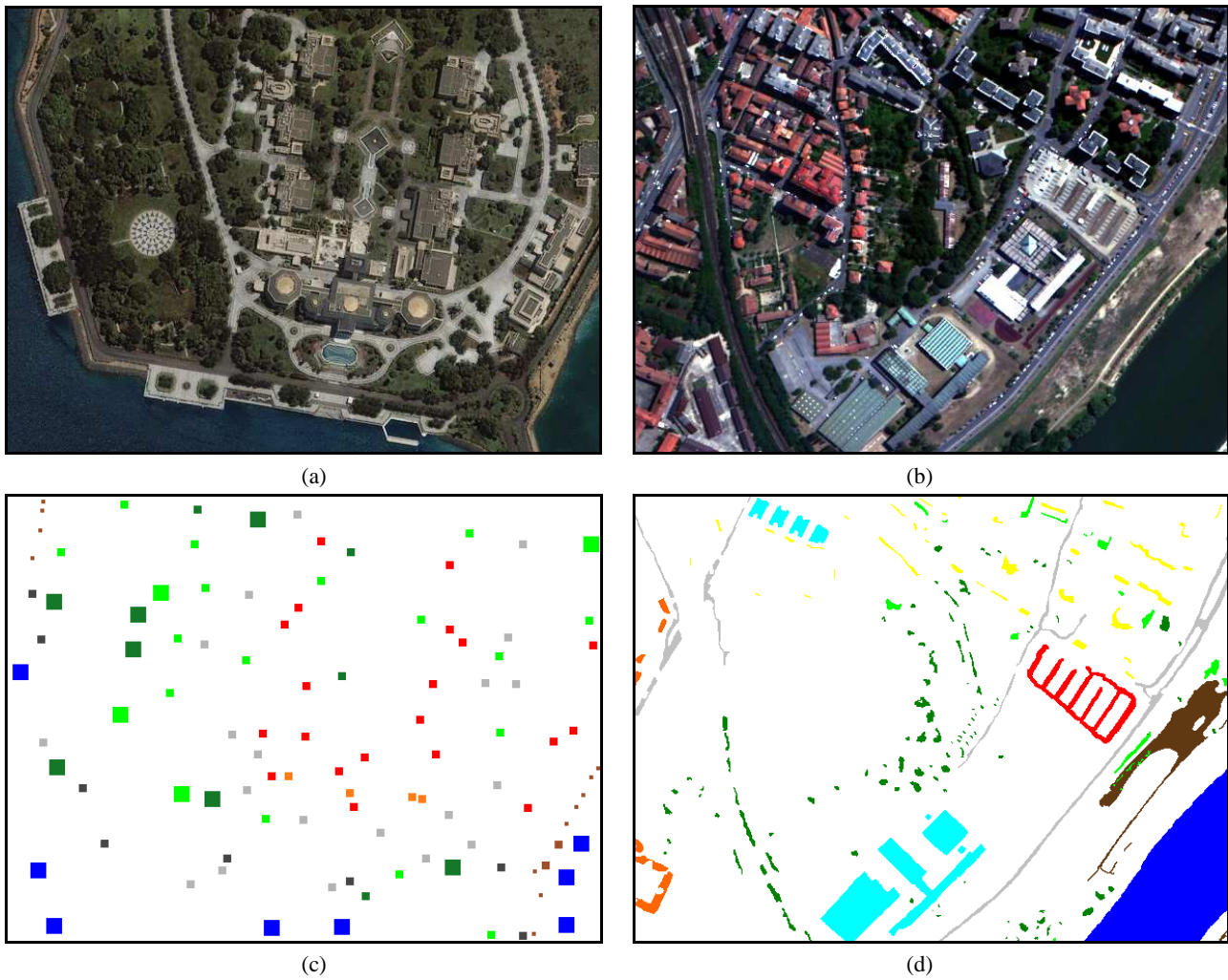


Fig. 8.3. Data sets used for the experiments. (a) RGB image for the Jeddah and (b) False-color image for the Pavia data sets. Test set for (c) the Jeddah and (d) the Pavia data sets.

TABLE 8.1  
CHARACTERISTICS OF THE IMAGES USED FOR THE EXPERIMENTS

Site information		Image information		
Location	Dimension [pixels]	Sensor	Acquisition date	Spatial resolution [m]
Jeddah (Saudi Arabia)	600x450	IKONOS	July, 2004	1.0
Pavia (Italy)	600x450	ROSIS	July, 2002	1.3

## 8.3. Experimental Results









### 8.3.1. Data Set Description and Experimental Setup

In order to validate the proposed ground-truth design framework, we conducted an experimental phase based on two real remote sensing images (Table 8.1).










The first data set represents a multispectral VHR image acquired by the IKONOS sensor in July 2004 (Fig. 8.3(a)). The image has three spectral bands with a spatial resolution of 1 m and refers to a portion of the city of Jeddah (Saudi Arabia), in which eight land cover types are dominant: two types of *Asphalt*, *Bare soil*, *Grass*, two types of *Roofs*, *Trees*, and *Water*.



TABLE 8.II  
NUMBER OF TEST SAMPLES FOR (a) THE JEDDAH AND (b) THE PAVIA DATA SETS

(a)		
Class	# test samples	
	Asphalt 1	512
	Asphalt 2	1280
	Bare soil	320
	Grass	2048
	Roofs 1	256
	Roofs 2	1280
	Trees	2048
	Water	2048
	<b>Total</b>	<b>9792</b>

(b)		
Class	# test samples	
	Asphalt	4372
	Bare soil	3840
	Bitumen	7277
	Bricks	2140
	Meadows	1029
	Shadow	1766
	Tiles	1064
	Trees	2768
	Water	12997
	<b>Total</b>	<b>37253</b>

The second data set is a hyperspectral image characterized by 102 bands and acquired by the Reflective Optics System Imaging Spectrometer (ROSIS) sensor over a part of the city of Pavia (Italy) in July 2002 (Fig. 8.3(b)). The spatial resolution is equal to 1.3 m. Nine classes were considered, namely: *Asphalt*, *Bare soil*, *Bitumen*, *Bricks*, *Meadows*, *Shadow*, *Tiles*, *Trees*, *Water*.

Both images have a dimension equal to 600x450 pixels. The proposed framework was executed on the whole image composed of 270,000 pixels. The ground-truths generated by the different strategies were used to train a supervised classifier based on support vector machines (SVMs) [23], which has shown especially effective for the classification of remote sensing images [24], [25]. Performances were evaluated on a test set composed of 9,792 and 37,253 samples for the Jeddah and the Pavia data sets, respectively. The available respective ground-truth maps are illustrated in Fig. 8.3(c) and Fig. 8.3(d), while the detailed numbers of test samples are reported in Table 8.II. The performance comparisons were done in terms of several measures which are: 1) the overall accuracy (OA), which is the percentage of correctly classified samples among all the considered samples, independently of the classes they belong to; 2) the Kappa statistic [26]; 3) the average accuracy (AA), which is the average over the classification accuracies obtained for the different classes; 4) the standard deviations ( $\sigma$ ) of OA, Kappa index, and AA, obtained by running ten times all experiments, in order to evaluate stability of the approaches; 5) the probability of detection of the thematic classes (PD) in order to evaluate the automatic class detection capabilities of the proposed approach. PD is defined as the ratio of the number of detected classes to the total number of classes.

For the sake of comparison, we considered the traditional ground-truth generation by photo-interpretation. In particular, ten ground-truths consisting of square ROIs were generated by ten different photo-interpreters so that to account for variable experience levels. In this case, all available classes were a priori included in the ground-truth, and accordingly PD=100%. Finally, additional ground-truths were collected by following pixel-based approaches: 1) selection of pixels in a completely random way (Pix-R); 2) clustering of pixels in #int groups and random selection of one pixel from each cluster (Pix-RC); 3) selection of pixels by active learning (Pix-AL). In particular, we adopted the state-of-the-art margin sampling

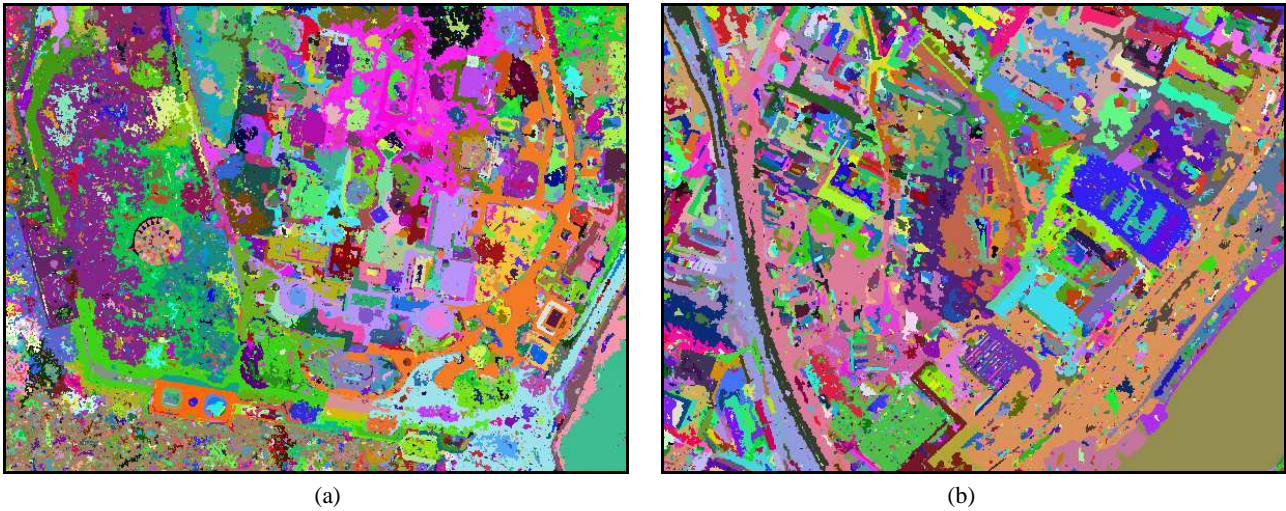


Fig. 8.4. Segmented images obtained using the hierarchical level set segmentation algorithm for (a) the Jeddah and (b) the Pavia data sets.

algorithm [27], which has shown good performances in the remote sensing field [28]. Also, in this case, all classes were a priori included in the initial training set (PD=100%).

### 8.3.2. Experimental Results

Considering the Jeddah data set, the result of the hierarchical level set segmentation algorithm applied on the first principal component of the image is shown in Fig. 8.4(a), in which each segment is represented with an arbitrary color. The segmentation result appears in general satisfactory, although the algorithm tends to oversegment the image. We note that, in our context (i.e., in ground-truth collection), oversegmentation is preferred to undersegmentation, because we desire that all pixels of a each segment belong to the same class.

The first set of experiments has the purpose to evaluate how the process of segmentation is helpful for incrementing classification performance without increasing the  $\#_{int}$  with the human user. Results obtained with  $\#_{int}$  equal to 8 are shown in Fig. 8.5(a),(c),(e) and Table 8.III(a). In particular, the best performances are highlighted in bold font. For the pixel-based approaches (Pix-R, Pix-RC, Pix-AL), the  $\#_{int}$  coincides with the number of selected samples. In this situation, very poor performances in terms of OA, Kappa index and AA were obtained. By contrast, additional samples can be extracted with zero labeling cost from the selected segments by the proposed methods (Design-R, Design-M, Design-MC). From Fig. 8.5(a),(c),(e), it is clear how an increment of the number of extracted samples tends to improve performance, with a saturation behavior when too many samples are extracted from each segment. This means that further samples are not necessary since they are redundant with the pixels already selected. In general, the Design-MC strategy presents the best performances, both in terms of accuracy and stability. An additional consideration derives from the comparison between the Design-MC and the ROIs selection methods. In particular, the proposed method gives better performances in terms of OA, Kappa index and stability. Poor values of stability in the ROIs case suggest how this approach depends strongly from user's experience. Therefore, an automatic procedure of segment selection can be particularly helpful for users that are not very familiar with remote sensing images.

In the second part of experiments we evaluate how performances of the different approaches vary in function of the  $\#_{int}$ . The results are reported in Fig. 8.6(a),(c),(e) and Table 8.III(b), in which the number of samples extracted from each segment for the segmentation-based approaches was set to 25. The proposed Design-MC strategy confirms the best performance with respect to the other methods. Moreover, an additional consideration on the clustering process can be done. We note that for Pix-R and Pix-RC strategies very similar performances were obtained. This suggests that applying the clustering procedure before the



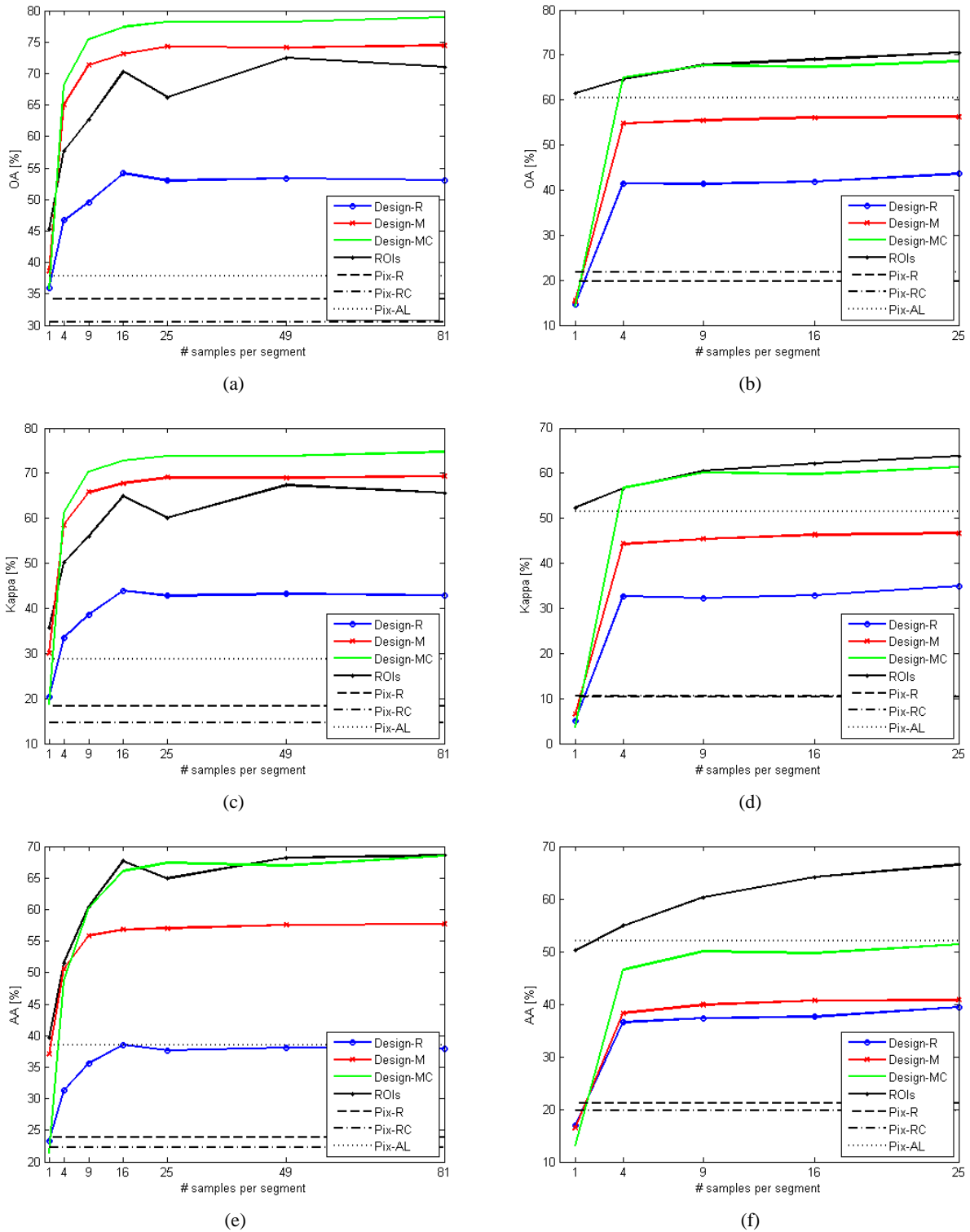


Fig. 8.5. Performances achieved on (a), (c), (e) the Jeddah and (b), (d), (f) the Pavia data sets in terms of (a), (b) OA, (c), (d) Kappa, (e), (f) AA. Each graph shows the results in function of the average number of samples per segment. All results are averaged over ten runs of the approaches. Design-R = random segments, Design-M = maximum size segments, Design-MC = maximum size segments after clustering, ROIs = regions of interest, Pix-R = random pixels, Pix-RC = random pixels after clustering, Pix-AL = pixels with active learning.

sample selection process does not improve the accuracies obtained by the classification process. Instead, an improvement of the performances is experimentally verified for the proposed Design-MC strategy with respect to the Design-M one. Therefore, the clustering algorithm appears fundamental in the step of

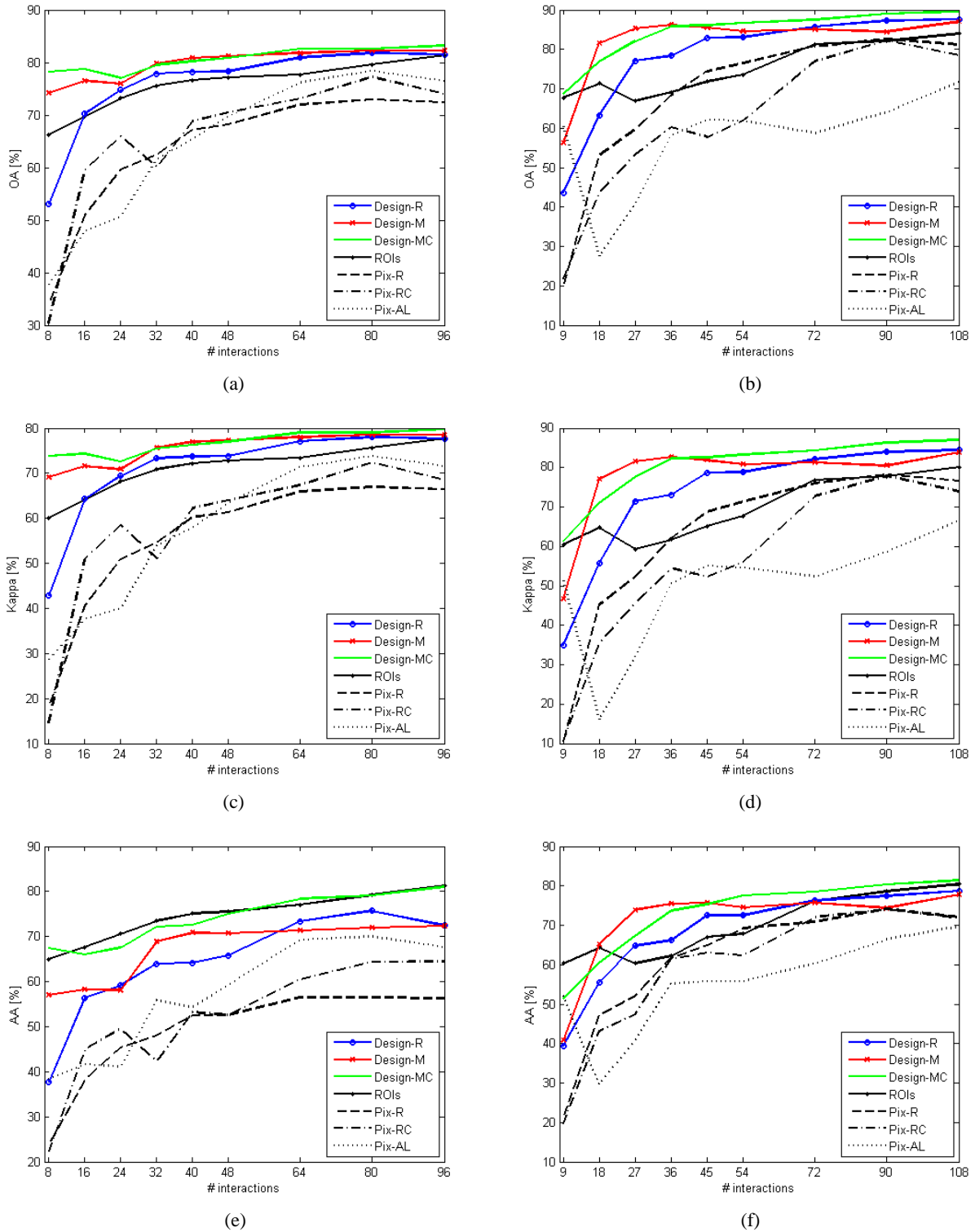


Fig. 8.6. Performances achieved on (a), (c), (e) the Jeddah and (b), (d), (f) the Pavia data sets in terms of (a), (b) OA, (c), (d) Kappa, (e), (f) AA. Each graph shows the results in function of the number of interactions. All results are averaged over ten runs of the approaches. Design-R = random segments, Design-M = maximum size segments, Design-MC = maximum size segments after clustering, ROIs = regions of interest, Pix-R = random pixels, Pix-RC = random pixels after clustering, Pix-AL = pixels with active learning.

automatic segment selection in order to select segments and consequently samples that better span the feature space.

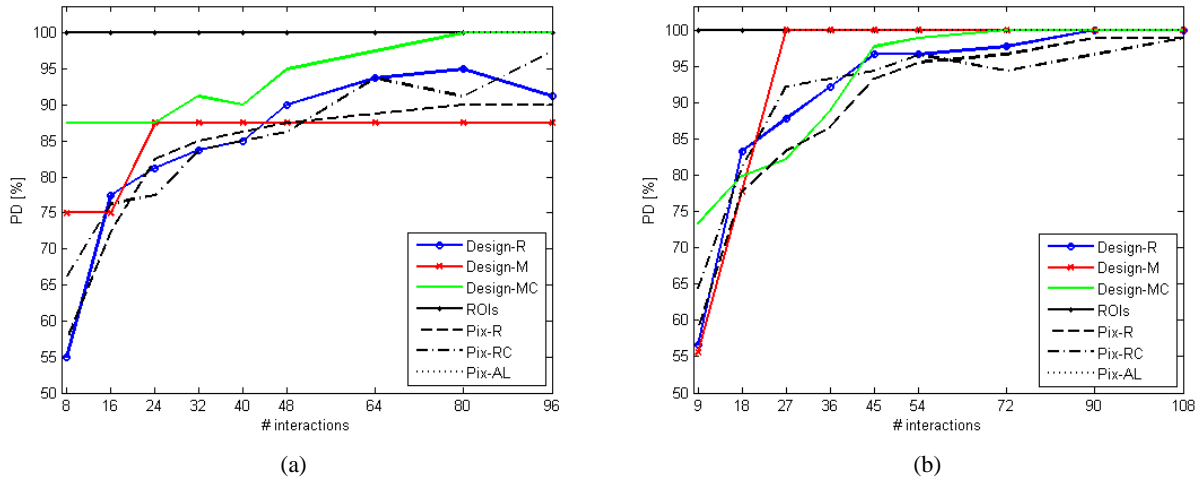


Fig. 8.7. Performances achieved on (a) the Jeddah and (b) the Pavia data sets in terms of class detection probability. Each graph shows the results in function of the number of interactions. All results are averaged over ten runs of the approaches. Design-R = random segments, Design-M = maximum size segments, Design-MC = maximum size segments after clustering, ROIs = regions of interest, Pix-R = random pixels, Pix-RC = random pixels after clustering, Pix-AL = pixels with active learning.

To evaluate the capabilities of the proposed strategies to detect automatically the different classes present in the image, we show in Fig. 8.7(a) the class detection probability (PD) in function of the number of interactions  $\#_{int}$ . It is interesting to observe that the proposed Design-MC method presents the best detection capabilities with respect to the other strategies. We note also that for ROIs and Pix-AL approaches all classes are always detected since they are manually defined by the human user.

To conclude the discussion on the Jeddah data set, we show in Fig. 8.8(a)-(g) some examples of ground-truths obtained by the different methods. In particular, (a)-(c) the segments, (d) the regions and (e)-(g) the pixels selected by the Design-R, Design-M, Design-MC, ROIs, Pix-R, Pix-RC, and Pix-AL strategies, are respectively represented. In particular, for each strategy, we considered the run that gives the value of OA closest to the mean OA reported in Table 8.III(b) for a  $\#_{int}$  equal to 32. We note visually how, once the cost of the labeling process (i.e., the number of interactions  $\#_{int}$ ) is fixed, the proposed strategies allow to increment considerably the portion of the map covered by training samples. Among the proposed methods, the Design-R and Design-M ones select segments belonging to seven different thematic classes, while the class Roof 1, which is a small class in this data set, is excluded. A better covering of the image is obtained using the Design-MS strategy, for which all eight classes are selected.

Similar experiments were conducted on the Pavia data set. First, we show in Fig. 8.4(b) the result of the segmentation algorithm. Also in this case the obtained segmented map is acceptable, although some regions of the image are oversegmented.

The results of the first part of the experiments, in which we fixed the number of interactions  $\#_{int}$  to 9 (i.e., the number of classes) and varied the number of samples extracted from the segments, are summarized in Fig. 8.5(b),(d),(f) and Table 8.IV(a). Also for this data set, we observe an improvement of the performances for the proposed strategies when the number of samples extracted from the segments is incremented. In particular, the Design-MC method confirms the best accuracies among the proposed ones. However, we note that ROIs selections and Pix-AL method present similar or better performances, in particular in terms of AA. This can be explained by the fact that for such strategies we forced manually the inclusion of regions or pixels belonging to all thematic classes. Instead, the proposed methods select segments in a completely automatic way, thus with the risk of not detecting some classes when the number of interactions  $\#_{int}$  is very limited as in this case.

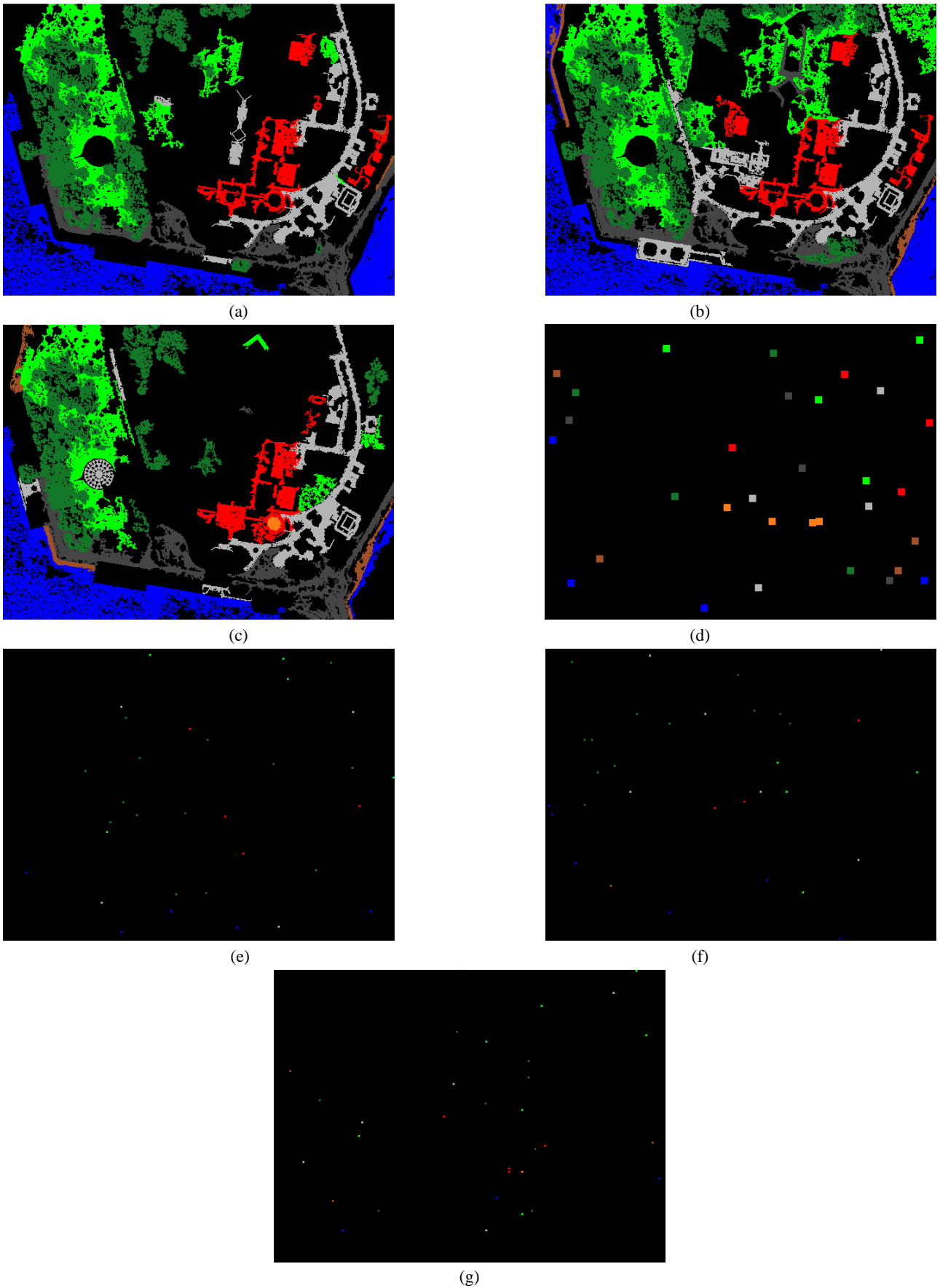


Fig. 8.8. Example of (a)-(c) segments, (d) regions and (e)-(g) pixels selected to build the ground-truth for the Jeddah data set. (a) Design-R = random segments, (b) Design-M = maximum size segments, (c) Design-MC = maximum size segments after clustering, (d) ROIs = regions of interest, (e) Pix-R = random pixels, (f) Pix-RC = random pixels after clustering, (g) Pix-AL = pixels with active learning.

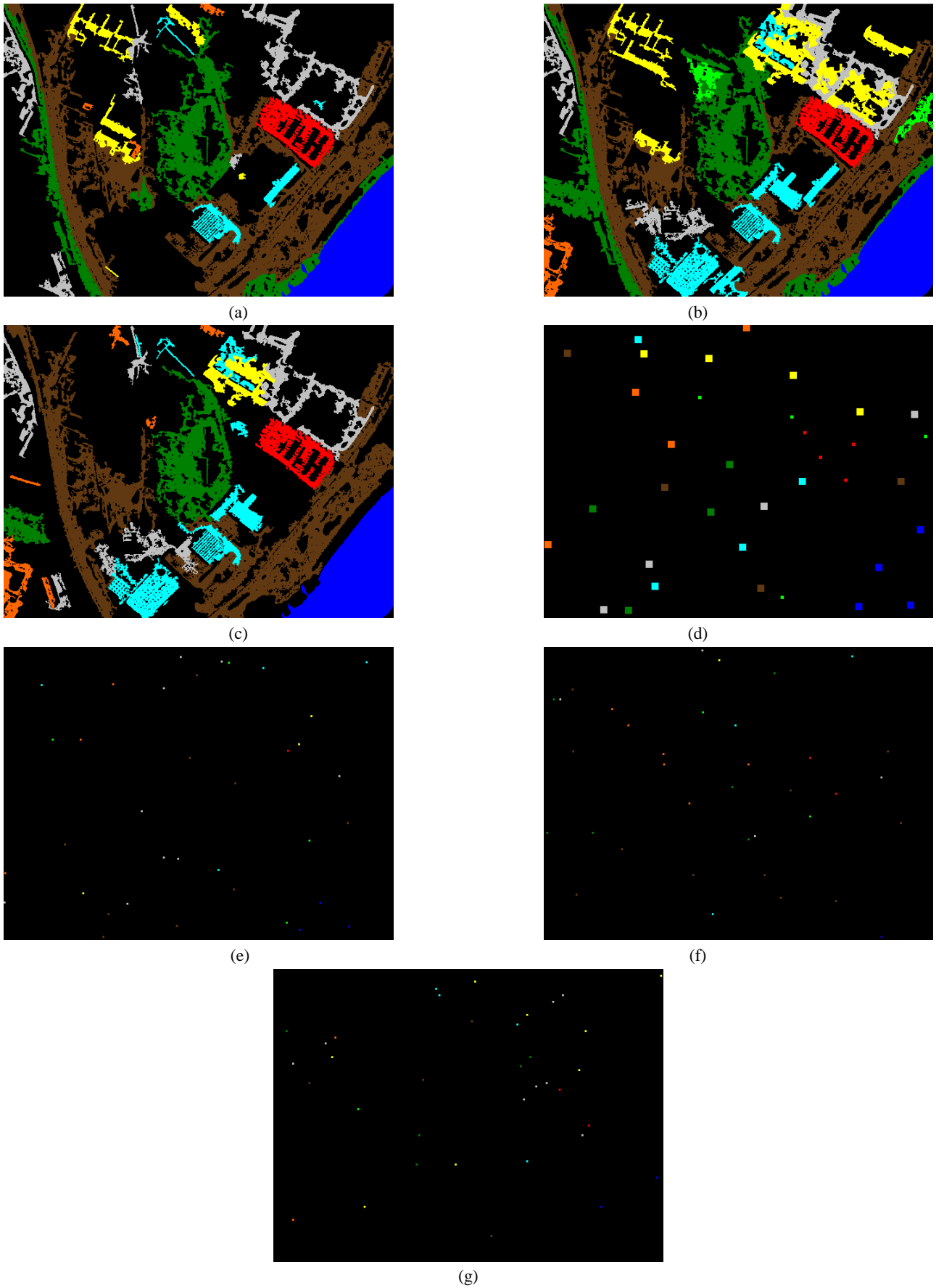


Fig. 8.9. Example of (a)-(c) segments, (d) regions and (e)-(g) pixels selected to build the ground-truth for the Pavia data set. (a) Design-R = random segments, (b) Design-M = maximum size segments, (c) Design-MC = maximum size segments after clustering, (d) ROIs = regions of interest, (e) Pix-R = random pixels, (f) Pix-RC = random pixels after clustering, (g) Pix-AL = pixels with active learning.

TABLE 8.III

OA, KAPPA INDEX, AA, STANDARD DEVIATION ( $\sigma$ ), AND PD ACHIEVED ON THE JEDDAH DATA SET (ALL EXPRESSED IN [%]).

(a)

Method	# <sub>int</sub>	# samples per segment	OA	$\sigma_{OA}$	Kappa	$\sigma_{Kappa}$	AA	$\sigma_{AA}$	PD
Design-R	8	9	49.54	17.80	38.6	22.1	35.62	11.76	55.0
Design-M			71.36	3.86	65.8	4.4	55.84	<b>2.45</b>	75.0
Design-MC			<b>75.51</b>	<b>2.42</b>	<b>70.4</b>	<b>2.9</b>	60.27	4.92	87.5
ROIs		9	62.64	6.95	56.0	8.0	<b>60.46</b>	9.00	100
Pix-R		1	34.12	9.36	18.3	12.5	23.83	7.85	57.5
Pix-RC			30.47	4.95	14.5	6.3	22.33	4.43	66.3
Pix-AL			37.79	12.92	28.7	13.7	38.43	12.73	100

(b)

Method	# <sub>int</sub>	# samples per segment	OA	$\sigma_{OA}$	Kappa	$\sigma_{Kappa}$	AA	$\sigma_{AA}$	PD
Design-R	32	25	77.86	2.36	73.3	2.8	63.96	4.95	83.8
Design-M			<b>79.75</b>	1.74	<b>75.6</b>	2.0	68.84	<b>1.82</b>	87.5
Design-MC			79.52	<b>1.41</b>	75.4	<b>1.7</b>	72.12	4.28	91.3
ROIs		25	75.56	4.51	70.9	5.3	<b>73.45</b>	7.11	100
Pix-R		1	62.47	10.55	54.6	12.4	48.13	8.78	85.0
Pix-RC			60.08	9.41	51.0	11.9	42.25	9.59	83.8
Pix-AL			61.53	14.04	54.0	16.9	55.83	14.94	100

TABLE 8.IV

OA, KAPPA INDEX, AA, STANDARD DEVIATION ( $\sigma$ ), AND PD ACHIEVED ON THE PAVIA DATA SET (ALL EXPRESSED IN [%]).

(a)

Method	# <sub>int</sub>	# samples per segment	OA	$\sigma_{OA}$	Kappa	$\sigma_{Kappa}$	AA	$\sigma_{AA}$	PD
Design-R	9	4	41.49	22.55	32.7	21.9	36.52	9.39	56.7
Design-M			54.75	<b>3.90</b>	44.3	<b>5.0</b>	38.29	<b>5.05</b>	55.6
Design-MC			<b>65.06</b>	8.56	<b>56.7</b>	10.3	46.50	6.19	73.3
ROIs		4	64.62	11.35	<b>56.6</b>	13.0	<b>54.97</b>	11.78	100
Pix-R		1	19.71	16.11	10.3	16.3	21.10	8.21	58.9
Pix-RC			21.75	13.16	10.6	12.4	19.73	7.56	64.4
Pix-AL			60.48	8.51	51.4	10.3	52.07	11.94	100

(b)

Method	# <sub>int</sub>	# samples per segment	OA	$\sigma_{OA}$	Kappa	$\sigma_{Kappa}$	AA	$\sigma_{AA}$	PD
Design-R	36	9	78.36	5.10	73.1	6.2	66.20	5.31	92.2
Design-M			<b>86.21</b>	<b>2.14</b>	<b>82.8</b>	<b>2.7</b>	<b>75.42</b>	2.86	100
Design-MC			85.86	3.27	82.4	4.1	73.67	5.08	88.9
ROIs		9	69.14	7.48	61.7	9.4	62.23	9.24	100
Pix-R		1	68.33	13.49	62.1	13.9	61.71	6.55	86.7
Pix-RC			60.20	18.68	54.5	19.4	61.44	9.31	93.3
Pix-AL			58.24	17.48	50.7	17.9	55.22	12.27	100

The results obtained in the second part of the experiments, in which we incremented the number of interactions #<sub>int</sub>, are shown in Fig. 8.6(b),(d),(f) and Table 8.IV(b). For such analysis, we considered a number of samples extracted from each segment equal to 9. The proposed strategies exhibit better performances with respect to the other ones. In particular, the Seg-MC method gives in general the best accuracies, although similar or better performances are verified using the Seg-M, in particular when the

number of interactions  $\#_{int}$  is small. This result can be better understood by analyzing Fig. 8.7(b), in which we show the number of detected classes in function of the  $\#_{int}$ . For small values of  $\#_{int}$ , the Seg-M method exhibits better detection capabilities with respect to the Seg-MC one.

Finally, in Fig. 8.9(a)-(g) we show some examples of ground-truths obtained by the different strategies. Also in this case, for each strategy, we considered the run that gives the value of OA closest to the mean OA reported in Table 8.III(b) for a  $\#_{int}$  equal to 36.

## 8.4. Conclusion

In this chapter, we have proposed an innovative framework for the assisted design of the ground-truth for remote sensing image classification problems. First, the original image is segmented using a new method of segmentation based on level sets. Then, significant segments are selected by unsupervised procedures based on clustering, and form the ground-truth after human user labeling. In this way, both spatial and spectral information are considered in the process of ground-truth design. The proposed approach exhibits some advantages: 1) it is performed in a single iteration, thus reducing waiting time for the human user; 2) the labeling process is based on segments, thus facilitating the human user intervention; 3) ground-truth initialization from the human user is no more required; 4) the generated ground-truth is classifier-free. It can be further improved by making it classifier-driven through an active learning process.

In order to validate the proposed approach, we conducted experiments on VHR and hyperspectral images acquired by the IKONOS and the ROSIS sensors, respectively. The obtained results show promising capabilities of the proposed approach in terms of ground-truth design. In particular, advantages in terms of classification accuracy have been empirically evaluated with respect to strategies in which ground-truths are collected by defining ROIs or by following pixel-based approaches.

The main drawback of the proposed framework is given by the necessity to set a priori some free parameters. In particular, the segmentation algorithm requires to fix the parameters  $\mu_1$  and  $\mu_2$ , which allow to control the tradeoff between the goodness of fit and length of the curve. In all the experiments presented in this work, we fixed them empirically to 0.2 and 0.05, respectively. Another important parameter is represented by the desired number of interactions with the human user  $\#_{int}$ . In general, it has to be contained as most as possible in order to minimize the cost related to the human intervention.

## 8.5. Acknowledgment

The authors would like to thank Prof. P. Gamba (University of Pavia, Italy) for providing the hyperspectral image and Dr. C.-C. Chang and Dr. C.-J. Lin for supplying the software LIBSVM used in this research.

## 8.6. References cited in Chapter 8

- [1] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. New York: Wiley, 2001.
- [2] J. A. Richards and X. Jia, *Remote Sensing Digital Image Analysis: An Introduction*. Berlin, Germany: Springer-Verlag, 1999.
- [3] C. E. Brodley and M. A. Friedl, "Identifying mislabeled training data," *J. Artif. Intell. Res.*, vol.11, pp.131–167, 1999.
- [4] U. Rebbapragada and C. E. Brodley, "Class noise mitigation through instance weighting," in *Proc. ECML*, Warsaw, Poland, Sep. 2007, pp. 708–715.
- [5] N. Ghoggali and F. Melgani, "Automatic ground-truth validation with genetic algorithms for multispectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 7, pp. 2172–2181, Jul. 2009.
- [6] Q. Jackson and D. A. Landgrebe, "An adaptive classifier design for high-dimensional data analysis with a limited training data set," *IEEE Trans. Geosci. Remote Sens.*, vol. 39, no. 12, pp. 2664–2679, Dec. 2001.
- [7] A. Palau, F. Melgani, and S. B. Serpico, "Cell algorithms with data inflation for non-parametric classification," *Pattern Recognit. Lett.*, vol. 27, no. 7, pp. 781–790, May 2006.

- [8] T. Joachims, "Transductive inference for text classification using support vector machines," in *Proc. ICML*, 1999, pp. 200–209.
- [9] N. Ghoggali, F. Melgani, and Y. Bazi, "A multiobjective genetic SVM approach for classification problems with limited training samples," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 6, pp. 1707–1718, Jun. 2009.
- [10] P. Mitra, C. A. Murthy, and S. K. Pal, "A probabilistic active support vector learning algorithm," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 3, pp. 413–418, Mar. 2004.
- [11] E. Pasolli and F. Melgani, "Active learning methods for electrocardiographic signal classification," *IEEE Trans. Inf. Technol. Biomed.*, vol. 14, no. 6, pp. 1405–1416, Nov. 2010.
- [12] Y. Zhang, X. Liao, and L. Carin, "Detection of buried targets via active selection of labeled data: application to sensing subsurface UXO," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 11, pp. 2535–2543, Nov. 2004.
- [13] S. Rajan, J. Ghosh, and M. M. Crawford, "An active learning approach to hyperspectral data classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 4, pp. 1231–1242, Apr. 2008.
- [14] W. Di, M. M. Crawford, "Active learning via multi-view and local proximity co-regularization for hyperspectral image classification," *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 3, pp. 618–628, Jun. 2011.
- [15] D. Tuia, F. Ratle, F. Pacifici, M. F. Kanevski, and W. J. Emery, "Active learning methods for remote sensing image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 7, pp. 2218–2232, Jul. 2009.
- [16] E. Pasolli, F. Melgani, and Y. Bazi, "Support vector machine active learning through significance space construction," *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 3, pp. 431–435, May 2011.
- [17] D. Tuia, E. Pasolli, and W. J. Emery, "Using active learning to adapt remote sensing image classifiers," *Remote Sens. Environ.*, vol. 115, no. 9, pp. 2232–2242, Sep. 2011.
- [18] D. Mumford and J. Shah, "Optimal approximation by piecewise smooth functions and associated variational problems," *Commun. Pure Appl. Math.*, vol. 42, no. 5, pp. 577–685, 1989.
- [19] T. Chan and L. Vese, "Active contours without edges," *IEEE Trans. Image Process.*, vol. 10, no. 2, pp. 266–277, Feb. 2001.
- [20] S. Osher and J. A. Sethian, "Fronts propagating with curvature-dependent speed: Algorithm based on Hamilton-Jacobi formulation," *J. Comput. Phys.*, vol. 79, no. 1, pp. 12–49, Nov. 1988.
- [21] Y. Bazi, F. Melgani and H. D. Al-Sharari, "Unsupervised change detection in multispectral remotely sensed imagery with level set methods," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 8, pp. 3178–3187, Aug. 2010.
- [22] A. Paoli, F. Melgani, and E. Pasolli, "Clustering of hyperspectral images based on multiobjective particle swarm optimization," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 12, pp. 4175–4188, Dec. 2009.
- [23] V.N. Vapnik, *Statistical Learning Theory*, New York: Wiley, 1998.
- [24] E. Blanzieri and F. Melgani, "An adaptive SVM nearest neighbor classifier for remotely sensed imagery," in *Proc. IGARSS*, Denver, CO, Aug. 2006, pp. 3931–3934.
- [25] E. Pasolli, F. Melgani, and M. Donelli, "Automatic analysis of GPR images: A pattern-recognition approach," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 7, pp. 2206–2217, Jul. 2009.
- [26] J. Cohen, "A coefficient of agreement for nominal scales," *Educational and Psychological Measurement*, vol. 20, no. 1, pp. 37–46, 1960.
- [27] G. Schohn and D. Cohn, "Less is more: Active learning with support vectors machines," in *Proc. 17th ICML*, Stanford, CA, 2000, pp. 839–846.
- [28] P. Mitra, B. Uma Shankar, and S. Pal, "Segmentation of multispectral remote sensing images using active support vector machines," *Pattern Recogn. Lett.*, vol. 25, no. 9, pp. 1067–1074, Jul. 2004.



## 9. Conclusions

*Abstract – In this chapter we report general conclusions on the methodological and experimental developments conveyed by the present thesis. The reader is referred to the previous single chapters for more detailed discussions about the different proposed methods.*

In this thesis, the active learning approach has been investigated to address the problem of training sample collection for classification and regression problems. Several methodological aspects and theoretical solutions have been proposed and validated experimentally in different application fields, such as remote sensing, biomedical, and chemometrics. In the following, we will briefly summarize the conclusions drawn for each of the presented strategies. We refer the reader to the single chapters for more details.

In Section 2, the active learning approach has been introduced in the biomedical field for the classification of electrocardiographic (ECG) signals. Three strategies based on support vector machine (SVM) classification have been presented, namely margin sampling (MS), posterior probability, and query by committee. The experimental results obtained on simulated and real ECG data show good capabilities of the proposed methods for selecting training samples. In general, all the proposed methods are characterized by higher performance in terms of both accuracies and stability with respect to a completely random selection strategy. Comparing them, the strategy based on the MS principle seems the best as it quickly selects the most informative samples. Another interesting result is that active learning methods are able to give accuracies slightly better than the “full” classifier, confirming their usefulness in reducing mislabeling risks. While in this research the initial training set was chosen in a random way, we think that a more sophisticated initialization strategy could further improve the performance of the active learning process.

In Section 3, we have proposed a new strategy specifically developed for SVM classification of remote sensing images. The experimental results obtained on very high resolution (VHR) and hyperspectral images show good capabilities of the proposed method in terms of training sample selection. Advantages in terms of convergence speed, stability, and reduction of the number of support vectors (SVs) have been empirically evaluated with respect to the state-of-the-art MS strategy. The proposed strategy exhibits two main drawbacks. First, in case of overfitting (due for instance to model selection problems), most of the samples become SVs; and so, most of the learning samples are detected as significant, thus making the proposed algorithm tend to a simple random sample selection. Second, an increment of the computational cost is verified, given by the training of two stages of SVM classifiers. Also for this method, we think that the active learning process could be further improved using a more elaborated initialization strategy.

While the active learning strategies present in the remote sensing literature work in the spectral domain only, in Section 4 we have proposed to combine spectral and spatial information. For this purpose, we have introduced three different criteria in the spatial domain in order to favor the selection of samples distant from the samples already composing the current training set. The three criteria are based on Euclidean distances, Parzen window method, and entropy variation, respectively. Experiments on two VHR images show the proposed approach exhibits advantages in terms of classification accuracy and classification reliability with respect to strategies that do not exploit spatial information. The main drawback of the proposed method is represented by an increment of the computational cost, given by the calculation of further measures in order to take into account the spatial contribution. While in this work we considered the state-of-the-art MS strategy as spectral heuristic, for its simplicity and effectiveness, the proposed approach can be in general applied in conjunction with any traditional active learning method that exploits the samples in the spectral domain.

In Section 5, we have proposed a way to use active learning to solve the problem of covariate shift, which may occur when a classifier trained on a portion of the image is applied to the rest of the image. The experimental results obtained on hyperspectral and VHR data sets demonstrate good capability of the proposed method for selecting samples that allow rapid convergence to an optimal solution. Moreover, the use of a clustering-based selection strategy allows us to discover new classes in case they have been omitted in the initial training set. Such strategies for optimal sampling guarantee signature extension and can be extended to a large variety of applications dealing with spectral data, as it is not dependent on the image characteristics of the data. An example could be the classification of ECG signals.

After focusing on classification problems in the previous chapters, in Section 6 the active learning approach has been applied in the regression context to the estimation of biophysical parameters from remote sensing data. Different strategies for Gaussian Process (GP) and SVM regression have been proposed. For GP regression, the first two methods are based on adding samples that are distant from the current training samples in the kernel space, while the third one uses a pool of regressors in order to select the samples with the greater disagreements between the regressors of the pool. Finally, the last strategy exploits an intrinsic GP regression outcome to pick up the most difficult samples. For SVM regression, the method based on the pool of regressors and two additional strategies based on the selection of the samples distant from the current support vectors are proposed. The experimental results obtained on simulated MERIS and real SeaBAM data sets show good capabilities of the proposed strategies for selecting significant samples. In general, the proposed methods are characterized by higher performances in terms of both accuracy and stability with respect to a completely random selection strategy. Though in this work we focused on GP and SVM regression, the active selection of the training samples could be used in combination with other supervised regression approaches. Moreover, while the initial training set was chosen in a random way, a more sophisticated initialization strategy could be envisioned in order to improve further the active learning approach.

Similarly to the previous chapter, in Section 7 the active learning has been applied for regression problems, but in this case to estimate the chemical concentrations from spectroscopic data. Some strategies for partial least squares regression (PLSR) and SVM regression have been proposed. For PLSR, the first method is based on adding samples that are distant from the current training samples in the feature space, while the second one is based on the pool of regressors. For SVM, the method based on the pool of regressors and an additional strategy based on the selection of the samples distant from the current support vectors are presented. The experimental results on three different real data sets show higher performances of the proposed strategies in terms of both accuracy and stability with respect to a completely random selection strategy. Comparing them, the best active strategy appears the one based on the pool of regressors for both PLSR and SVM. It is however the most computational demanding since it needs the training of different regressors to build the pool.

Finally, in Section 8 we have proposed a framework for the assisted design of the ground-truth for remote sensing image classification problems. First, the original image is segmented using a method of segmentation based on level sets. Then, significant segments are selected by unsupervised procedures based on clustering, and form the ground-truth after human user labeling. In this way, both spatial and spectral information are considered in the process of ground-truth design. The proposed approach exhibits some advantages. First, it is performed in a single iteration, thus reducing waiting time for the human user. Second, the labeling process is based on segments, thus facilitating the human user intervention. Third, ground-truth initialization from the human user is no more required. Fourth, the generated ground-truth is classifier-free and it can be further improved by making it classifier-driven through an active learning process. The experimental results on VHR and hyperspectral images show promising capabilities of the proposed approach in terms of ground-truth design. In particular, advantages in terms of classification accuracy have been empirically evaluated with respect to strategies in which ground-truths are collected by defining regions of interest or by following pixel-based approaches. The main drawback of the proposed framework is given by the necessity to set a priori some free parameters. In particular, the segmentation algorithm requires to fix the parameters that control the tradeoff between the goodness of fit and length of the curve. Another important parameter is represented by the desired number of interactions with the human user. In general, it has to be contained as most as possible in order to minimize the cost related to the human intervention.

To conclude, the contributions provided in this thesis have been focused on the development of active learning methodologies to address the problem of training sample collection for classification and regression

problems. Such contributions have been critically analyzed considering the state-of-the-art of the related research topics, and have been compared with reference approaches by means of in-depth testing experiments. The results turned out to be satisfactory, and confirmed that the research reported in this dissertation have made interesting contributions to the faced methodological issues.

## 10. List of Related Publications

### 10.1. Published Journal Papers

- [J.1] E. Pasolli and F. Melgani, “Active learning methods for electrocardiographic signal classification,” *IEEE Trans. Inf. Technol. Biomed.*, vol. 14, no. 6, pp. 1405–1416, Nov. 2010.
- [J.2] E. Pasolli, F. Melgani, and Y. Bazi, “SVM active learning through significance space construction”, *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 3, pp. 431–435, May 2011.
- [J.3] D. Tuia, E. Pasolli, and W. J. Emery, “Using active learning to adapt remote sensing image classifiers,” *Remote Sens. Environ.*, vol. 115, no. 9, pp. 2232-2242, Sep. 2011.

### 10.2. Journal Papers in Revision

- [J.4] E. Pasolli, F. Melgani, N. Alajlan, and Y. Bazi, “Active learning methods for biophysical parameter estimation,” submitted to *IEEE Trans. Geosci. Remote Sens.*.
- [J.5] F. Douak, F. Melgani, N. Alajlan, E. Pasolli, and N. Benoudjit, “Active learning for spectroscopic data regression,” submitted to *J. Chemometr.*

### 10.3. Journal Papers in Preparation

- [J.6] E. Pasolli, F. Melgani, D. Tuia, F. Pacifici, and W. J. Emery, “SVM active learning using spatial information”.
- [J.7] E. Pasolli, F. Melgani, and N. Alajlan, “A framework for computer-aided ground-truth collection for optical image classification”.

### 10.4. Conference Proceedings

- [C.1] E. Pasolli and F. Melgani, “Model-based active learning for SVM classification of remote sensing images,” in Proc. *IGARSS*, Honolulu, HI, Jul. 2010, pp. 820–823.
- [C.2] D. Tuia, E. Pasolli, and W. J. Emery, “Dataset shift adaptation with active queries,” in Proc. *JURSE*, Munich, Germany, Apr. 2011, pp. 121–124.
- [C.3] E. Pasolli, F. Melgani, D. Tuia, F. Pacifici, and W. J. Emery, “Improving active learning methods using spatial information,” in Proc. *IGARSS*, Vancouver, Canada, Jul. 2011.
- [C.4] E. Pasolli and F. Melgani, “Gaussian process regression within an active learning scheme,” in Proc. *IGARSS*, Vancouver, Canada, Jul. 2011.
- [C.5] E. Pasolli and F. Melgani, “Ground-truth assisted design for remote sensing image classification,” in Proc. *IGARSS*, Vancouver, Canada, Jul. 2011.