NOVA IMS Information Management School

# DOCTORATE PROGRAM

## Information Management

### Specialization in Geographical Information Systems

## Contributions for the improvement of specific class mapping

**Joel Dinis Baptista Ferreira da Silva**

A thesis submitted in partial fulfilment of the requirements for the degree of Doctor in Information Management

**November, 2016**

**NOVA Information Management School**
NOVA University of Lisbon

Prof. Doutor Mário Caetano, Co-supervisor

Prof. Doutor Fernando Bação, Co-supervisor

**Contributions for the improvement of specific class mapping**



This document was created using the (pdf)LaTeX processor, based in the "unlthesis" template[1], developed at the Dep. Informática of FCT-NOVA [2].
[1] https://github.com/joaomlourenco/unlthesis [2] http://www.di.fct.unl.pt

*When solving a problem of interest, do not solve a more general problem as an intermediate step*
*- Vladimir Vapnik*

# Acknowledgements

A few lines are too short to make a complete account of my deep appreciation for my supervisors. To Professor Mário Caetano and Professor Fernando Bação, I wish to thank them for their trust and encouragements which have been essential throughout these four years. Thank you for being an enthusiastic and sagacious individuals who provided me a sound support. And I also wish to thank Professor Giles Foody. Although I have never met him personally, he has always provided great insights and replies to my often annoying e-mails.

I am indebted to my friends and colleagues from NOVA IMS for providing me a good environment in which to work. I am grateful to Professor Marco Painho, Tiago Oliveira, Luis Almeida, Alexandre Baptista, Cleiton Rodrigues, Otávio Sian, Ângela Santos, Rita Ferreira and Raquel Silva.

I wish to thank to my aunt Guida, to my mother Mina, to my sister Maria and to my father Luis for their encouragements throughout my graduate work. And finally I wish to thank to my girlfriend and partner Raquel for her love and support.

My big thank you to you all.

# Abstract

The analysis of remotely sensed imagery has become a fundamental task for many environmental centred activities, not just scientific but also management related. In particular, the use of land cover maps depicting a particular study site is an integral part of many research projects, as they are not just a fundamental variable in environmental models but also base information supporting policy decisions.

Land cover mapping assisted by supervised classification is today a staple tool of any analyst processing remotely sensed data, insomuch as these techniques allow users to map entire sites of interest in a comprehensive way.

Many remote sensing projects are usually interested in a small number of land cover classes present in a study area and not in all classes that make-up the landscape. When focus is on a particular sub-set of classes of interest, conventional supervised classification may be sub-optimal for the discrimination of these specific target classes.

The process of producing a non-exhaustive map, that is depicting only the classes of interest for the user, is called specific class mapping.

This is the topic of this dissertation. Here, specific class mapping is examined to understand its origins, developments, adoption and current limitations. The main research goal is then to contribute for the understanding and improvement of this topic, while presenting its main constrains in a clear way and proposing enhanced methods at the reach of the non-expert user. In detail, this study starts by analysing the definition of specific class mapping and why the conventional multi-class supervised classification process may yield sub-optimal outcomes.

Attention then is turn to the previous works that have tackled this problem. From here a synthesis is made, categorising and characterising previous methodologies. Its then learnt that the methodologies tackling specific class mapping fall under two broad categories, the binarisation approaches and the singe-class approaches, and that both types are not without problems. This is the starting point of the development component of this dissertation that branches out in three research lines.

First, cost-sensitive learning is utilised to improve specific class mapping. In previous studies it was shown that it may be susceptible to data imbalance problems present in the training data set, since the classes of interest are often a small part of the training set. As a result the classification may be biased towards the largest classes and, thus, be

sub-optimal for the discrimination of the classes of interest. Here cost-sensitive learning is used to balance the training data set to minimise the effects of data imbalance. In this approach errors committed in the minority class are treated as being costlier than errors committed in the majority class. Cost-sensitive approaches are typically implemented by weighting training data points accordingly to their importance to the analysis. By shifting the weight of the data set from the majority class to the minority class, the user is capable to inform the learning process that training data points in the minority class are as critical as the points in the majority class. The results of this study indicate that this simple approach is capable to improve the process of specific class mapping by increasing the accuracy to which the classes of interest are discriminated.

Second, the combined use single-class classifiers for specific class mapping is explored. Supervised algorithms for single-class classification are particularly attractive due to its reduced training requirements. Unlike other methods where all classes present in the study site regardless of its relevance for the particular objective to the users, single-class classifiers rely exclusively on the training of the class of interest. However, these methods can only solve specific classification problems with one class of interest. If more classes are important, those methods cannot be directly utilised. Here is proposed three combining methodologies to combine single-class classifiers to map subsets of land cover classes. The results indicate that an intelligent combination of single-class classifiers can be used to achieve accurate results, statistically non-inferior to the standard multi-class classification, without the need of an exhaustive training set, saving resources that can be allocated to other steps of the data analysis process

Third, the combined use of cost-sensitive and semi-supervised learning to improve specific class mapping is explored. A limitation of the specific class binary approaches is that they still require training data from secondary classes, and that may be costly. On the other hand, a limitation of the specific class single-class approaches is that, while requiring only training data from the specific classes of interest, this method tend to overestimate the extension of the classes of interest. This is because the classifier is trained without information about the negative part of the classification space. A way to overcome this is with semi-supervised learning, where the data points for the negative class are randomly sampled from the classification space. However that may include false negatives. To overcome this difficult, cost-sensitive learning is utilised to mitigate the effect of these potentially misclassified data points. Cost weights were here defined using an exponential model that assign more weight to the negative data points that are more likely to be correctly labelled and less to the points that are more likely to be mislabelled. The results show that accuracy achieved with the proposed method is statistically non-inferior to that achieved with standard binary classification requiring however much less training effort.

**Keywords:** Remote sensing; land cover mapping; specific class mapping; cost-sensitive learning; semi-supervised learning; single-class learning

# Resumo

A analise de imagens de satélite tornaram-se uma peça fundamental em diversas aplicações de cariz ambiental, não só científicos, mas também administrativos. Em particular, a utilização de mapas de ocupação do solo é uma componente integral em muitos projetos de investigação, devido não só à sua utilização como parâmetro em modelos ambientais, mas também como informação base ao apoio a decisões politicas.

Contudo, em muitos destes projetos, os utilizadores estão tipicamente interessados num pequeno subconjunto das classes presentes na sua área de estudo e não numa completa caracterização. Nestes casos, o uso de métodos convencionais de classificação assistida podem não ser ótimos para a discriminação dessas classes de interesse. A este processo de classificação não exaustiva chama-se mapeamento de classes especificas.

Este é o tópico desta dissertação. Nesta analise, o conceito de mapeamento de classes especificas é examinado de forma a melhor compreender as suas origens, desenvolvimentos, adoção e limitações. O objetivo principal desta dissertação é assim contribuir para o entendimento e melhoramento deste tópico, apresentando as suas principais condicionantes de uma forma clara, propondo métodos melhorados ao alcance do utilizador médio.

Este estudo começa por analisar a definição do conceito de mapeamento de classes especificas e a razão pela qual os métodos de classificação assistida convencionais podem não ser apropriados.

Seguidamente, é realizada uma análise aos estudos anteriormente desenvolvidos que exploraram este tema. Daqui é feita a síntese, caracterizando e categorizados as metodologias anteriores.

Deste exame resulta que as metodologias de mapeamento de classes especificas encaixam-se em dois grandes grupos, abordagens de binarização e abordagens monoclasse, e que ambos têm vantagens e desvantagens. Daqui ramificam-se as linhas de investigação que se dividem em três.

Na primeira, é utilizada uma metodologia de aprendizagem *cost-sensitive* para melhorar o mapeamento de classes especificas por binarização. Estudos anteriores demonstraram que a abordagem por binarização conduzir a classificadores treinados com dados desbalanceados uma vez que as classes de interesse são tipicamente apenas uma pequena componente no conjunto de treino. Como resultado, a classificação pode

ficar enviesada na direção da classe maioritária, pelo que resulta numa classificação sub-ótima para a discriminação das classes de interesse. A aprendizagem *cost-sensitive* é então utilizada para balançar o conjunto de treino. Com esta abordagem os erros cometidos na classe minoritária são mais caros que aqueles cometidos na classe maioritária. Estas abordagem são usualmente implementadas fazendo associar a cada ponto um peso que representa o custo em classificar erroneamente esse ponto. Desta forma é possível guiar o processo de aprendizagem de forma a não sacrificar os pontos da classe minoritária para minimizar o erro de classificação global. Os resultados deste estudo indicam que esta abordagem foi capaz de ultrapassar os problemas de desbalanceamento e produzir um mapa com as classes de interesse exato.

Na segunda linha, é explorada a combinação de classificadores monoclasse para o mapeamento de classes especificas. Os classificadores monoclasse são particularmente atrativos porque só necessitam de dados da classe para o qual estão a ser treinados. Contudo, estes classificadores não conseguem classificar problemas com mais que uma classe. Se houver mais que uma classe de interesse, estes métodos não podem ser aplicados diretamente. Neste estudo são propostas três formas de combinar múltiplos classificadores monoclasse para mapear um subconjunto de classes de interesse. Estas abordagens foram comparadas com a abordagem assistida convencional. Os resultados indicam que uma combinação inteligente de classificadores monoclasse podem ser utilizados para mapear diversas classes de interesse sem perda de exatidão, requerendo, contudo, apenas treino para as classes de interesse.

Na terceira linha de investigação, o uso combinado da aprendizagem *cost-sensitive* com aprendizagem semi-assistida é explorado com o objetivo de melhorar o mapeamento de classes especificas. Uma limitação com as abordagens de binarização é que apesar de direcionada para a discriminação de classes de interesse requerem ainda treino para todas as classes, independentemente de serem ou não de interesse, o que pode ser dispendioso. Por outro lado, as abordagens monoclasse apesar de necessitarem de treino apenas para as classes de interesse tendem sobrestima-las. Isto acontece porque os classificadores monoclasse são treinados sem acesso a informação da parte negativa do espaço de classificação. Uma forma de ultrapassar estas dificuldades é recorrendo à aprendizagem semi-assistida, onde os pontos representando treino das classes sem interesse são obtidos por meio de uma amostragem e seguidamente são rotulados como negativos. Contudo, esse processo pode incluir falsos negativos no conjunto de treino. Para minimizar os efeitos destes potenciais falsos negativos, uma abordagem *cost-sensitive* é utilizada. Os pesos associados a estes pontos refletem a probabilidade de serem um falso negativo, de modo a que pontos com elevada probabilidade recebem menos peso e pontos com baixa probabilidade recebem mais peso. Os resultados indicam que este método combinado é estatisticamente não inferior ao benchmark, requerendo, contudo, muito menos treino.

**Palavras-chave:** Deteção remota; mapas de ocupação de solo; mapeamento de classes específicas; apredizagem *cost-sensitive*; aprendizagem semi-assistida; aprendizagem monoclasse

# Contents

# List of Figures

# List of Tables

# Acronyms

| | |
|---|---|
| BM | Benchmark. |
| BSVM | Biased Support Vector Machines. |
| DAG | Directed Acyclic Graph. |
| DDAG | Decision Directed Acyclic Graph. |
| FN | False Negative. |
| FOVO | Focused One-vs-One. |
| FOVR | Focused One-vs-Rest. |
| FP | False Positive. |
| FSVM | Focused Support Vector Machines. |
| GLOVIS | Global Visualization Viewer. |
| HM | High Mangrove. |
| KKT | Karush-Kuhn-Tucker. |
| LM | Low Mangrove. |
| NDVI | Normalised Difference Vegetation Index. |
| OCSVM | One-Class Support Vector Machines. |

| | |
|---|---|
| OVA | One-vs-All. |
| OVO | One-vs-One. |
| OVR | One-vs-Rest. |
| SSVM | Standard Support Vector Machines. |
| SVDD | Support Vector Data Description. |
| SVM | Support Vector Machines. |
| TN | True Negative. |
| TP | True Positive. |
| UNESCO | United Nations Educational, Scientific and Cultural Organization. |
| USGS | United States Geologic Survey. |
| WSVM | Weighted Support Vector Machines. |

Chapter 1

# Introduction

## 1.1 Motivation and research lines

Remote sensing is today an integral part of any Earth science and the information derived from remotely sensed data is utilised in a multitude of applications. In particular, the supervised classification of satellite imagery is of upmost importance, since land cover maps are an important component in many environment research activities, but also for managerial purposes. Thus the number of different types of users and applications is vast and so are their requirements. Although land cover maps that characterise an entire region of analysis are necessary and satisfy particular needs, often users are only interested in a subset of classes present in the region and not on its entire characterisation.

Specific class mapping consists in producing a non-exhaustive thematic map. This contrasts with the conventional mapping process that produces a thematic land cover map depicting all classes thereof present, and thus exhaustively characterising it. The goals of specific class mapping is, then, to depict only a subset of land cover classes of interest present in the study area.

The broad objective of this research is to investigate the usefulness and limitations of specific class mapping processes and explore possible ways to improve it with machine learning methodologies.

When users are interested in just a subset of classes present in the study area, often users adopt a conventional supervised classification methodology. In other words, users produce a map depicting all classes regardless of their interest for their application. Indeed, users solve a large problem than that they need to solve. If the specific class mapping problem is solved by tackling it as a larger problem, that is by mapping

all classes instead of just those of interest, users may obtain sub-optimal results. Additionally, this solution may force users to allocate extra time to collect training pixels to classes of no relevance for theirs objective, thus making the mapping process less efficient.

However, applying a conventional supervised approach to solve the specific class mapping problem directly leads to two technical problems. First, most learning algorithms require an exhaustive definition of the study area. In other words, the algorithms of classification assume that the training data set utilised to inform the learning of the classifier represents an partition of the space to be classified. If this requirement is not fulfilled, pixels belonging to untrained classes are allocated to classes represented in the training set. Second, with conventional classification methods the aim is to minimise the general misclassification rate but not necessarily to optimally discriminate the classes of interest. This is because the model that best discriminates the classes of interest may induce classification errors between classes of no interest and thus may not minimise the general misclassification rate.

Specific class mapping has been tackled before, essentially in two different ways. The first way is by binarisation. That is, the multi-class problem is broken down in smaller binary classification problems. Typically this is done by combining every class of no interest in one single nominal class, usually called "others". The second way is by single-class learning. In other words, by developing a classifier utilising only data from the class of interest and thus focusing the entire process on that particular class. However, these approaches are not without problems.

The main difficulties with the binarisation approach are essentially four. First, by agglomerating all classes of no interest in one big nominal class, the resulted class composition may lead to data imbalanced related issues. As consequence, it may bias the learning process toward this big class, underestimating the extension of the classes of interest. Second, if two or more classes are of interest, classifier combination schemes have to be utilised, which may not always be a trivial matter. Third, the free parameters definition have to be focused on the discrimination of the classes of interest. If such is not done, it is not guaranteed that the resulted classifier is optimally defined for the identification of these classes. Fourth, from the operational point of view, a user has to collect training data for all classes present in the region of interest. Thus, in this sense, the training requirements are similar to that of the conventional multi-class supervised methods.

On the other hand, the main difficulties with the single-class approach are essentially two. First, since the only available training data are points from the class of interest, it is not possible to compute the classification accuracy to fix the free parameters. The only possible metric is sensitivity, which only represents one side of the problem. Thus, fine tuning such algorithm may be difficult. Second, if focus is in more than one class of interest, it is necessary the combination of multiple classifiers, which for this type of classifiers is not clear how and is still an on-going research topic. In

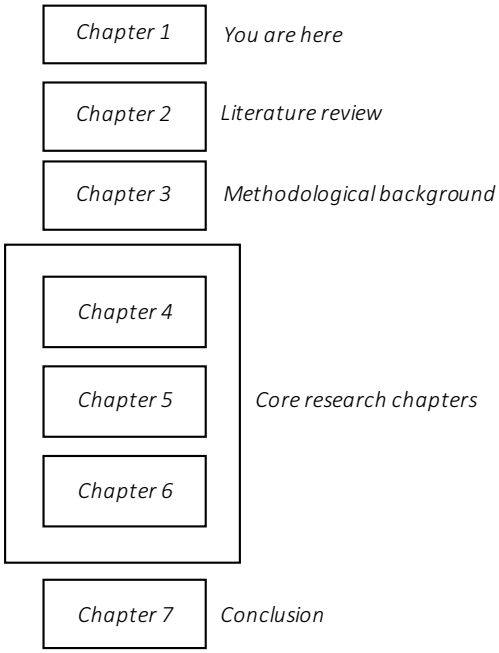



Figure 1.1: Diagram representing this document structure.

general, the difficulties faced by this approach is a consequence of its attractive point, that is the limited need of training and focused approach to the class of interest.

This the starting point of this research. From here three research lines branch out. First, the effects of data imbalanced that may occur in specific class mapping when adopting a binarisation approach is understood and explored and cost-sensitive learning is utilised to minimise its effects. Second, the combined use single-class classifiers with the intention to apply to a specific class mapping scenario is examined and combination schemes are tested for its viability. Third, the combination of semi-supervised learning and cost-sensitive learning is combined to improve the process of specific class mapping.

## 1.2 Document organisation

The dissertation is composed by six chapters.

Chapter 1, the one the reader is presently reading, summarises the research context, goals and main motivations. Subsection 1.1 *Motivation and research lines* provides a brief characterisation of this dissertation.

Chapter 2 presents the literature review in a digested form, since the core chapters, that is chapters 4, 5, and 6, provide a more specific reviews.

Chapter 3 elaborates, with different degrees of detail, the most important technical concepts that base this research. In other words, concepts support vector machines, one-class support vector machines, and the particularities of how to tune and compare different classifiers are presented in a contained way. This chapter is long and may

be technically dense. However, the reader will find all the necessary elements to fully understand the most technically challenge parts of the core chapters.

Chapters 4, 5, and 6 are the core of this research. They are composed by three standalone manuscripts submitted (at the writing time) in Peer-Reviewed Journals. Their order is not arbitrary in the sense that they are organised as a progression in the research process. Chapter 3 for example starts where previous research on specific class mapping ended, and resumes it tackling its limitations. Chapter 4 learns from the previous chapter and improves it, and the same happens with chapter 5. These three documents share the same structure. They first start with an introduction to the problem and how that work is a follow up of previous analysis. Then a detailed background analysis is undertaken to inform the reader about the most important concepts that sustain the methodological approach. However, if the reader has read chapter 3, this part may be omitted for time saving reasons. Next the data and methods section, where the data employed in the study is described and the methodological steps are explained. The chapters end with a result and discussion section, to explain the most important results of the analysis and with a summary and conclusion.

Chapter 7 presents the overarching final remarks of this work, highlighting the more important results and possible follow ups of this research thematic.

## 1.3 Papers originated from this thesis

Three papers were originated from this thesis:

1. Silva, J., Dieng, M., Bacao, F., Foody, G., and Caetano, M. *Improving specific class mapping by cost-sensitive learning*, submitted to *International Journal of Remote Sensing*.

2. Silva, J., Bacao, and Caetano, M. *Combined use one-class classifiers for specific class mapping: An experiment with forest classification*, submitted to *International Journal of Digital Earth*.

3. Silva, J., Bacao, and Caetano, M. *Specific land cover class mapping by semi-supervised weighted support vector machines*, submitted to *Remote Sensing*.

## 1.4 Note about notation

Vectors are represented with bold lowercase letter (latin or greek), for example $\boldsymbol{x}$. Vectors with subscript, like $\boldsymbol{x}_i$, means the $i$-th vector and not the $i$-th element of the vector $\boldsymbol{x}$. To represent the $i$-th element of the vector $\boldsymbol{x}$ it is used $x_i$. Matrices are represented with bold uppercase letter (latin or greek), for example $\boldsymbol{X}$. The $(i, j)$ element of $\boldsymbol{X}$ is represented as $X_{ij}$. Scalars are represent with normal (i.e. not bold) lowercase letter

(latin or greek), for example $x$. For summation notation, $\sum_{i,j}$ represents the double summation $\sum_i \sum_j$.

Chapter 2

# Literature review

**Abstract** Remote sensing is an important source of data for the production of land cover maps. Particularly important is the supervised classification methodology that allow users to produce maps that meet their specific research goals. In this chapter specific class mapping is defined and the literature review about this topic is presented. The purpose here is not to present a technically explicit description of the concepts and methods, but rather present a short account of these two elements in the remote sensing context. In other words, how they came to be recognised as relevant, defined and what are the main advances in the process of remotely sensed data classification, concerning the classification of specific classes. From the literature, it was possible to organise the different approaches in two boarder categories: the binarisation methods and the single-class methods. The main advantages and disadvantages of each approach are highlighted. It was possible to identify that the process of specific class mapping also require the incorporation of specific accuracy metrics and that bias, to and against, the class of interest may occur in the process.

## 2.1 Introduction

The land cover is the observed biophysical cover on the Earth's surface, and is key for environmental information [60]. The information provided associated to land cover is utilised in many scientific domains, resource management and policy purposes, and for a range of human activities [107], since it is the manifestation of the local climate, landforms and of the human use of land [25].

Remote sensing data is an important source of land cover information and has been extensively used to map and monitor land cover classes over time in order to fulfil a variety of of scientific and managerial purposes [107]. And supervised classification,

in particular, has been frequently used to derive thematic maps depicting the land cover classes present in the study area from remotely sensed data. Indeed, remotely sensed data classification evolved with the development of the computational domain typically known as machine learning. In early studies, the most commonly utilised methods where parametric, like the maximum likelihood classifiers or unsupervised like K-means and ISODATA [100]. In the 1990s, non-parametric learning algorithms became an option for remote sensing users and the number of studies utilising decision trees and neural networks increased [36]. These approaches do not make any assumption about data distribution, unlike the parametric classifiers, where it is required the data to be normally distributed. Recent studies where different data sources are utilised also favoured non-parametric methodologies like the support vector machines and the random forest algorithm [36]. The utilisation of these more sophisticated methods was facilitated by the adoption of free software solutions and open source environments such R and Python.

Supervised methods require the user to collect a set of pixels of known class. In general, a training data set is such that:

1. It should have a sufficient number of independent training data points for each land-cover class [112];

2. It should be exhaustive, in the sense that it should contain samples describing all classes and preferably for all apparent classes present in the image [122];

Condition 1 is necessary for a reliable estimation of the classifier parameters. In other words, insufficient independent training data points may result in the Hughes phenomenon [112]. This is an important point specially when dealing with hyperspectral data [113]. However, for the purpose of this research, focus is on the second point, due to its importance in the formulation of the specific class mapping problem.

Note the phrasing of condition 2: all classes of interest and preferably for all apparent classes. In other words, beside including all land cover classes, the training data set should also include patterns resulted from intraclass variability. For example, their could be different types of water or different types of pasture in the same image frame. Failure to fulfil this condition, may result in an unrepresentative training sets [23].

To fulfil this second condition, remote sensing projects require time and expert image analyst to collect training data, ancillary data and sometimes field work. This may represent too much effort that organisations and institutions may not be able to provide. Additionally, the identification of the apparent classes is complicated, since the definition of the land cover classes and the spectral pattern are often difficult to an human operator to consistently and accurately identify. This becomes even more obvious if the image sampling is executed by more than one image analyst. Nevertheless, collecting training data points is a must for remotely sensed data analysis with supervised learning algorithms.

Supervised methods allow users to tailor the mapping process to suit their own applications, since in each instance the needs can be quite specific [55]. Indeed, users are often not interested in all classes present in the study area but just on a small subset [71]. This is evident in studies where major land cover transformations are object of study, such as deforestation [104] and urbanisation [24, 39], or where specific classes are of interest, such as abandoned agriculture [3, 99, 119], specific tree species [5, 54, 59, 68, 131], wetland species and crops [77, 87], mangrove ecosystems [74, 90, 133, 140], to name just a few. In such applications, the use of multi-class image classification methods can be inappropriate [45, 55, 103]. In other words, building a classifier capable of handling effectively only a subset of classes may be a better alternative. In this dissertation, that is called specific class mapping.

The rest of this chapter is organised in the following way: in the section 2.2 a brief overview of the topic is presented; for example the term "specific class mapping"is defined and it is explained why conventional supervised methods may yield suboptimal results. In the section 2.3 it is presented an analysis of the previous works where the specific class mapping was the central issue. The section is mostly descriptive and tries to cover all the major developments and applications. In section 2.4 the synthesis is presented and the most important points learnt from analysis of previous works are highlighted and the most relevant concepts are summarised and categorised. This chapter ends with the conclusion.

## 2.2 Specific class mapping

The term "specific class mapping"refers to the process of non-exhaustive supervised classification of a given region of interest. In other words, it refers the mapping process of only a subset of classes present in that region [55]. The term, however, is not consensual and, thus, often the problem and the process are not mention directly in research papers. One can find researchers using the phrases like "non-exhaustively defined set of classes"[44, 56], "supervised classification without an exhaustively defined set of classes"[46], or more closely "mapping a specific class"[47], or less frequently "targeted classification"[103]. In some cases, the title briefly suggests that the base problem is that of classifying a specific class or classes of interest, such in "Operational tree species mapping in a diverse tropical forest with airborne imaging spectroscopy"[6], "Tree Species Discrimination in Tropical Forests Using Airborne Imaging Spectroscopy"[40], "Tree species classification in the Southern Alps"[28], to list just a few. In this thesis, the term "specific class mapping"was preferred for two reasons: first, although it was not the term utilised in the first study, like [88], it was used in early remotely sensed studies tackling the problem, such as [13, 55, 124, 125]. And second, it clearly associates the process to the land cover mapping process.

In this way, one can defined the problem of specific class mapping in the following way: the objective of specific class mapping is to map a subset (one or more) of land

cover classes, named as classes of interest, without considering other classes present in the region to be mapped [103]. Thus the specific class mapping process ideally requires the user to exclusively collect data points from those classes of interest, which represents a non-exhaustive sample of the class composition of the mapping region [44, 55]. The technical requirement here is to develop processes that are capable of mapping specific classes without significant decrease in classification accuracy when compared with conventional supervised classification methods. Note that the classes of no interest can be ignored by the analysts for different reasons; for example, the analysts may ignore them to save time and sampling effort or these classes may even be unknown to them [103]. That is common to many operational scenarios where gathering an exhaustive data set for the mapping region is difficult, costly, or not even impossible.

But why are conventional multi-class supervised methods not suitable for specific class mapping? There are essentially two reasons for that. One problem in using multi-class approach for specific class mapping is the exhaustive training requirement. That is the training sample has to contain all classes present in the study area regardless of its importance for the analysis [71]. If the training set is not exhaustive, the classifier, will commit pixels of untrained classes into the set of classes in which the classifier was trained [52, 103, 125]. This may lead to classification errors that are not identified in the accuracy assessment process [44, 55]. For example, areas of untrained forest may consistently be committed into a particular crop or shrub. As a result, the resulted map overestimates the extension of that crop or shrub. Thus the analyst has to ensure all classes present in the study area are sampled to avoid such errors. This from the operational point of view represents additional sampling effort that could be reallocated to other processing steps, such as tuning the learning algorithm or sampling data for accuracy assessment.

Another concern is that a multi-class classifier is often developed to maximise classification accuracy over all land cover classes present in the training rather than focus on the specific classes of interest [55, 88]. That is, the classification algorithm seeks to output a classifier with maximum overall classification accuracy, measured over all classes. However, the classes of interest are typically only a small part of the overall training set, and thus the analysis may not be optimal for the discrimination of these specific classes. Indeed, when fixing the free-parameters of classification algorithm in a multi-class problem, the analyst searches for the particular parameterisation that yields the lowest classification error in a series of testing trials, for example 10-fold cross-validation [9]. At each trial, a classifier is trained and its overall accuracy is assessed irrespective of classes under analysis. It is possible then to a particular parameterisation to yield an accurate classifier that omits part or all classes of interests. For example, if the classes of interest represent only 10% of training set and a particular parameterisation yields a classifier that omits these 10% but accurately classifies the remain 90%, the overall accuracy associated to this classifier will be 90% although the

important areas are completely omitted. This parameterisation, then, may be regarded as a good parameterisation by the fine-tuning process for that particular classification problem. Thus, the classifier derived from this particular parameterisation is an accurate classifier for the majority of the classes present in the classification space but inaccurate in the most relevant classes. Conversely, if a particular parameterisation renders a classifier that accurately discriminates the classes of interest (10%) but misclassifies all others (90%), this classifier will score low and thus its parameterisation is likely to be rejected.

What is proposed with specific class mapping is to apply the Vapnik's principle that is: *when solving a problem of interest avoid solving a more general problem as intermediate step* [139]. In other words, if focus is on a particular subset of classes present in the area of study, the analysis should focus on the discrimination of those classes regardless of other classes present in the classification space. Indeed, if one is interested on just a subset of classes, misclassifications between classes of no interest are of no importance [51]. Therefore, when interest is on a subset of classes present in the study area, it may be preferable to follow an alternative approach to the conventional multi-class supervised classification method [125]. In other words, building a classifier capable of handling effectively only a subset of classes may be a better alternative for specific class mapping.

## 2.3 Analysis of previous works

The concept of specific class mapping can be dated back to 1995 with [88], for is cited in several studies concerning specific class mapping. Although, in [88] the author does not formulate the need for a more specific driven mapping analysis in the same way as in the previous section, he argues that the producer (who carries out the mapping procedure) ultimately has to optimise the mapping process in terms of the requirements of the user (who commissions the map). This is because the most important aspect of a map is its utility, which depends on two factors [88]: the usefulness of a particular class and the quality with which that class was mapped. The responsibility for ensuring that the defined classes are useful lies with the user, which defines the product requirements. The responsibility for ensuring that that class is well mapped, on the other hand, lies with the producer. And here is the key point motivating specific class mapping. In other words, specific needs require specific maps that require specific mapping processes. The entire work in [88] can then be summarised in three points:

1. Users who require a map to answer a particular question will be interested in the accuracy of the classes that allow them to answer that question. For example, when study mangrove systems, the dynamics of particular crops, urban and semi-natural features may be secondary [90].

2. The accuracy metrics concerning those classes are then more important to the producer than others. For example, if users are interested in the class mangrove, the accuracy of the class mangrove (say the user accuracy and the producer accuracy) that is of importance. In this way, a user may prefer a map with lower overall accuracy but high user and producer accuracy for the mangrove class, since errors between secondary classes are irrelevant.

3. Thus, it is necessary to drive the classification process "to produce a map for a particular user in which the components of accuracy that concern him or her are optimized at the expense of components of accuracy that are not of interest"[88].

In [88] it is not argued that a specific driven approach is a must if the conditions do not allow the analyst to define an exhaustive training set, but simply that better and useful maps can be produced if the analysts move their classification process to a more user specific driven process. That is, this study main contribution is to contend that there are various different errors which characterise the accuracy of a map and that a particular user will be concerned only with a subset of these. And the classification process has to take that into account. Thus a classifier optimised on one or a few components of accuracy, for example overall accuracy, will not necessarily yield the best possible map for any user.

Perhaps the first reference about non-exhaustively defined training data sets or untrained classes in remote sensing can be found in [58]. Here a method was developed to provide an estimation for *a posteriori* probability vectors and the estimation of the *prior* probability of classes. The idea of estimating statistical properties from the training data to later infer the probability of an unseen pixel to belong to a particular class was followed by other researchers, such as [1, 102, 103].

In [48] and [45], the main concern were the effects of non-exhaustive training data sets in the classification and how one could minimise it. The failure to satisfy the exhaustive composition of the training set was investigated and assessed with reference to hard and soft land cover classifications using neural network. [48] in particular evaluates the use of relative and absolute class membership strength under the presence of untrained classes, fuzzy $c$-means and possibilistic $c$-means respectively, to evaluate the sub-pixel land cover class composition. Here three classes were of interest, asphalt, grass and tree, using Daedalus 1268 sensor with eleven spectral wave-bands and 1.5 m geometric resolution. The study main conclusion is that with the frequent presence of untrained classes in the image, it may be inappropriate the use of methods that base their decision on relative class membership, such as the fuzzy $c$-means, for the estimation of sub-pixel class composition. In contrast, methods that provided absolute membership metrics may be better suited to derived the class composition of sub-pixels, such as the possibilistic $c$-means. This is because relative methods compute class membership with respect to all defined classes. As result, the

magnitude of class membership are large or small according to the classes that have been defined in the training stage.

[45] explores how supervised image classification methods, like the multi-layer perceptron and radial-basis neural networks, behave with and without the presence of an exhaustively defined set of classes. Since these two supervised methods are based on one key assumption, that the set of classes has been defined exhaustively, if this assumption is unsatisfied, pixels of an untrained class will commissioned into the set of trained classes. As result some classes will be over-predicted, leading to an over-estimation of their true extension. Here a data set derived from airborne thematic mapper (ATM) imagery with a spatial resolution of 5 m acquired with a Daedalus 1268 sensor was utilised. A set of six crops were of interest, namely sugar beet, carrots, wheat, barley, grass and potatoes. Although the classifications derived each method differed in the pattern of class allocation, they provided the same high classification accuracy > 85%. However, when the learning algorithms were trained with a rejection option, that is they were allowed to classify a pixel or reject to classify by labelling the pixel as unknown, the classification accuracy decrease significantly, roughly about 12.5%. This is a relevant result and it shows the necessity to identify training data at least for all classes of interest and preferably for all apparent classes in the segment of image to be analysed [122].

In [13] explicit interest in focusing the mapping process in a particular class of interest is mentioned. This differ from previous works in the sense that the authors followed [88] advice to adapt the mapping process to the user needs. Since most supervised learning algorithms aim to maximise the overall probability that a pixel is allocated to a class correctly, rather than focus on the accuracy with which the specific class of actual interest is classified, may lead to an sub-optimal classifier to discriminate the class of interest. Indeed, overall accuracy is just one component of classification accuracy that may not even be the most useful for the user [88]. Here a Landsat 7 ETM+ image was used and the land cover class fen was considered the class of interest as opposed to the other seven, salt marsh, grazing marsh, agriculture, forest, urban, sand and water. These seven classes constituted the others class. Two binary classifiers, trained with class of interest and others data set, were compared with a conventional classifier, trained with all the eight classes. The binary approach were significantly more accurate that the multi-class conventional approach, indicating that a class specific driven classification approach may be more suitable to discriminate particular classes of interest than conventional multi-class way. The authors also indicate that the satisfaction of the assumption of an exhaustively defined set of classes requires that much effort, since part of the process is reserved to classes of little, if any, direct interest; and thus savings in training could be achieved by focusing on the specific class of interest.

Indeed, the idea of saving resources in the training stage were taken even further in [55], where training set size requirement for the classification of a specific class was

explored. The authors explored four different approaches to reduce training set size to assess the minimal requirement to an accurate classification: intelligent selection, selective class exclusion, acceptance of imprecise descriptions for spectrally distinct classes, and the adoption of a one-class classifier. These approaches were compared with the often suggested heuristic termed $30p$. In other words, each class should be sampled such that there is at least $30 \times p$ where $p$ is the number of bands. The study site was composed by cotton, local rice, basmati rice, sand and built-up land, but only cotton was of interest. Experiments were conducted with data acquired from the Indian Remote Sensing Satellite LISS-III sensor with three spectral wavebands and 20 m resolution. The results indicate that all approaches were capable to reduce the training requirement and yield accurate classifications comparable to that of the conventional widely used heuristics $30p$ without significant impact on the discrimination of the class of interest.

In the sequence of [13], [47] explore the use of ensemble methods to classify a specific class of interest. The use of an ensemble could be useful since in many classifiers, such as support vector machines and neural networks, the need to fix free parameters can be difficult and time consuming. The use of multiple classifiers could in principle mitigate that need [148]. The study was carried out with various binary classifiers used to discriminate a specific class of interest from all others. A Landsat ETM+ image of the test site was acquired and training set for the two classes, fenland and "others", were extracted from the imagery using stratified random sampling, as in [13]. The ensemble was composed by five classifiers, commonly utilised by remote sensing analysts, discriminant analysis, decision trees, support vector machines, multi-layer perceptron, and radial basis function neural network. The outputs of the classifiers were combined using a simple voting procedure to determine class allocation. The accuracy of this ensemble was 95.6%, marginally better, but statistically insignificantly, less accurate than the most accurate individual classifier. However, it is difficult to specify the most appropriate classifier in advance, and thus the ensemble approach may represent an operational solution for specific class mapping.

In [124] and [125], it is suggested that a single class classification approach could be appropriate if interest focuses on a specific class. The authors illustrated this with the classification of fenland from Landsat 7 ETM+ imagery and evaluated a range of one-class classifiers, with particular attention to the support-vector data description algorithm. The overall accuracy yield by the benchmark analysis was 68.8%. This represents in general a low accurate map, however as supported by [88], analysis should focus on the classes that best suites the user needs. Here the class is fenland. This was classified with a commission error of 10.0% and omission error of 28.0%. The accuracy of this classification is below used map accuracy targets and the map derived from it may be viewed as being inadequate for use in the monitoring of the class of interest [125]. In particular, the benchmark classification shows a large degree of omission error and, as consequence, the extent of the class of interest was substantially

underestimated. This is relevant since indicates that a map derived from a general mapping procedure use for specific purposes (here fenland monitoring) may not be optimal for those purposes. The one-class approach yielded a classification that was significantly more accurate than that from the benchmark with commission errors of 2.5% and omission errors of 6.4% for the class of interest. The main highlight is that very accurate land-cover maps of the class of interest can be produced with effort and resources directed on the class of interest

In the same year, [108] explored the same algorithm, support vector data description algorithm, to classify a specific class with hyperspectral data. This method was compared with other classification approaches in the discrimination of the class of interest. The experimental results, obtained on different kinds of data (synthetic, hyperspectral, and multisensor images), indicate the effectiveness of the technique to classify remotely sensed data in the presence of incomplete training data. In particular, the support vector data description provided good results particularly on the multisensor data set, with significant improvement of the classification accuracy with respect to other well-established one-class approaches.

The exploration of single class classifiers for specific class mapping was followed by a string of works. In [109], two semisupervised one-class support vector machine approaches were presented for remote sensing data classification to mitigate the need to heavy fine tuning. Indeed, typically fine tuning is hard because the free parameters of the model need to be finely adjusted, but there is not clear indication on how to correctly perform it. Here the authors suggest the use of unlabelled data to overcome that requirement by modelling the data marginal distribution with the graph Laplacian built with both labeled and unlabelled data points. The testing trials were conducted with hyperspectral data. In [94] the authors follow the a similar idea of utilising unlabelled data to improve the learning process of a specific class of interest. A classifier is developed with training data set from the class of interest and unlabelled data, and estimates the probability that a positive training data point has been labeled, and generates binary predictions for testing pixels using an adjusted threshold. The experiments where conducted using aerial imagery and indicated that the new algorithm provides high classification accuracy. A similar method is presented in [29] to exploit the information contained in the unlabelled data points to improve the classification accuracy of a standard binary support vector machines. In [15], the mean and product combination rules were applied to the probabilistic outputs generated by single class classifiers, and their performance were evaluated in a urban monitoring application with multi-sensor (optical and SAR) data and multi-source (spectral and contextual) features were used. The results obtained by the ensemble show an improvement in the accuracy despise the high dimensional classification space. And in [101], authors suggest a user-oriented one-class classification strategy, based on the visualisation and interpretation of the classifier outcomes during the data processing. They show that

careful interpretation of the diagnostic plots fosters the understanding of the classification outcome, such as the class separability and suitability of a particular threshold, for the classification of a particular class of interest.

In applicational studies, where the object of analysis is not the method itself, but rather some application research question, the use of specific class mapping approaches is not frequent. In the majority of the cases, users commonly apply a conventional multi-class approach. For example, in [87], interest was on four invasive plant species by a a map with fourteen land cover classes was produced; in [74], interest was on one class, mangrove forest, but twelve were utilised to obtain the land cover map; in [99], interest was on abandoned agricultural land (one class) but five classes were utilised to compute the land cover map random forest and support vector machines; and in [119], where interest was also on abandoned agricultural land, the thematic classification was done using twelve land cover classes using support vector machines.

Although rare, there are instances of applicational studies aware of specific class mapping. For example, in [54] where specific tree species in ancient semi-natural woodland, sycamore, fir, oak, ash and line, are the classes of interest and a two-phase classification approach was adopted to map specific species from aerial sensor imagery. First the *prior* probabilities were adjusted manually, followed by a sequence of binary classifications in series, like a decision-tree classifier. In particular, following [88] indication that the accuracy of the classes that allow them to answer that question are of more importance, the optimisation was performed by adjusting the *prior* probabilities of class membership to ensure that no pixels of the selected class were omitted from it in the correct classification. In other words, the process was undertaken in a manner that sought to optimise the producer's accuracy of the class of interest. A consequence this approach is that the selected class of interest may commission pixels that actually belong to other classes and hence overestimate its abundance. In the second phase, the adopted process sought to remove these commissioned pixels through a series of binary classifications applied to the sample of cases predicted to belong to the class of interest from the first phase of the analysis, to gradually subdivide the data until the class of interest have been identified. The first phase of this process is indicates a relevant point for specific class mapping, that is the classification optimisation process should be guided by class specific metrics, such as the producer's accuracy, to drive the classification process to solve the specific problem instead of global classification problem. In [5] the same methodology was utilised.

In [6, 28, 59] the binarisation process "classes of interest vs. others" as in [13] was adopted. [59] in particular, has utilised it to discriminate tree species. Here twenty-one tree species were of interest and the remain twenty-four were combined in one single class others with Airborne Taxonomic Mapping Systems sensor package. The authors used a support vector machine classifier for its ability to handle high dimensional data with small training sets, and the one-vs-one approach was adopted generating a total of 210 binary classifiers, one for each pair of classes of interest where one of the classes

is the nominal class others. Evaluation metrics where specific, similarly to what was done in [54]. Here precision and recall were utilised. Precision and recall are informative metrics for multi-class models, that were combined in the F-score metric. However, is not clear if the free parameter determination was done utilising this metric or the conventional classification accuracy. Nevertheless, the authors accurately identified the effects of data imbalance that may occur when utilising the approach indicated in [13]. To overcome data imbalance problems the authors adopted an weighted approach, by implementing them when running the support vector algorithm after the training data had been split into the training and testing groups. The authors quantified the effect of class imbalance on model accuracy, and verified a trend where species with more samples were consistently over predicted while species with fewer samples were under predicted. Additionally it was verified that standardising the sample size reduced the classifier accuracy but also reduced the level to which the classes were over- or under-predicted.

In [28], in particular, four data sets were utilised to discriminate specific tree species utilising hyperspectral data, multispectral and LiDAR. In the first data set, seven classes were of interest and the eighth consists in the others class. In the second data set, five classes were of interest and the sixth the nominal class. In the third data set, there were two classes of interest and the nominal class. And in the final data set a simple binary classification problem, interest-vs-non-interest was utilised. The support vector machines and random forest were compared in each data set. Both methods yield high accuracies > 85%, but indicating that the support vector machine outperformed the random forest in each data set, regardless of the type of data being used. Unlike, [59], there is no clear indication if data imbalance related problems were detected and, if so, how they were tackled.

In [6], the process of specific class mapping is termed focal species mapping, where focal classes are the classes of interest and non-focal are the classes of no interest. Here three classes were of interest and the remotely sensed data consisted in data acquired from the Carnegie Airborne Observatory Airborne Taxonomic Mapping System and LiDAR. Several approaches were compared, the binary approach and the biased approach. One-vs-rest binary support vector machine and biased support vector machine were found to outperform the standard support vector machine for remote species identification. Additionally, they found that binary support vector machine provided greater sensitivity but slightly lower specificity than biased support vector machine. And in particularly, the precision of binary support vector machine would likely improve, if more training data were gathered for the classes of no interest. However, pursuing this track can be costly, specially with high tree species diversity it becomes very costly to comprehensively sample the non-focal tree species. The Biased support vector machine is advantageous here, since it comprehensively samples the spectra of all vegetation present in the region of interest and uses this information to constrain the focal classes.

## 2.4 Synthesis of the previous works

The analysis to the literature showed that there are essentially two general ways to implement such a specific class classification process. One is to decompose the multi-class problem in a series of binary classification problems to separate the classes of interest from all the rest; and the second is to adopt one-class classifiers that explicitly focus all processing in one class of interest. The first is the binary, or binarisation, approach, where the problem is decompose in a series of binary decisions; and the second is the single-class approach, where single-class classifiers are utilised.

The binary approaches tend to define simpler decision boundaries which reduce the competence areas of each classifier producing locally specialised models [82]. From these small binary problems, the original multi-class problem can be solved using combining strategies such as one-vs-one and one-vs-all [129]. Although studies have shown that binary decomposition performs well in most multi-class problems, it has nevertheless limitations such as being dependent of the combination method and being susceptible to data imbalance and sparse distributions [81]. From the operational point of view, the binary classification approach still requires the sampling of land cover classes of no interest at training collection stage, since it is necessary to sample the classification space outside the classes of interest. And like the multi-class supervised method, if this space is ill-sampled, that is some classes are omitted from the training or under-sampled, it is possible for a classifier to commit some of those areas into a class of interest, over-estimating the true extension of these important classes.

In the binarisation category, one can also include the semi-supervised approaches, since this type of classifier typically utilises a binary classifier or binary decomposition approach. Although, like the single-class approaches, only the classes of interest are sampled by the analyst. But, these tend to be computationally more expensive than the single-class methods, since additional information is extracted from an typically very large number of unlabelled samples [95]. However, they can be much more accurate, specially if significant spectral ambiguities between the classes of interest and the negative class exist. In such cases, a single-class approach may not perform as accurate as a semi-supervised approach or binary approach [69, 93].

The binarisation approaches require a form combination scheme. The three most common schemes are the one-vs-rest (also known as one-vs-all), the one-vs-one and the decision directed a cyclic graph. The most common form of the three is the one-vs-rest, use for example in [13]. But the other schemes have been used also, for example [59] used the one-vs-one, and [54] a form of decision directed a cyclic graph. Which of these schemes is the best is an open question [9, 30, 123], and one can find different studies suggesting different answers [57, 138]. For example, one-vs-rest leads to less classifiers, but may suffer from imbalance related issues. One-vs-one, on the other hand, break the problem into simpler problems but require much more classifiers. And decision directed a cyclic graph is the most flexible of the three, but may become too complex.

Nevertheless, the literature review indicates that the binarisation approaches are the most commonly used by non-expert researchers. That is, in applicational studies users tend to use this approach, either by adopting a one-vs-all approach, like [28, 54, 59], or a semi-supervised approach, like in [6].

With the single-class approach, on the other hand, the user adopts a one-class learning algorithm to develop a classifier to identify a single class of interest [64, 73]. In this approach only training data belonging to the class of interest is utilised to develop the classifier, which is its most attractive feature in terms of sampling effort and resources on the class of interest. In this category are also included the density estimation based methods, like those present in [1, 58, 102, 103].

The single-class classifier may not always be the best approach, since only data about one class is available and thus only one side of the discriminative boundary can be determined [137]. It can then be difficult to determine how tightly the boundary should fit in all directions around the data in feature space. To overcome this difficulty some one-class classifiers, like the support vector data description, assume that the non-interest class has a particular distribution around the class of interest [137]. When the true distribution deviates from the assumption, the method may underperform. That deviation however can only be assessed with training points outside of the class of interest [136].

From literature, the one-class support vector machine and the support vector data description are well stablished single-class classifiers for remotely sensed data classification in methodological studies. As with the conventional support vector machines, free parameters have to be determined. These typically are the kernel parameters and a regularisation parameter, also known as penalisation cost. In practice, the penalisation parameter has been defined via the omission/false negative rate, like in [109]. Thus the user has to specify the percentage of the positive training data to be rejected by the classifier. This variable needs to be fix carefully to ensure an accurate classification result. Values such as 1% or 5% can be suitable when the positive class is well separable [92]. If not, the classifier may induce a high commission/false positive rate when a significant class overlap exists. And as consequence the classifier may over-estimate the area of the class of interest.

Literature also indicates that specific metrics should be utilised as suggested by [88]. These specific metrics constrain a particular step of the classification process, like the determination of the free parameters, to the discrimination of the classes of interest. In [59], the F-score (the harmonic mean between recall and precision) was utilised but other metrics could be used. For example, another metric often used is the geometric mean of sensitivity and specificity [145] which is defined as the square root of the product between sensitivity and specificity. Sensitivity and specificity are two metric utilised in binary classification. Sensitivity, also known as recall, is also utilised in the F-score. This metric represents the producer's accuracy of the class of interest. Specificity, on the other hand, represents the producer's accuracy of the

class of no interest. In contrast with the precision utilised in the F-score, that is the user's accuracy of the class of interest, this does not dependent of the geographical extension of the class of interest [111]. In other words, precision as typically utilised in binary classification, when applied to image classification, represents a biased estimator. There are no references in the literature discussing this point. Thus it is not clear if there are issues in utilising F-score for geographical data. But if the geometric mean is used instead, that possible objection does not apply.

A final point of importance derived from the analysis of previous studies where specific class mapping was the central problem is the bias that the classification process may suffer. This bias can be categorised in two types: intended and unintended. An example of intended bias is in [54], where the optimisation process was purposely manipulated to created a bias to the class of interest. That is, the process was minimising the omission errors and, as result, the classification process would commission cases that actually belong to other classes (classes of no interest). The consequence is that the class of interest end up being overestimated. In [54] the effect is then corrected in further analysis. An example of unintended bias is in [59] where the classification scheme one-vs-rest led to imbalanced issues and was underestimating the smaller classes and overestimating the larger ones. Since the classes of interest are typically small [88], and that is particularly evident when one-vs-rest is utilised, these may end up being underestimated. This bias is then against the classes of interest, and can be harmful if user is not aware of it.

## 2.5 Conclusion

In this chapter, the term specific class mapping was defined as consisting in the process of producing a non-exhaustive thematic map of a particular region. This contrasts with the conventional supervised classification that by design output a complete characterisation of the mapping region. Defining an exhaustive training can be problematic for time consuming and economical reasons but also because the intrinsic difficulty of the task. Thus conventional supervised methods may not be the best approach in these cases, because an exhaustive class composition at the training stage may not be possible or desirable. Additionally, the conventional supervised methods may not be fine-tuned for the optimal discrimination of the classes of interest, since these methods are effectively solving a larger problem. From the analysis to the literature, it was possible to organise the different approaches in two boarder categories: the binarisation methods and the single-class methods. Both categories present advantages and disadvantages. For example, the binarisation approaches have access to a complete information, that is information about the classes of interest and information about the classes of no interest, but depend on the combination schemes. The single-class approaches, on the other hand, require exclusively training data from the classes of interest and nothing else, but the fixing the free parameters can be a difficult task

that may lead to classifiers with high number of false positives. It was identified that specific accuracy metrics may improve the process, for example when determining free parameters. Examples of those metrics are the F-score and the geometric mean between sensitivity and specificity. Finally, it was identified that bias in the process may occur. These can be to or against the class of interest. If to, this may lead to an overestimation of the class of interest. If against, this may lead to a underestimation. Bias can also be intended or not. If not, these bias can be harmful if the user is not aware of them.

Chapter 3

# Methodological background

**Abstract** In this chapter the concepts and methods that support the research in the core chapters 4, 5, and 6 are presented. The purpose here is not so much to present an exhaustive account of all topics but rather to clarify and highlight details that may not be clear or were briefly discussed in the next chapters. In this sense, some details, like the problem of data imbalance, cost-sensitive learning and the adaptation of support vector machines to this type of learning, are not discussed here because they were comprehensively elaborated in further chapters. The chapter starts with the support vector machines classifier, in particular with the soft margin formulation. Here is also discussed, the kernel trick, the adaptation of the support vector machines to multi-class classification problems, one-class support vector machines and model selection. The chapter ends with a discussion about accuracy metrics for binary classification assessment and classification comparison.

## 3.1 Introduction

The Support Vector Machines (SVM) is a supervised non-parametric binary learning algorithm, which entails that no assumption is made on the underlying data distribution [139]. This is particularly important in remotely sensed data analysis since in general it is unlikely to know the data distribution of land cover classes beforehand. Another attractive feature of SVM is its relatively low training requirements. In other words, a limited quantity of training data points is enough for SVM to yield an accurate classifier. Indeed, some studies, for example [51, 52], indicate that only a quarter of the recommended number of training data is enough to produce good results.

However, SVM are usually difficult to fine tune [17], which may take considerable time. This is usually done by cross-validation but there is no guarantee that the result

is indeed the best possible parameterisation [9]. Another limitation of SVM is its inability to deal with multi-class classification problems directly. Although there are SVM formulations that solve multi-class problems these methods have not been widely adopted in remote sensing community. Typically SVM solve multi-class problems by decomposing the problem in smaller binary problems. Some studies indicate that these simple approaches are more suitable for practical use than the more sophisticated multi-class approaches [91] but there is no definitive answer regarding which one is the best. The SVM principles can also be adapted to solve single-class problems by rewriting the SVM optimisation problem, however the main concepts, like support vectors, are maintained.

Since one overarching theme of this dissertation is binary classification, this chapter also tackles specific accuracy metrics to assess the performance of binary classifiers and how to interpret them. In particular, sensitivity and specificity since these metrics are central for the analysis of classifiers in further chapters. Another topic, related with accuracy metrics, is the classification comparison. It is explained why conventional statistical hypothesis testing are not always suitable for the purpose of comparison and it is presented an alternative.

## 3.2 Support vector machines

### 3.2.1 Soft margin support vector machines

The soft margin SVM was introduced by [11] and represents an improvement over the hard margin SVM introduced years before [30]. The soft margin SVM aim is to induce a linear classifier that "best" discriminates two classes not necessarily separable. Best here is defined as the linear classifier with maximum margin [86], where margin is defined as the distance from a training data point to the decision boundary defined by the classifier [138]. However, since the two classes are not necessarily (linearly) separable, the maximisation problem has to involve trade-off metrics, unlike the hard margin SVM [139].

Figure 3.1 represents two hypothetical classes. They are not linearly separable and thus some points will be misclassified by the linear classifier. The margin, as with the hard margin classifier, is maximised by minimising the quantity $\frac{1}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w}$ [30, 138], where $\boldsymbol{w}$ is the normal vector defining the discriminant plane. However, an additional condition is necessary to accommodate the misclassified data points. Commonly the soft margin SVM utilises one of these two: linear loss function (also known as hinge loss), $\sum_i \xi_i$, and the quadratic loss function, $\sum_i \xi_i^2$ [30]. The variables $\xi_i$ are called slack variables and represent the error committed in the data point $\boldsymbol{x}_i$ by missing the margin. The discussion will focus on soft margin SVM with linear loss function, since this is the most common implementation [139]. The optimisation problem is then formulated in the following way [86]:

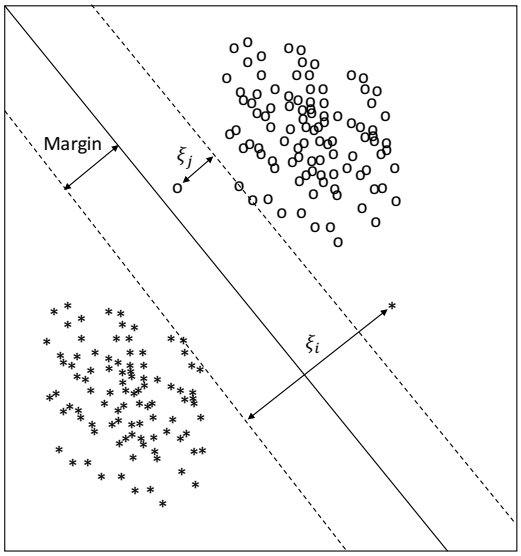



Figure 3.1: Soft margin support vector machine. The margin is maximum distance between the discriminate plane and the training data points. However, some points not correctly discriminated.

$$min_{w,b,\xi} \ \frac{1}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w} + C\sum_i \xi_i \tag{3.1}$$

subject to $y_i(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + b) \geq 1 - \xi_i$ and $\xi_i \geq 0$ for all training data points $(\boldsymbol{x}_i, y_i)$. The optimisation problem is then composed by two terms: the $\frac{1}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w}$ that maximises the margin and $C\sum_i \xi_i$ (the regularisation term) that minimises the number of misclassified data points. The parameter $C$ is called the penalisation cost (for misclassifying a point) and controls how severe should the optimisation process be with misclassifications. In other words, for large $C$ the number of misclassifications is small and, as consequence, the margin is narrow. This leaves less room to accommodate atypical data points. On the other hand, small values of $C$, allows the optimisation process to accept solutions that misclassify some points, which results in a larger margin. The topic of how to define this parameter is discussed in section 3.2.5.

The soft margin SVM optimisation problem can be solved with quadratic programming by reducing this formulation (typically termed primal form) in its dual form [86]. This is done by composing the Lagrangian objective function [43]. Concretely, the Lagrangian objective function of the problem 3.1 is:

$$L = \frac{1}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w} + C\sum_i \xi_i - \sum_i \alpha_i[y_i(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + b) - 1 + \xi_i] - \sum_i \beta_i\xi_i \tag{3.2}$$

Applying the Karush-Kuhn-Tucker (KKT) conditions [43] it is possible to infer:

$$\nabla_{\boldsymbol{w}} L = w - \sum_i \alpha_i y_i x_i = 0 \Rightarrow \boldsymbol{w} = \sum_i \alpha_i y_i \boldsymbol{x}_i \tag{3.3}$$

$$\nabla_b L = -\sum_i \alpha_i y_i = 0 \Rightarrow \sum_i \alpha_i y_i = 0 \tag{3.4}$$

$$\nabla_{\xi_i} L = C - \alpha_i - \beta_i \Rightarrow C = \alpha_i + \beta_i \tag{3.5}$$

$$\forall\ i\ \alpha_i[y_i(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + b) - 1 + \xi_i] = 0 \Rightarrow \alpha_i = 0 \vee\ y_i(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + b) = 1 - \xi_i \tag{3.6}$$

$$\forall\ i\ \beta_i \xi_i = 0 \Rightarrow \beta_i = 0 \vee \xi_i = 0 \tag{3.7}$$

The first observation is that the normal vector $\boldsymbol{w}$ is a linear combination of data points and its Lagrange multipliers (equation 3.3). This is relevant since some of the Lagrange multipliers have to be zero (equation 3.6). The data points with non-null Lagrange multipliers are called support vectors, since only these points are relevant for the classification [138]. Effectively, if $\alpha_i \neq 0$, then $y_i(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + b) = 1 - \xi_i$. Since $\xi_i \geq 0$, $y_i(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + b) = 1$ if $\xi_i = 0$ or $y_i(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + b) < 1$ if $\xi_i > 0$. In the first case, the point is on the margin and in the second case the point is on the wrong side of the plane. Thus the support vectors are either on the margin or on the wrong side of the plane and thus misclassified.

The dual form of the problem is obtained by inserting the definition of $w$ in the Lagrangian (3.2). This yields:

$$max_{\boldsymbol{\alpha}}\ \ L = \sum_i \alpha_i - \frac{1}{2}\sum_{i,j} \alpha_i \alpha_j y_i y_j \boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{x}_j \tag{3.8}$$

subject $0 \leq \alpha_i \leq C$ and $\sum_i \alpha_i y_i = 0$ for all training data points. This problem is quadratic and convex and thus the solution is unique [43]. It is important to note here that, from the KKT conditions, the Lagrange multiplier of the misclassified support vectors, those with $\xi_i > 0$, is such that $\alpha_i = C$. Effectively, these support vectors have the largest possible contribution for the solution of $\boldsymbol{w}$ [138].

The solution of the dual problem can be used directly to determine the classification rule by the soft margin SVM classifier:

$$f(\boldsymbol{x}) = sign(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b) = sign(\sum_i \alpha_i y_i \boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{x} + b) \tag{3.9}$$

where only the support vectors are utilised in the computation, and $sign(x) = 1$ if $x \geq 0$, and $-1$ otherwise. The variable $b$ can be determined by averaging all possible $b$ with $b = y_i - \sum_i \alpha_i y_i \boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{x}$ [30].

### 3.2.2 Introducing the kernel trick

Kernel methods are machine learning algorithms that utilise a non-linear function from the input space to a higher-dimensional space, where linear separation is possible. This is called the kernel trick. This transformed space can indeed be very large.

Transformed spaces with $10^{15}$ is not uncommon in practice [86]. However, the computational cost to generate such space and to define a linear classifier is negligible. That is what makes the kernel methods such a powerful machine learning tool. The mathematics behind the kernel function theory is dense and long. For brevity sake, only the most important elements will the discussed here.

What is then a kernel? A kernel function is a function $K : R^d \times R^d \rightarrow R$ such that for any two data points $\boldsymbol{x}$ and $\boldsymbol{x}'$ of $R^d$, $K(\boldsymbol{x},\boldsymbol{x}') = \boldsymbol{\phi}^T(\boldsymbol{x})\boldsymbol{\phi}(\boldsymbol{x})$ where $\boldsymbol{\phi}$ is a non-linear function. This non-linear function $\boldsymbol{\phi}$ is such that $\boldsymbol{\phi} : R^d \rightarrow R^N$. Thus effectively what $\boldsymbol{\phi}$ is doing is to project a single point from the classification space to a very high-dimensional space. The kernel function $K$ is then only the inner-product of two projected data points. Note that since the inner-product of two vectors is often interpreted as a similarity measure, the kernel function is measuring the similarity between these two projected data points. In other words, provided two data points in the feature space, with the kernel function $K$ is possible to assess how similar are they are in the projected space. This is important since the SVM learning algorithm tries to find a small subset of training data points that best discriminant both classes. If two points are very similar, then only one suffice since the two are reductant. On the other hand, if two data points are very dissimilar, the two points contain different information about the classification space and thus both maybe useful for the discrimination of the classes. The support vectors are typically points that are very dissimilar from the rest of the class.

Although new kernel functions are frequently presented in research, the most commonly used kernels are still [19], the linear

$$K(\boldsymbol{x},\boldsymbol{x}') = \boldsymbol{x}^{\mathrm{T}}\boldsymbol{x}' \tag{3.10}$$

the radial-basis

$$K(\boldsymbol{x},\boldsymbol{x}') = \exp(-\gamma d^2(\boldsymbol{x},\boldsymbol{x}')) \tag{3.11}$$

where $\gamma$ is a free-parameter and $d^2$ the squared distance metric, usually the euclidean distance; the polynomial function:

$$K(\boldsymbol{x},\boldsymbol{x}') = (\gamma \boldsymbol{x}^T\boldsymbol{x}' + a)^d \tag{3.12}$$

where $a$ and $d$ are scalars; and the sigmoid function:

$$K(\boldsymbol{x},\boldsymbol{x}') = \tanh(\gamma \boldsymbol{x}^T\boldsymbol{x}' + a) \tag{3.13}$$

where $\gamma$ and $a$ are scalars. Which kernel function to apply often depends on the problem at hand, but further discussion can be found in section 3.2.5.

The kernel function $K$ is incorporated in the SVM learning problem in the following way:

$$max_{\boldsymbol{\alpha}} \; L = \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j K(\boldsymbol{x}_i, \boldsymbol{x}_j) \tag{3.14}$$

subject $0 \leq \alpha_i \leq C$ and $\sum_i \alpha_i y_i = 0$ for all training data points. The function $K$ occupies now the place of $\boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{x}_j$ in problem (3.8), which is a special case. Since $K(\boldsymbol{x}_i, \boldsymbol{x}_j)$ is a scalar and determined before the optimisation process begins, there is no fundamental difference between problem (3.8) and problem (3.14), and thus problem (3.14) can be solved with the same optimisation algorithm then problem (3.8).

The decision rule has now to accommodate the application of the kernel function. This done in the following way:

$$f(\boldsymbol{x}) = sign(\sum_i \alpha_i y_i K(\boldsymbol{x}_i, \boldsymbol{x}) + b) \tag{3.15}$$

where only the support vectors are utilised in the computation. The variable $b$ can be determined in the same way as in the previous cases by $b = y_j - \sum_i \alpha_i y_i K(\boldsymbol{x}_i, \boldsymbol{x})$. Note that if $\boldsymbol{x}$ is very similar to the support vector $\boldsymbol{x}_i$, then $K(\boldsymbol{x}_i, \boldsymbol{x})$ is large, and thus the contribution of the term $\alpha_i y_i$ is large in the summation. This pushes the summation in the direction of the class of the support vector $\boldsymbol{x}_i$.

All development so far was done taking the binary problem as starting point. In the next section the strategies to apply the SVM learning algorithm to multi-class classification problem will be described and discussed.

### 3.2.3 Adapting SVM to multi-class classification problems

The SVM was originally defined as a binary classifier. The adaption of the SVM to a multi-class context is not direct and is still an on-going research topic [30, 94]. Nevertheless, the multi-class adaptation approaches can be broadly categorised in two ways: the combination approaches and the "all-together". The combination approaches consist in utilising several binary classifiers to solve the multi-class problem. Essentially, these approaches represent different ways to break down the multi-class problem into a sequence of small binary problems. The "all-together" approaches consist in solving a large optimisation problem, similar to that of the soft margin classifier, but considering all classes of the problem instead of just two. Examples of these approaches are the Crammer and Singer SVM [27] and the methods proposed in [143]. Here focus is on the combination approaches, and this is for two reasons: first, although available since the beginnings of the use of SVM, the "all-together" have never been fully adopted by the remote sensing community; second, studies comparing these methods and the combination approaches indicate that the combination approaches tend to be more suitable for practical use [94, 123]. Combination approaches, as mention before, represent different ways to break down a multi-class problem in a set of smaller binary classification problems. The three more common methods are the One-vs-All (OVA) [12], the One-vs-One (OVO) [75, 83] and the Directed Acyclic Graph (DAG) [117].

Let be assumed that the multi-class problem is composed by $k$ classes. The OVA is perhaps the most intuitive. It consists in picking one class of the set of classes as the class of interest and reunite all other into one large nominal class, and then develop a binary classifier. This binary classifier is then an expert identifying that class of interest. The procedure is repeated for all other classes. The OVA is then an ensemble of $k$ classifiers. At allocation time, the point being classified is evaluated by each binary classifier and each one outputs a confidence metric. The confidence metric for SVM can be defined as [94]:

$$f(\boldsymbol{x}) = argmax_{j=1\ldots k}|\sum_{i}\alpha_i y_{ji} K(\boldsymbol{x}_{ji},\boldsymbol{x}) + b_j| \tag{3.16}$$

where $\boldsymbol{x}_{ji}$ is the $i$-th support vector of the $j$-th classifier (expert in the $j$-th class). Note that equation 3.16 is the metric utilised in equation 3.15. However this approach is not without limitations. Firstly, the scale of the confidence values may differ between the binary classifiers [9]. This can in part be minimised by scaling the confidence metrics [129]. Secondly, even if the class distribution is balanced in the training set, the binary classifiers see unbalanced distributions. This is because the class of interest typically only a small component of the whole, that is the set of negatives is compose by all other classes. Thirdly, the general decision rule may produce tied cases. Those cases are often solved randomly.

The OVO tackles the problem differently. Here all possible pairs of classes are enumerated and a binary classifier is developed for each one of them. Thus each classifier is sensitive only to two classes, and thus for any point to be classified, the component classifiers can only be allocated in one of the two. The final decision is done by voting. The class with more votes is the class to be assigned. However the OVO can also have tied cases. In those situations there is not clear way to solve it and the majority of the implementations solve these cases randomly [94]. For a problem of $k$ classes the number of binary classifiers is $\frac{1}{2}k(k-1)$. This is considerable; for example, for a problem consisting in 10 classes the OVO has to compute 45 binary classifiers, where the OVA has to compute only 10; and for 20 classes the number of classifier for OVO jumps to 190.

The DAG, like the OVO, breaks down the classification problem in pairs. In fact, the same pairs of classifier of OVO. Thus, DAG comports as many classifiers as OVO. However, instead of organising these pairs sequentially, like OVO, DAG organise them in a tree. Figure 3.2 illustrates the application of DAG for a 3-class problem. The allocation phase with DAG is like the decision flow in a decision tree. The first classifier is the root (2 vs. 3). If the decision falls for 2, the flow moves left, otherwise moves right. The process eventually ends in a leaf node, the final decision. The process is faster than that of the OVO since OVO is effectively doing a binary search and OVO a linear search [117]. In fact, for a 10-class problem, both approaches comport 45 classifiers. But at allocation phase, all classifiers of OVO have to operate while only

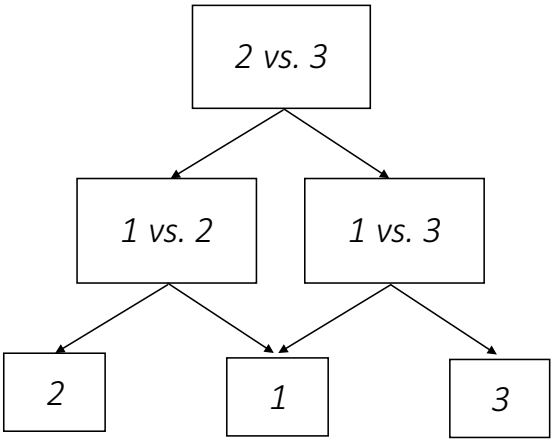



Figure 3.2: The directed acyclic graph approach to the application of SVM to multi-class problems. Here a 3-class problem.

roughly 5 of DAG will operate; and for a 20-class problem, only 8 operate against the 190 of OVO.

### 3.2.4 One-class support vector machines

As it was shown in section 3.2.1, the SVM was developed to solve binary classification problems. However, the same principles can be applied to solve one-class problems, also know as novelty detection problems [30]. This problem consists in detecting objects from a particular class, often called target class or class of interest. These problems differ greatly from the standard supervised classification in the sense that the training set is composed exclusively by data points from the target class and thus there are no counterexamples to define the classification space outside the class of interest. One-class classification has been utilised in a variety of applications [30] and has great potential in remotely sensed data processing. There are two approaches to one-class classification based on SVM principles, One-Class Support Vector Machines (OCSVM) [128] and the Support Vector Data Description (SVDD) [137]. In this dissertation, however, focused in on the use of OCSVM.

The basic idea behind the OCSVM is to determine a function that signals positive if the given data point belongs to the target class and negative otherwise. To achieve that the classification space origin is treated as the only available member of the non-target class (figure 3.3). The problem is then solved by finding a hyperplane with maximum margin separation from the origin. Non-linear problems are dealt with a kernel function as in the binary SVM. The OCSVM optimisation problem is formulated as follows [128]:

$$\min_{w,\xi,\rho} \frac{1}{2} w^{\mathrm{T}} w - \rho + \frac{1}{\nu m} \sum_{i} \xi_i \tag{3.17}$$

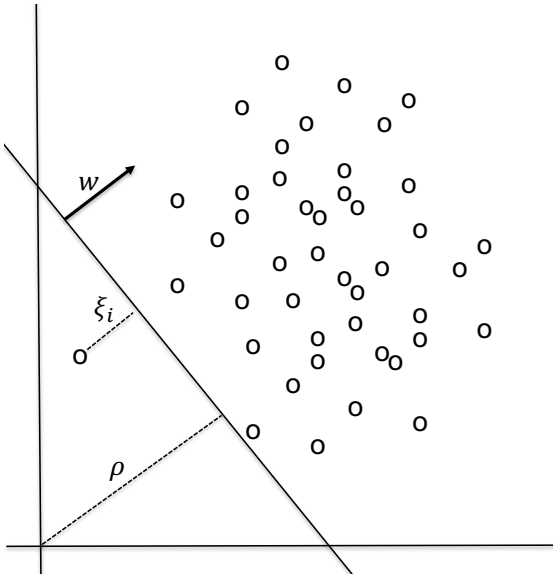



Figure 3.3: One class support vector machine. The origin is treated as the only available member of the non-target class. The vector $w$ is normal to the separating hyperplane.

subject to $\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\phi}(\boldsymbol{x}_i) \geq \rho - \xi_i$ and $\xi_i \geq 0$. Here, $m$ is the number of training data points, $\boldsymbol{w}$ is the vector perpendicular to the hyperplane that defines the target class boundaries and $\rho$ is the distance to the origin (figure 3.3). The function $\boldsymbol{\phi}$ is related with the kernel function [128]. The use of slack variables $\xi_i$ used in the OCSVM to allow the presence of class outliers, similar to binary SVM (figure 3.3). The parameter $\nu$ ranges from 0 to 1 and controls the trade-off between the number of data points of the training set labelled as positive by the OCSVM decision function:

$$f(\boldsymbol{x}) = sign(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\phi}(\boldsymbol{x}) - \rho) \tag{3.18}$$

Applying the KKT conditions to the original OCSVM problem, this can be rewritten as depending of the Lagrange multipliers $\alpha$:

$$\min_{\boldsymbol{\alpha}} \sum_{i,j} \alpha_i \alpha_j K(\boldsymbol{x}_i, \boldsymbol{x}_j) \tag{3.19}$$

Subject to $\sum_i \alpha_i = 1$ and $0 \leq \alpha_i \leq \frac{1}{\nu m}$ for all training data points, where $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{\phi}^T(\boldsymbol{x}_i)\boldsymbol{\phi}(\boldsymbol{x}_j)$ is the kernel matrix defined by the kernel function $\boldsymbol{\phi}$. Note that the algorithm utilised to solve the SVM optimisation problem can also be used the OCSVM problem. From this, the decision function can be rewritten depending only on the non-null Lagrange multipliers and on the kernel matrix values:

$$f(\boldsymbol{x}) = sign(\sum_i \alpha_i K(\boldsymbol{x}_i, \boldsymbol{x}) - \rho) \tag{3.20}$$

The data points with non-null Lagrange multipliers are effectively the support vectors of the one-class classifier. The classification rule is based on the signal of the decision

value, positive if the data point is located inside the target class, negative otherwise. The absolute value of the decision value is directly related with the distance of the data point to the separating hyperplane in the transformed classification space [128].

### 3.2.5 Model selection of support vector machines

In the SVM classification algorithm, there is a set of free parameters that need to be set. These are the kernel and its parameters and the penalty factor. For example, to use a SVM with the radial-basis kernel function it is necessary to set the radial factor $\gamma$ and the penalty factor $C$. These parameters can affect considerably the performance of the resulting classifier, and thus parameter tuning becomes crucial [17]. Indeed, often tuning is more important than the choice of learning algorithm [89], and the SVM in particular is harder to tune than other classification algorithms [18]. Choosing the parameters that yield the classifier with the best performance in terms of the discrimination of unseen data points is known as model selection problem [61]. This problem is ill-defined, since the distribution of the unseen data points is unknown [17]. Hence, most users utilise proxy procedures to estimate the testing error such as cross-validation [76]. Nevertheless, the first parameter to be defined is the kernel function which has to be defined prior to the cross-validation process.

As discussed in section 3.2.1 there the most common kernels are the linear, the radial-basis, the polynomial and the sigmoid function. But, the two most common functions are the linear and the radial-basis [19, 30]. However, there two good reasons why the radial-basis kernel function should be the off-the-shelf kernel for most applicational problems. The first reason is that, excluding the linear kernel, the radial-basis function is the kernel with the least number of parameters to be fixed. This is an advantage since the more parameters is available in the kernel the harder it is to tune the learning algorithm. The other reason is that the linear kernel is a special case of the radial-basis [72]. In other words, for any SVM classifier trained with a linear kernel and a penalty factor $C$ there is a parameterisation $(\gamma, C')$ that yields a classifier with the same performance of that linear SVM. In this way, discussion will be focused on the determination of the radial-basis parameters.

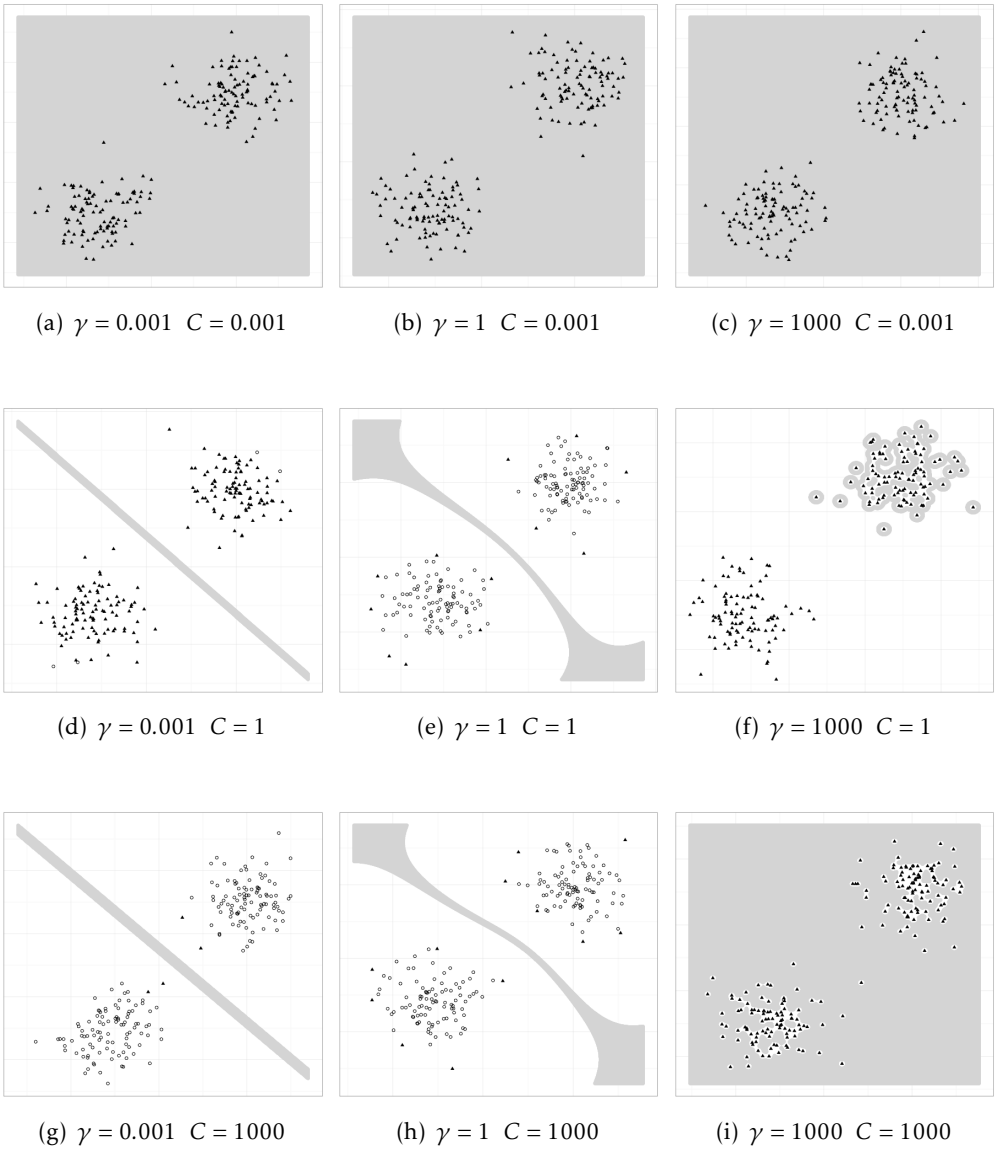
As previously mentioned, to train a radial-basis SVM is necessary to fix the penalty factor $C$ and the radial factor $\gamma$, and the is typically done by cross-validation trails. Here, the training set is divided in two parts. One is used to training and the other is used to test the classifier. This idea is often implemented in a $k$-fold process, where $k$ is an integer number typically 3, 5 or 10. This $k$-fold cross-validation breaks down the training set in $k$ equal or approximately equal parts, and uses each part as testing set and the remain as training set. Then an accuracy metric is estimated for each fold and the results are averaged. The identification of the particular parameterisations is commonly done by grid-search. In other words, the range of possible values of each free-parameters is broken down in small elements and then the Cartesian product is

determined, resulting in a grid of trail points. [19] suggests to break the range of $\gamma$ and $C$ is powers of two. For example, for $\gamma$ a good breakdown is $2^k$ for $k = -15, -14, ..., 3$, and for the penalty factor $C$ a good breakdown is $2^k$ for $k = -3, -2, ..., 15$. From this decomposition, it is derived a $19 \times 19$ grid where each point is pair $(\gamma, C)$. For each one of these pairs, a $k$-fold cross-validation is employed and the pairs with highest accuracy is selected. [19] suggest to start with a coarse grid to identify a "good" region, and finer grid inside that region to identify more specific values. Typically, the accuracy metric utilised in the grid-search cross-validation is the classification accuracy. However, other metrics can be used, and in some cases, such as imbalanced data sets, other metrics have be used. Examples of such metrics are the sensitivity, the specificity, $G$-mean, $F$-measure, etc [63].

The cross-validation process is a generic method that is staple for any analyst intending to apply supervised classification algorithms that need fine tuning. Nevertheless, the understanding of how the algorithm parameters inform the induction process can be useful, since it may guide the analyst to select parameter in an informed way.

The parameter $C$ controls the penalisation to be assigned to every misclassification, and as consequence the number of support vectors of the classifier. In detail, if $C$ is large, the cost of misclassification is large. The optimisation process thus is forced to search for a solution where the least number of training data points are misclassified. This is because to minimise the term $C \sum \xi_i$ in the SVM optimisation problem, each $\xi_i$ has to be small (close to zero) to overcome the influence of the large constant $C$. This forces the optimisation process to search for very sparse solutions [43, 144], where only a small number of $\xi_i$ are not zero. These points, as it was discussed in 3.2.1, constitute the support vectors. Thus large $C$ leads to classifiers small number of support vectors. This may result in an accurate classifier or, in the presence of a noisy training data set, in a overfitted classifier [86, 138]. On the other hand, when $C$ is small, more data points are allowed to be misclassified, since the optimisation process does not need to force the $\xi_i$ to be small to overcome $C$. As result, more data points are included as support vectors. For very small values (close to zero) of $C$, the number of support vectors may include the entire training data set.

Figure 3.4 represents a set of examples of SVM trained with different combinations of $\gamma$ and $C$. The two classes are linearly separable and were artificially generated. The centre of the positive class $(1, 1)$ and the centre of the negative class is $(5, 5)$. Variance of both classes is $0.5I$ where $I$ is the identity matrix. The grey represents the regions where decision is made with a difference not superior to 5%, that is $|p_+ - p_-| < 0.05$ where $p_+$ is the probability of a point to belong to the positive class and $p_-$ is the probability of a point to belong to the negative class. The probability values were infer from the SVM using the Platt's scalling technique [116]. The classification problem was purposely easy to solve and the value 5% was used for illustration purposes only.



(a) $\gamma = 0.001$ $C = 0.001$ (b) $\gamma = 1$ $C = 0.001$ (c) $\gamma = 1000$ $C = 0.001$

(d) $\gamma = 0.001$ $C = 1$ (e) $\gamma = 1$ $C = 1$ (f) $\gamma = 1000$ $C = 1$

(g) $\gamma = 0.001$ $C = 1000$ (h) $\gamma = 1$ $C = 1000$ (i) $\gamma = 1000$ $C = 1000$

Figure 3.4: Two linearly separable classes, the positives and the negatives, artificially generated. Full triangles represent the support vectors and circles represent other data points. The centre of the positive class $(1, 1)$ and the centre of the negative class is $(5, 5)$. Variance of both classes is $0.5I$ where $I$ is the identity matrix. The grey are represents the region where decision is made with a difference not superior to 5%, that is $|p_+ - p_-| < 0.05$ where $p_+$ is the probability of a point to belong to the positive class and $p_-$ is the probability of a point to belong to the negative class.

Frames $(a)$, $(b)$, $(c)$ were generated with $C = 0.001$; frames $(d)$, $(e)$, $(f)$ were generated with $C = 1$; and frames $(g)$, $(h)$, $(i)$ were generated with $C = 1000$. As it can be observed the number of support vectors increase when $C$ decreases and decreases when $C$ increases. Particularly illustrative are frames $(a)$, $(b)$, $(c)$ where $C$ is extremely small, 0.001. Here the number of support vectors constitute de entire training data set and the classification decisions are all based on difference inferior to 5%, this indicates low confidence classifications. In contrast with other cases where the grey area of ambiguity is much reduced.

The parameter $\gamma$ is the radial-basis kernel and is responsible to control the similarity between two data points. More concretely, the kernel function, as mention in section 3.2.1, is a similitude function, and it is utilised in the SVM to find the most dissimilar data points, which tend to become the support vectors. The $\gamma$ parameter then controls how close two data points must be to be considered similar. This is important, in particular, during the allocation phase, because the weight associated to a support vector is more influential in the allocation decision of a given point the more similar that point is to the support vector. The parameter $\gamma$ is inversely proportional to the variance in the neighbourhood of the support vectors [128]. Indeed, it is common to use the relation $\gamma = \frac{1}{2\sigma^2}$ in implementations of the radial-basis function, where $\sigma$ represents the kernel width, to better represent the control of this parameter over variance. Thus, for large values of $\gamma$, the variance around a given support vector is small and thus a point has to be very similar (and thus close) to the support vector to be allocated in the same class of the support vector. For small values of $\gamma$, however, the variance around a support vector is large and dissimilar points (thus futher) are allowed to be included in the class of the support vector. Although $\gamma$ is not the main responsible to control the number of support vectors, it can affected. Indeed, for very large values of $\gamma$, each individual data point becomes important to describe its small neighbourhood since its neighbour points are also restricted. Smaller values of $\gamma$ may entail less support vectors because each point is allowed to expand its neighbourhood and thus a small number of data points may be enough to describe the training set.

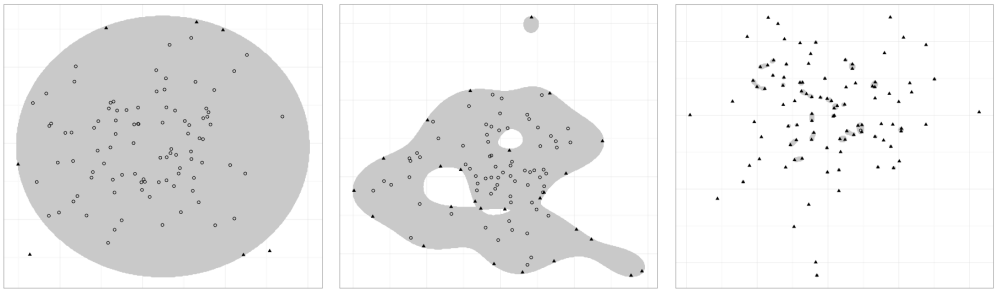
The effect of different magnitudes for $\gamma$ can be observed in figure 3.4. Frames $(a)$, $(d)$, $(g)$ were generated with $\gamma = 0.001$; frames $(b)$, $(e)$, $(h)$ were generated with $\gamma = 1$; and frames $(c)$, $(f)$, $(i)$ were generated with $\gamma = 1000$. Excluding the top row (frames $(a)$, $(c)$, $(c)$), where small value of $C$ overtake the effects of $\gamma$, small values of $\gamma$ tend to reduce the grey regions indicating the decision with low uncertainty. Note that for sufficiently small values of $\gamma$ the decision boundary will resemble a straight line, although the radial-basis function is not linear [72] (frames $d$ and $g$), and frames $(e)$ and $(h)$ represent a quasi-straight line. Indeed, the main difference between frames $(d)$ and $(g)$ is the number of support vectors due to the different values of $C$. Frames $(e)$and $(h)$ are similar in terms of grey region and number of support vectors despise the different values of $C$. Frame $(f)$ represent an extreme case where $\gamma$ is extremely large entailing very low variability around the support vectors. This leads to two results:

first the increase of support vectors, because each point becomes important to describe the small region around itself and the domination of one class over the classification space. In this case the grey regions are around the top class indicating that the regions of uncertainty are localised around the support vectors of this class, thus restricting the area of classification of this class to its support vectors. This is expected since in a binary classification all data points have to fall in one, and only one, of the classes. Each class will be constrained depends in the way the algorithm was implemented [30]. Frame (*i*) represent another extreme case where $\gamma$ and $C$ are both large. This leads to the definition of a large number of support vectors, despite the value of $C$, because each data point is restricted to its small neighbourhood.

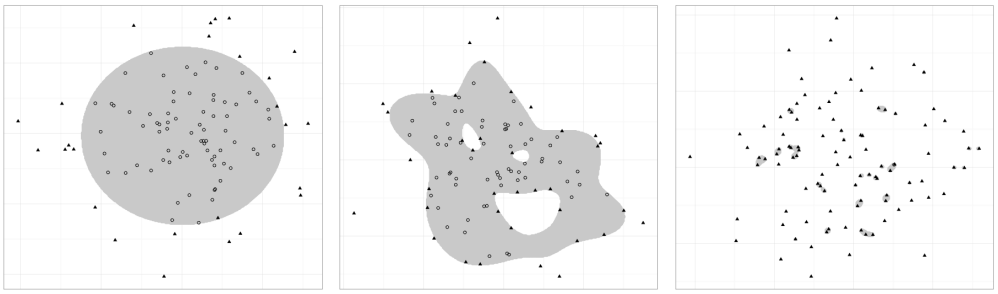
In practical terms, how can a user utilise these observations to better guide the tuning of a SVM and ultimately producer better maps? If the user realises that the class of interest is being underestimated, this may indicate that the classifier is being strict and is allocating to the class only those pixels that are very similar to the support vectors. In this case, the classifier may be lacking variability. That may be improved by increasing the variability around the support vectors (decrease $\gamma$) and/or by increasing the number of support vectors (decrease $C$). On the other hand, if the class of interest is being overestimated, that may indicate that the model has too much variability. That could be improved by increasing $\gamma$ and/or increasing $C$. These are only guidelines and do not represent an universal solution.

Similar analysis can be applied to OCSVM, to assess the effects of $\gamma$ and $\nu$. The parameter $\gamma$ behaves in the same way with OCSVM and SVM. The $\nu$ parameter on the other hand is particular of OCSVM. This parameter that $0 < \nu < 1$, is particularly important, since it defines the upper bound of the fraction of training data points regarded as outliers and the lower bound of the fraction of training data points regarded as support vectors [127]. Thus if $\nu$ is increased, the optimisation process is allowed to exclude more data points and regard them as class outliers. This will lead to smaller class regions but also to a larger number of support vectors. On the other hand, by reducing $\nu$, the number of data points that can be excluded as outliers is smaller. Figure 3.5 represent the combination of multiple values of $\gamma$ and $\nu$. Frames (*a*), (*d*), (*g*) show the region of the class of interest, in grey, for a small $\gamma = 0.001$. The shape resembles that of a circle and the effect of increasing $\nu$ is the reduction of the area of that region. The increase of $\nu$ leads to a larger number of data points to be regarded as outliers. Thus only those innermost points are elements of the class of interest. As a result, the class variability is expected to be small.

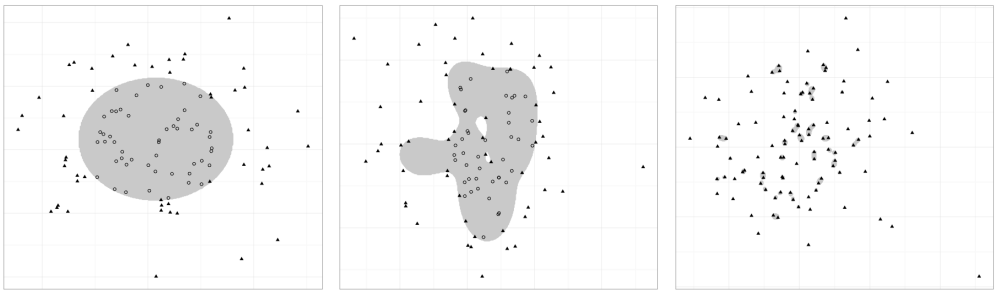
When $\gamma$ increases (frames (*b*), (*e*), (*h*)), the variability around the training data points is reduced and the region of the class starts to acquire class specific shapes and in some cases holes. However, the effect of increasing $\nu$ is the same: a reduction of the overall region of the class of interest. Frames (*c*), (*f*), (*i*) represent an extreme case where $\gamma$ is very large. Here the variance in the neighbourhood of each data points is extremely small and, independently of the value of $\nu$, the region of the class of interest



(a) $\gamma = 0.001$ $\nu = 0.05$ (b) $\gamma = 1$ $\nu = 0.05$ (c) $\gamma = 1000$ $\nu = 0.05$



(d) $\gamma = 0.001$ $\nu = 0.25$ (e) $\gamma = 1$ $\nu = 0.25$ (f) $\gamma = 1000$ $\nu = 0.50$



(g) $\gamma = 0.001$ $\nu = 0.50$ (h) $\gamma = 1$ $\nu = 0.50$ (i) $\gamma = 1000$ $\nu = 0.50$

Figure 3.5: An artificially generated class. Circles represent points and full triangles the support vectors. The centre of the positive class $(1,1)$ and variance is $0.5I$ where $I$ is the identity matrix. The grey regions represent the regions classified as class of interest.

Table 3.1: A $2 \times 2$ confusion matrix.

| | Actual positive | Actual negative |
|---|---|---|
| Predicted positive | $TP$ | $FP$ |
| Predicted negative | $FN$ | $TN$ |

is severely reduced.

To fine tune a OCSVM is harder than to fine tune a SVM. The training set of OCSVM does not contain data points outside the class of interest, and thus it is not possible to assess the overall accuracy nor the specificity (the proportion of data points outside the class of interest that were correctly classified) of the classifier in the cross-validation process [94, 109]. Effectively only sensitivity can be assessed. Using the sensitivity alone to parameterise a classification algorithm may result in a classifier with high sensitivity and low specificity, overestimating the extension of the classes of interest.To minimise the effects of this limitation, the cross-validation process can be carried out using the ratio between the sensitivity and the number of support vectors as metric [4, 35]. This ratio enforces high sensibility while limiting model complexity which usually indicates good model generalisation ability [35].

## 3.3 Accuracy metrics for binary classification

The design and implementation of a learning algorithm require the use of accuracy metrics to assess the quality and compare the performance of alternative classifiers. For example, when fine-tuning a classification algorithm, it is often necessary to compute an accuracy metric to determine the parameterisation that yields the best score. Although commonly used, the overall classification accuracy (the proportion of correctly classified data points) may not always be a reliable metric. One such cases is when the training set is imbalanced. This is because the majority class dominates the behaviour of this metric, and thus it gives optimistically biased results [145].

Indeed, the definition of the accuracy metric is particular important for binary classification, since the performance of the classifiers can be particularly sensitive to the classes' relative size [16, 145]. In this conditions, the results of the fine-tuning process may be unreliable not because of the process but rather because of the accuracy metric employed in the process. If the training data set is imbalanced and the classification accuracy is utilised, the outcome of the fine-tuning process will indicate that a particular parameterisation is the one with the highest classification accuracy. But may indeed be biased towards the majority class, since that parameterisation may yield a classifier that identifies very accurately the majority class in detriment of the minority class [67].

There are however better alternative accuracy metrics to the classification accuracy specially when the data set is imbalanced, for example sensitivity and specificity [61].

At the basis of this analysis is the binary confusion matrix (table 3.1).

In table 3.1, True Positive (TP) represents the number of actual positive cases correctly classified, True Negative (TN) the number of actual negative cases correctly classified, False Positive (FP) the number of actual negatives predicted as positives, and False Negative (FN) the number of actual positives predicted as negatives. The classification accuracy is then the proportion of true positives and true negatives which is commonly used to metric classification performance in multi-class problems [129, 145]:

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.21}$$

But in binary classification, classification accuracy may not be a reliable indicator particularly due to possible data imbalances in the training set. Alternatively, sensitivity and specificity can be utilised [129, 145]. Sensitivity is the proportion of true positives correctly classified (true positive rate) and specificity is the proportion of true negatives correctly classified (true negative rate) [61, 63]:

$$sensitivity = \frac{TP}{TP + FN} \tag{3.22}$$

$$specificity = \frac{TN}{TN + FP} \tag{3.23}$$

Effectively, sensitivity is the producer's accuracy of the positive class while specificity is the producer's class of the negative class. In this way, sensitivity indicates how good the classifier is recognising positive cases and specificity indicates how good the classifier is recognising negative cases [145]. Indeed, sensitivity and specificity are the accuracies metrics associated to the rates of type-I error (also known as false negative rate) and type-II error (also known as false positive rate), respectively. That is,

$$sensitivity = 1 - false\ \ negative\ \ rate \tag{3.24}$$

$$specificity = 1 - false\ \ positive\ \ rate \tag{3.25}$$

In this sense, sensitivity and specificity are closely related with the receiver operating characteristic (ROC) that displays the relation between sensitivity and false positive rate ($= 1 - specificity$) [14].

Although not common, the analyst may rely on the scatter plot with the specificity depending on the sensitivity, for a better visual comparison between multiple classifiers. Figure 3.6 illustrates such a plot, where the dashed line represents the 1:1 straight line and the points A, B, C and D represent the hypothetical performances of four classifiers. Note that specificity and sensitivity range from 0 to 1 and thus this chart is similar to plot the performance of each classifier in a ROC. However, displaying directly sensitivity and specificity is easier for the reader that is not used to the analysis of binary classifiers.

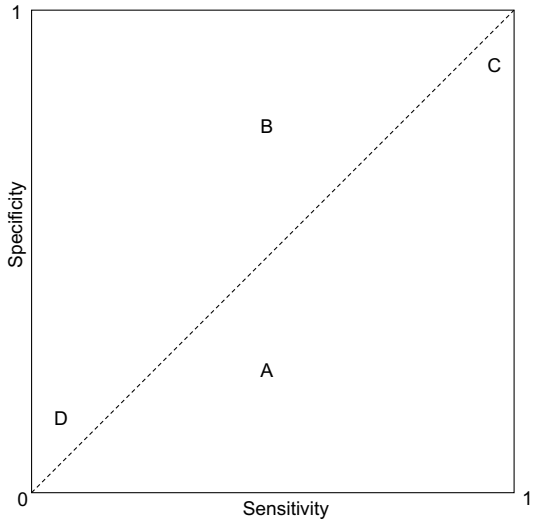



Figure 3.6: Graphical representation of sensitivity and specificity. The dashed line represents the 1:1 line. Note that if a point is located on the 1:1 line, this indicates that sensitivity is equal to specificity. Points above the 1:1 line indicates that specificity is larger than sensitivity; and points bellow 1:1 line indicates specificity is smaller that sensitivity. Points A, B, C and D represent the hypothetical performances of four classifiers.

The elements of figure 3.6 can be interpreted in the following way: each classifier is presented by a point characterised by $(x, y)$, where $x$ represents sensitivity and $y$ represents specificity. Thus a point on the dashed line indicates a classifier with specificity equal to sensitivity. If the point is bellow the line, this indicates a classifier where specificity is smaller than sensitivity; but if above the line, indicates a classifier where specificity is larger than sensitivity. This is relevant and informative since a classifier bellow the dashed line, like that in point A, represents a classifier with a large number of false positives. In other words, classifier A may be overestimating the positive class, which is typically associated with the class of interest. A classifier like B, on the other hand, represents the opposite. That is, classifier with a large number of false negatives, and thus may be underestimating the positive class. To compare classifiers, the analyst can visually assess the proximity of the classifier to the top-right corner. If a classifier like C is closer to the top-right corner than, say, A, this indicates a "better" classifier since sensitivity and specificity are simultaneously high. In contrast, a classifier, like D, closer to the bottom-left corner, indicates a "worst" classifier since at least one of the components, sensitivity or specificity, is lower. Although, this approach is simple to interpreter, is difficult to use for computational purposes.

Often sensitivity and specificity are combined in one metric for a better numerical comparison [135]. In particular, the geometric mean between sensitivity and specificity [84] is particularly useful since it is based on the multiplication between these two quantities:

$$G = \sqrt{sensitivity \times specificity} \tag{3.26}$$

The geometric mean ($G$) indicates the balance between classification performances on the positive and negative class. High misclassification rate in the positive class will lead to a low geometric mean value, even if all negative data points are correctly classified [67]. For example, in a binary classification problem, if 10% of training set is in the minority class and 90% is located in the majority class, a classifier that simply classifies every data point as belonging to the negative class yields an overall accuracy of 0.90 [145]. However, its sensitivity is 0.0 and specificity 1.0; thus geometric mean $G$ is 0.0. In this way, if both sensitivity and specificity are high, the geometric mean $G$ is also a high value; but if one of the component accuracies, sensitivity or specificity, is low, the geometric mean $G$ is affected by it. Indeed, the geometric mean can be an important accuracy metric for class specific mapping, since it is particularly sensitive to the over-fitting to the negative class (i.e. others class) and to the degree in which the positive class (i.e. class of interest) is neglected [110]. In others, since the class of interest is typically a small component (the minority class), the geometric mean establish then balance between the minority class and the majority class. This is important to assess how biased to the majority class a particular classifier is. This aspect is particular important the definition of the SVM algorithm free-parameters [63].

Note however that sensitivity, specificity and their geometric mean are not the only alternative metrics to overall accuracy. Indeed, from the binary matrix is possible to derive other metrics depending on their applicational value. For example, precision is a commonly utilised metric in binary classification in the area of information retrieval and is given by [37]:

$$precision = \frac{TP}{TP + FP} \tag{3.27}$$

Thus in this sense precision is equivalent to the users' accuracy of the class of interest. This metric is often combined with sensitivity (equivalent to the producers' accuracy) of the class of interest through the harmonic mean; this metric is usually called $F$-measure or $F$-score [37]:

$$F - measure = 2 \times \frac{precision \times sensitivity}{precision + sensitivity} \tag{3.28}$$

The $F$-measure thus establish a balance between the precision and sensitivity. And this is important in information retrieval since only the number of positive in the total of retrieved elements (that is the predicted elements) is important [115]. In other words, the number of true negatives is not relevant for that application. However, for specific class mapping, the rate of true negatives (specificity) is relevant for the goal since to ignore it may lead to high sensitive classifiers and, as consequence, to the

overestimation of the class of interest. For this reason, the geometric mean between sensitivity and specificity may be a better metric than $F$-measure for specific class mapping.

## 3.4 Comparing the accuracy of classifiers

Comparing the classification accuracies of different methods is utilised in remote sensing research as basis of comparative studies, to compare for example classification methods [50], image processing methods [132], etc. These studies have typically been done using hypothesis testing approaches based on the statistical significance of the difference. These testes are commonly the comparison of the kappa coefficients [94], proportion of correctly allocated cases [6] or the McNemar test [53]. Most studies have focused on the magnitude of difference in accuracy, regardless of its direction, to show the inequality between methods. This usually done by testing the method under analysis and a benchmark showing that the accuracy of the first is larger than that of the second, and then performing an hypothesis test to show the difference is statistically significant.

However in some cases the use of this approach is not appropriate. A typical case is when that purpose of the study is to show that the method under analysis is at least not worst than the benchmark but it is more convenient in some way. For example, in [114], where the purpose was to shown that relevance vector machine and multinominal logistic regression approaches where at least non-inferior to SVM although requiring less training data points, and thus being more a convenient approaches.

For cases like this, the test for inequality is not useful. The reason why standard hypothesis testing are not suitable to test non-inferiority or equality is subtle but relevant, and thus important to make it clear. Conventional statistical hypothesis testing evaluates two competing hypothesis. The null hypothesis ($H_0$) that states that there is no difference in accuracy ($p_1 - p_0 \neq 0$), where P1 is the classification accuracy (a proportion) yield by the testing method and P0 is the classification accuracy yield by the benchmark, and the alternative hypothesis ($H_1$) that negates the null hypothesis $H_0$, that is $p_1 - p_0 \neq 0$. To show that the purposed method and the benchmark are different, it is necessary to reject $H_0$. By rejecting $H_0$, the difference between accuracies is viewed as statistically significant, depending on the statistical parameters of the test. Then $H_1$ has to be accepted, showing the methods are different. However, if $H_0$ is not rejected, that does not entail the acceptance of $H_0$, but only the failure to reject $H_0$. Within the scientific principle of falsification [118], it is the rejection of the null hypothesis $H_0$ that is useful for the progress of scientific knowledge. In other words, it is by showing that there are evidences to reject the hypothesis that the methods are different that one can conclude that the methods are similar [49], and not by failing to reject.

How then can an analyst compare two proportions with interest focused on the equivalence or non-inferiority? This is briefly answered in the next paragraphs. But a

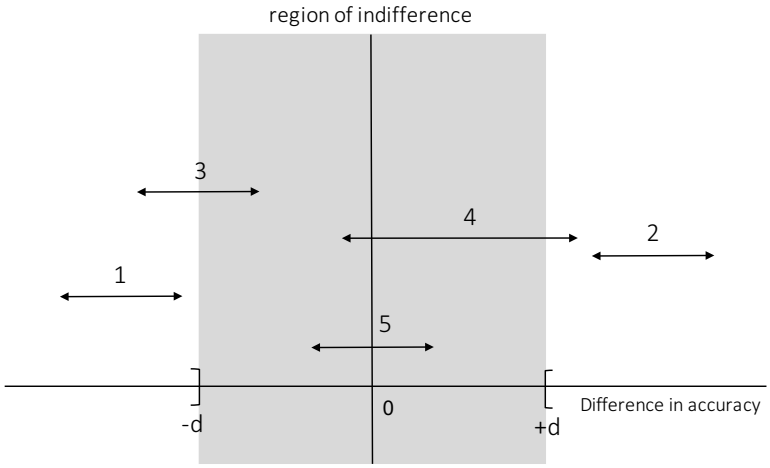



Figure 3.7: Scenarios to illustrate the interpretation of confidence intervals of the difference between proportions. The grey are represents the region of indifference.

more complete examination of the question can be found in [49], where it is presented the application of the method to remotely sensed data analysis, and in [42] where an exhaustive elaboration of all statistical details is discussed.

Important for the non-inferiority and equivalence tests is the definition of the region of indifference. The region of indifference is the maximum allowed difference in accuracy to consider the difference negligible [42]. Thus the null hypothesis is not that the difference is zero, but rather that the difference is no larger that indifference magnitude. In figure 3.7, the region of indifference is represented in grey and is numerically represented by $]-d,+d[$. The amplitude of the region of interest is typically application dependent. However studies applying this statistical procedure, such as [50, 113, 114] tend to define the region of interest between $d = 1\%$ and $d = 2\%$. In this way, the null hypothesis is redefined considering this region of indifference.

For the non-inferiority test, if $p_1$ is the accuracy yield by the method under analysis and $p_0$ the accuracy yield by the benchmark, the null hypothesis $H_0$ claims that $p_1 \leq p_0 - d$. In other words that the method under analysis is inferior to the benchmark. Clearly, the alternative hypothesis $H_1$ claims the opposite that $p_1 > p_0 - d$, that is the method is not inferior to the benchmark. For the equivalence test, the null hypothesis $H_0$ claims that $|p_1 - p_0| \geq d$, that is the method is different from the benchmark. Note that the claim entails nothing about which one has greater accuracy, only that the difference is outside the region of indifference. Thus the alternative hypothesis $H_1$ claims that $|p_1 - p_0| \leq d$. Note that to test the difference the burden of proof has to reverse, that is $H_0$ thus claims $|p_1 - p_0| \leq d$ and $H_1$ the opposite $|p_1 - p_0| \geq d$.

The confidence interval of the difference between two proportions is given by [42]:

$$p_1 - p_0 \pm z_\alpha SE \tag{3.29}$$

Table 3.2: McNemar $2 \times 2$ confusion matrix comparing the results of two classifiers. 0 represents incorrect testing data point and 1 represents correct testing data point. $p_{00}$ represents the proportions of testing data points where both classifiers made incorrect predictions, $p_{11}$ represents the proportions of testing data points where both classifiers made correct predictions, $p_{01}$ represents the proportions of testing data points where the first classifier made an incorrect prediction but the second predicted correctly, and $p_{10}$ represents the proportions of testing data points where the second classifier made an incorrect prediction but the first predicted correctly.

| | | Classifier 2 | |
|---|---|---|---|
| | | 0 | 1 |
| Classifier 1 | 0 | $p_{00}$ | $p_{01}$ |
| | 1 | $p_{10}$ | $p_{11}$ |

where $SE$ is the standard error of the difference between the estimated proportions and $z_\alpha$ is the Z-table value for $\alpha$ level of significance. The $SE$ is computed using the standard formula of the variance of the difference between random variables [42]. However, to apply this equation it is necessary to know if the proportions are being estimated from dependent samples. Typically in remote sensing studies, users utilise the same testing data set in comparative studies. For example, in studies comparing classifiers it is common for the same testing set to be used to aid like-for-like comparison [49]. In these cases, the assumption of independence is unsatisfied and an alternative techniques should be used [42]. One approach, appropriate as test inequality of proportions is the McNemar test [85]. This approach has been adopted in remote sensing studies as a tool for evaluating the significance of the difference in accuracy, when a single testing data set is utilised, such as [52, 53, 55, 125]. However, the McNemar test can also be used to test non-inferiority and equivalence, and it is possible estimate the variance of the difference between two proportion.

The McNemar test utilises a binary confusion matrix of the classifications by the two classifiers under comparison. The main diagonal of this matrix (table 3.2) shows the proportion of pixels upon which both classifiers were correct ($p_{11}$) and on which both classifiers made an incorrect prediction ($p_{00}$). The McNemar test focuses on the proportion of discordant pixels, that is $p_{10}$ and $p_{01}$. These are the pixels upon which one classifier was correct but the other gave an incorrect allocation. In these conditions, $SE$ can be determined with [42]:

$$SE = \sqrt{\frac{p_{01} + p_{10} - (p_{01} - p_{10})^2}{n}} \tag{3.30}$$

Where $n$ is the testing data set size. Thus it is possible to define the confidence interval for the difference between accuracies with $]p_1 - p_0 - zSE, p_1 - p_0 + zSE[$ with confidence level of $(1 - \alpha) \times 100\%$.

In [49] it is suggested the use of a diagram, like the one in figure 3.29, to facilitate appreciation of the value of confidence intervals in comparative analyses. Cases 1 and

2 represent confidence intervals that lies entirely outside the region of indifference. Thus, the difference is statistically significant and the classifications are not equivalent. Case 3 and case 4 are more ambiguous than cases 1 and 2, since they partially lie inside the region of indifference. This suggest that further analysis is necessary, perhaps with a larger testing data set, to evaluate the difference in accuracy with more precision. Case 5 is the most clear case, since the interval is contained in the region of indifference, providing evidence of equivalence between classifications.

## 3.5 Conclusion

SVM has become a staple tool for data analysts processing remotely sensing data. However, tuning this algorithm may be difficult. Here, the SVM algorithm was briefly discussed and the behaviours of its parameters explained to better inform the user when tuning this type of algorithms. It was also shown how is possible to combine multiple binary SVM to solve multi-class problems and how to solve single class classification problems. Finally accuracy assessment metrics for binary classifiers were addressed, as well how to compare the performance of difference classifiers.

Chapter 4

# Improving specific class mapping by cost-sensitive learning

**Abstract** In many remote sensing projects one is usually interested in a small number of land cover classes present in a study area and not in all the land cover classes that make-up the landscape. Previous studies in supervised classification of satellite images have tackled specific class mapping problem by isolating the classes of interest and combining all other classes into one large class, usually called others, and by developing a binary classifier to discriminate the class of interest from the others. Here, this approach is called focused approach. The strength of the focused approach is to decompose the original multi-class supervised classification problem into a binary classification problem, focusing the process on the discrimination of the class of interest. Previous studies have shown that this method is able to discriminate more accurately the classes of interest when compared with the standard multi-class supervised approach. However, it may be susceptible to data imbalance problems present in the training data set, since the classes of interest are often a small part of the training set. A result the classification may be biased towards the largest classes and, thus, be sub-optimal for the discrimination of the classes of interest. This study presents a way to minimise the effects of data imbalance problems in specific class mapping using cost-sensitive learning. In this approach errors committed in the minority class are treated as being costlier than errors committed in the majority class. Cost-sensitive approaches are typically implemented by weighting training data points accordingly to their importance to the analysis. By changing the weight of individual data points, it is possible to shift the weight from the larger classes to the smaller ones, balancing the data set. To illustrate the use of the cost-sensitive approach to map specific classes of interest, a series of experiments with weighted support vector machines classifier and Landsat Thematic Mapper data were conducted to discriminate two types

of mangrove forest (high-mangrove and low-mangrove) in Saloum estuary, Senegal, a United Nations Educational, Scientific and Cultural Organisation World Heritage site. Results suggest an increase in overall classification accuracy with the use of cost-sensitive method (97.3%) over the standard multi-class (94.3%) and the focused approach (91.0%). In particular, cost-sensitive method yielded higher sensitivity and specificity values on the discrimination of the classes of interest when compared with the standard multi-class and focused approaches.

## 4.1 Introduction

Supervised classification has become an important method to derive land cover information from remotely sensed imagery [107]. One significant advantage of supervised classification is that it allows tailoring the classification process in order to obtain a map depicting only the classes of interest [55]. Indeed, users are often not interested in a complete characterisation of the landscape but rather on a sub-set of the classes existing in the study area. For example, the analysis may have to be focused on mapping urban classes [24, 39], abandoned agriculture [3], specific tree species [5, 54], invasive wetland species [87], and mangrove ecosystems [90, 140]. Fundamentally, the accurate discrimination of some classes is more important than the discrimination of others for some applications.

When users are only interested in a sub-set of the classes present in the study area, the use of conventional multi-class supervised classification may be sub-optimal for the purpose [45]. One of the reasons for this situation has to do with the classification algorithm fine-tuning process. This procedure, necessary in many classification algorithms, consists of finding the parameterisation that yields the maximum overall classification accuracy, that is to find the parameterisation that best discriminates all classes of the classification problem [61]. The common approach often seeks, by cross-validation grid-search, to maximise the overall classification accuracy, rather than the specific accuracy in the classification of particular classes. However, the parameterisation that yields the highest overall classification accuracy may not be necessarily the best to discriminate the classes of interest, since these are usually only a small part of the problem [88]. Indeed, overall accuracy is only one component of classification quality assessment and may not be suited to the requirements of a particular study [88]. Thus the conventional multi-class supervised classification algorithm is neither tuned nor trained to discriminate the classes of interest, since the class composition of the training set contains all classes regardless of their interest in the analysis and the tuning process searches for the best parameterisation in that larger problem.

The literature shows that there are essentially two alternatives to the standard multi-class supervised approach: one-class learning algorithms and the binarisation strategy [57, 79, 136].

With the one-class learning algorithms, the user adopts a one-class learning algorithm to develop a classifier to identify a single class of interest [e.g. 101, 125]. In this approach only training data belonging to the class of interest is utilised to develop the classifier, which is its most attractive feature in terms of focusing effort and resources on the class of interest. However, the one-class classifier may not always be the best approach, since only data about one class is available and thus only one side of the discriminative boundary can be determined [136]. It can then be difficult to determine how tightly the boundary should fit in all directions around the data in feature space. To overcome this difficulty some one-class classifiers (e.g. support vector data description) assume that the non-interest class has a particular distribution around the class of interest. When the true distribution deviates from the assumption, the method may underperform. That deviation however can only be assessed with training points outside of the class of interest [136].

With binarisation strategy, users decompose the multi-class problem in a series of small binary classification problems where one seeks to separate the classes of interest from all irrelevant classes [13, 41, 57, 82]. As binary classification is well-studied, binary decomposition of multi-class classification problems have attracted significant attention in machine learning research and has been shown to perform well in most multi-class problem [82]. Indeed, binary decomposition has been widely used to develop multi-class Support Vector Machines (SVM) showing better generalisation ability than other multi-class SVM approaches [65]. The possibility to parallelise the training and testing of the component binary classifiers is also a big advantage in favour of binarisation apart of their good performance [57]. In particular, binarisation can be achieved by combining all land cover classes of no interest into a large nominal class, called for example "others" [47]. In this way the class of interest can be regarded as the positive class and all others as the negative class in the binary classification scenario. Previous studies [13, 47, 90] have shown it to be possible to decompose the multi-class classification problem in a series of small binary classification problems and achieve results that are more suitable for the particular users' requests, namely the improvement of the discrimination of particular land cover classes of interest. Although specific class mapping can potentially be a better approach compared to the multi-class supervised classification, it has some particular difficulties, namely data imbalance in the training set. This is because often the classes of interest are only on a small component of the study area [88]. In fact, applying directly a binary decomposition to the classification problem may result in a highly unproportional allocation of training points to the negative class, leading to imbalance in the training data set [9].

Learning from imbalanced data sets is an important and challenging problem in knowledge discovery in many real-world applications [62]. Learning from imbalanced data means learning from data in which the classes have unequal numbers of training data points [63]. Although there are several degrees of data imbalance, there is

no agreement or standard concerning the exact degree of class imbalance required to have a negative effect in the learning process. The central issue with learning from imbalanced data sets is the effect of this condition on the performance of most standard learning algorithms [78]. Indeed, most learning algorithms aim to derive the simplest classifier that best fits the training data; this can represent a serious challenge to the development of classifiers with imbalanced data, since such classifier is often biased towards the majority class [41, 70]. For example, a classifier that omits a large proportion of the minority class cases can yield high overall accuracy, although it may underperform in the discrimination of that class. When trained with this type of data sets, learning algorithms usually fail to accurately learn the distributive characteristics of the data and, consequently, may provide inaccurate results [98]. A balanced data set is, therefore, a desirable feature of the training set.

In general, the methods to mitigate the effect of imbalances in data sets consist of either methods that manipulate data by oversampling the minority class or under-sampling the majority class, or methods that adapt the algorithm to the imbalance condition [78]. Data manipulating approaches can be problematic, since under-sampling may remove important data points for the discrimination of the classes [22] and over-sampling may render longer training time and over-fitted classifiers [62, 121]. The methods that adapt the learning algorithm to the imbalance condition seek to bias towards the minority class [145]. These methods are commonly known as cost-sensitive learning [61].

In cost-sensitive learning, misclassifications are not treat equally. Data points are assigned a weight representing their relative value: more weight accredits more value. By assigning more weight to a particular data point than to another, the analyst is highlighting its relative importance, and thus informing the learning algorithm that an error in the former is costlier than an error in the latter [145]. This additional information directs the learning process to the under-represented classes and thus minimises the effect of learning in imbalanced datasets. In this paper a support vector machine classifier is used to demonstrate the use of cost-sensitive learning to minimise the effects of data imbalances in specific class mapping.

Although data imbalance in the training set has been recognised as an important factor in the learning process and is common in natural resource applications using remotely sensed data [105], little attention has been given to its effects and errors in land cover mapping. Thus studies reporting its effects, or estimating its effects from previous studies, are rare in literature. Examples addressing data imbalance in remote sensing have been reported mostly in tree species classification problems. In [6] authors explore the use of standard SVM and biased SVM classification of three tropical tree species using airborne imaging spectroscopy. To mitigate the effects of data imbalance, the authors carefully tuned the classification algorithm using the harmonic mean between sensitivity and specificity of the classes of interest, also known as $F$-score. In [59] authors examine the effects of data imbalance in the supervised

classification of tree species in eight reported studies and address the problem in a twenty-class classification problem. The authors conclude that species with more training data points were consistently over-predicted while species with fewer data points were under-predicted. In [131], authors explore the multiple classification methods for tree species identification in temperate forests using Formosat-2 satellite image time series, reporting that minority classes were often the most confused. Thus, data imbalance problems are occurring in application studies where classifications are being used to infer information about land cover.

In this study to demonstrate the effects of data imbalance in the training set and how to mitigate them using cost-sensitive learning, two experiments were conducted: first, artificially generated data set was used to illustrate the effects of data imbalance in the development of a classifier.

Second, a series of experiments are presented in a study area located in the Saloum estuary, Senegal. Two land cover classes were defined as the classes of interest. These were classes of mangrove forest that differ in height: high-mangrove and low-mangrove. The distinction between these two classes is important since the transition from high-mangrove to low-mangrove is often a symptom of mangrove degradation [31, 140].

Three classification approaches are explored: a standard multi-class, a focused and a cost-sensitive approach to classification. In the standard multi-class approach, a single algorithm is used to solve a multi-class classification problem. The classes of interest are, in this case, derived after the classification process. In the focused approach, all classes of no interest are combined in one single nominal class (others). The classes of interest are derived in the classification process but nothing is done to mitigate possible class imbalances in the data set. In the cost-sensitive approach, similarly to the focused approach, all classes of no interest are merged into one large class "others" but here weights are utilised in each training data point to inform the learning algorithm of the relative misclassification cost value.

The innovations presented in this article are three-fold: first, cost-sensitive learning is presented as a way to mitigate problems associated with the use of an imbalanced training set in specific class mapping. In other words, this applicational study intents not only to show that imbalance data sets can undermine the mapping process, but also to show that cost-sensitive learning can minimise its effects. Second, three classes were used, the two classes of mangrove constituting the classes of interest and the class "others". This is relevant since the definition of more than one class of interest requires a process to combine the different outcomes of several binary classifiers which is not always trivial and was not fully addressed in previous studies, that have typically focused on a single class. Third, it is shown that the classifier parameterisation is an important step in specific class mapping and more accurate classifiers can be obtained using class specific metrics instead of an overall classification metrics, as is commonly utilised.

## 4.2 Classification with imbalanced data sets

Learning with an imbalanced data set is one of the most challenging problems in many real-world applications and it has been recognised as a crucial problem in machine learning and data mining [16, 22]. Class imbalance problems may occur when the training set is not evenly distributed among the classes [22]. There is no agreement, or standard, concerning the exact degree of class imbalance required for a dataset to lead to a biased classifier [63]. This uneven condition is usually quantified by the ratio between the size of the minority class and the size of the majority class, usually called balance ratio [141]. Data set balance ratios can vary greatly, for example from 1:1 (balance data set) to extreme cases such as 1:100 or more [e.g. 141]. In [142], a 26 binary-class datasets were analysed showing how class imbalance impacts minority class classification performance. The results suggest that class imbalance leads to poorer performance when classifying data points belonging to the minority class. Geometrically, a classifier developed with a imbalanced training set pushes the discrimination boundary away from the majority class, bring it closer to the minority class [63]. This happens because by pushing the boundary away from the majority class toward to minority class, the number of misclassifications on the majority class are minimised, which is the term that contributes the most for the overall classification error. This impact can be quite severe, as datasets with class imbalances between 1:5 and 1:10 can have a minority class error rate more than 10 times that of the error rate on the majority class [142]. This suggests that datasets with even moderate levels of class imbalance (e.g. 1:2) can suffer from class imbalance issues [63].

Most classifiers assume the classes present in the training set contain the same or similar number of data points [145]. Since classification algorithms are designed to generalise from data and output the simplest classifier that best fits the training data, classifiers will then typically seek to maximise overall accuracy, and thus tend to underperform on imbalanced data sets [2].

The methods to address the problem of imbalanced training data sets can be grouped into two categories: methods focusing on the data and methods focusing on the classification algorithm [78]. The first group of methods attempt to solve the problem of imbalanced training data sets by purposely manipulating the classes' distributions in the training data set either by over-sampling the minority class or by under-sampling the majority class [120]. In other words, in these methods data points are added to the minority class or removed from the majority class to balance the training set. There are however some issues with these procedures. Over-sampling may, for example, render longer training time and over-fitted classifiers [62, 121]. Since over-sampling, at its simplest way, appends replicated data to the original data set, the algorithm may become too specific and may not generalise well [70]. Under-sampling, on the other hand, may remove important data points for class discrimination [22]. The methods on the second group, on the other hand, adapt a classification algorithm

to bias towards the minority class, for example defining a cost function that penalises more misclassifications committed on data points of the minority class. The training data set is then balanced by shifting the weight of the training set from the larger classes to the smallest. These methods are generally named as cost-sensitive learning methods [145]. A way to implement the cost-sensitive approach is by incorporating the weight of data points Weighted Support Vector Machines (WSVM) classifier [145].

### 4.2.1 Weighted support vector machine

The SVM is a popular supervised classification algorithm that has been successfully applied in many domains [129]. In particular, in the classification of remotely sensed imagery, the study and application of SVM is extensive and well known [107]. In its origin, the SVM was developed to solve binary classification problems with linearly separable classes. However, SVM was extended with the introduction of the kernel trick and slack variables to solve non-linearly separable classes [30]. The use of kernels allowed the SVM to solve non-linear problems by mapping the original data points into a higher dimensional space where a linear classifier is able to discriminate them [130]. The introduction of slack variables, on the other hand, relaxed the original SVM optimisation problem; a non-zero slack variable allows a particular data point to not meet the margin requirement at a cost proportional to its magnitude, allowing some training data points to be misclassified [145]. This version is usually known as soft-margin SVM. The corresponding optimisation problem is formulated as follows [130]:

$$\min_{\boldsymbol{w},\xi} \frac{1}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w} + C\boldsymbol{e}^{\mathrm{T}}\boldsymbol{\xi} \tag{4.1}$$

subject to $y_i(\boldsymbol{w}^{\mathrm{T}}\phi(\boldsymbol{x}_i) + b) \geq 1 - \xi_i$ for $i = 1 \ldots m$ where $m$ is the number of training data points, $\boldsymbol{w}$ is the hyperplane normal vector, $\boldsymbol{\phi}$ is the kernel function, $\boldsymbol{e}$ is the all 1's vector and $\boldsymbol{\xi}$ is the vector of slack variables. The parameter $C$ represents the magnitude of penalisation. If $C$ is a large value, the optimal solution defines narrower margins in order to accommodate the misclassified training data points; in contrast, smaller values of $C$ lead to wider margins [127]. Applying the Karush-Kuhn-Tucker (KKT) conditions, the original soft-margin SVM problem is usually reformulated in its Lagrangian dual form [130]:

$$\min_{\alpha} \frac{1}{2}\sum_{i,j}\alpha_i\alpha_j y_i y_j K(\boldsymbol{x}_i,\boldsymbol{x}_j) - \sum_i \alpha_i \tag{4.2}$$

subject to $\sum_i y_i\alpha_i = 0$ and $0 \leq \alpha_i \leq C$ for $i = 1 \ldots m$, where $K(\boldsymbol{x}_i,\boldsymbol{x}_j) = \boldsymbol{\phi}^{\mathrm{T}}(\boldsymbol{x}_i)\boldsymbol{\phi}(\boldsymbol{x}_j)$, quantifies the similarity between two arbitrary training data points, $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ in the kernel space.

Note that under these conditions, the Lagrange multipliers are bounded by the parameter $C$ and thus all misclassifications of training cases are penalised in the same

amount. This might not be appropriate especially if the data set is imbalanced. For example, when trained with imbalanced data sets in which the number of negative instances outnumbers the positive instances, the performance of SVM may drop significantly [146]. Indeed, SVM may end up classifying all testing data set as belonging to the majority class [147]. The optimisation problem (4.2) tries to minimise first term, responsible to maximise the margin between the support vectors, and the second term, responsible to minimise the number of misclassified training cases. The regularisation parameter $C$ defines the trade-off between maximising the margin and minimising the classification error in the training set [67]. Thus, if $C$ is not large enough, SVM learns to classify everything as belonging to the negative class, since that makes the margin larger with maximum accuracy in the training set [145].

A way to adapt the SVM approach to cost-sensitive learning is by increasing the trade-off parameter $C$ associated to the minority class [67, 145]. With the WSVM each data point is assigned a particular weight value; this weight is usually associated to some class characteristic such as size [67]. The original SVM problem is then reformulated in the following way:

$$\min_{\boldsymbol{w},\xi} \frac{1}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w} + C\boldsymbol{\sigma}^{\mathrm{T}}\boldsymbol{\xi} \tag{4.3}$$

subject to $y_i(\boldsymbol{w}^T\phi(\boldsymbol{x}_i)+b) \geq 1-\xi_i$ for $i = 1\ldots m$, where $\boldsymbol{\sigma}$ is the vector of weights. The user can then set different weights to different data points according to a predetermined criterion. Applying the KKT conditions, the original WSVM problem can be reformulated in its dual form:

$$\min_{\alpha} \frac{1}{2}\sum_{i,j}\alpha_i\alpha_j y_i y_j K(\boldsymbol{x}_i,\boldsymbol{x}_j) - \sum_i \alpha_i \tag{4.4}$$

subject to $\sum_i y_i\alpha_i = 0$ and $0 \leq \alpha_i \leq C\sigma_i$ for $i = 1\ldots m$. Note that, unlike problem (4.2), the Lagrange multipliers are now bounded according to its weight. For imbalanced classification problems, many studies [e.g. 33, 66, 67, 97, 145] have defined the data points weight by the inverse of its correspondent class size. In this way, the misclassifications of elements belonging to the majority class receive proportionally less importance than those belonging to the minority class. Note that if data set is balanced, the number of negative data points equals the number of positive data points. Thus the WSVM with this weighting rule reduces to non-weighted SVM.

### 4.2.2 Combining binary classifiers

Like SVM, WSVM is at its core a binary classifier. If one wants to apply the WSVM to a multi-class problem, the two more common strategies are [20, 57]: One-vs-Rest (OVR) (figure 4.1 frames $(a)$, $(b)$, $(c)$ and $(d)$) One-vs-One (OVO) (figure 4.1 frames $(e)$, $(f)$, $(g)$ and $(h)$).

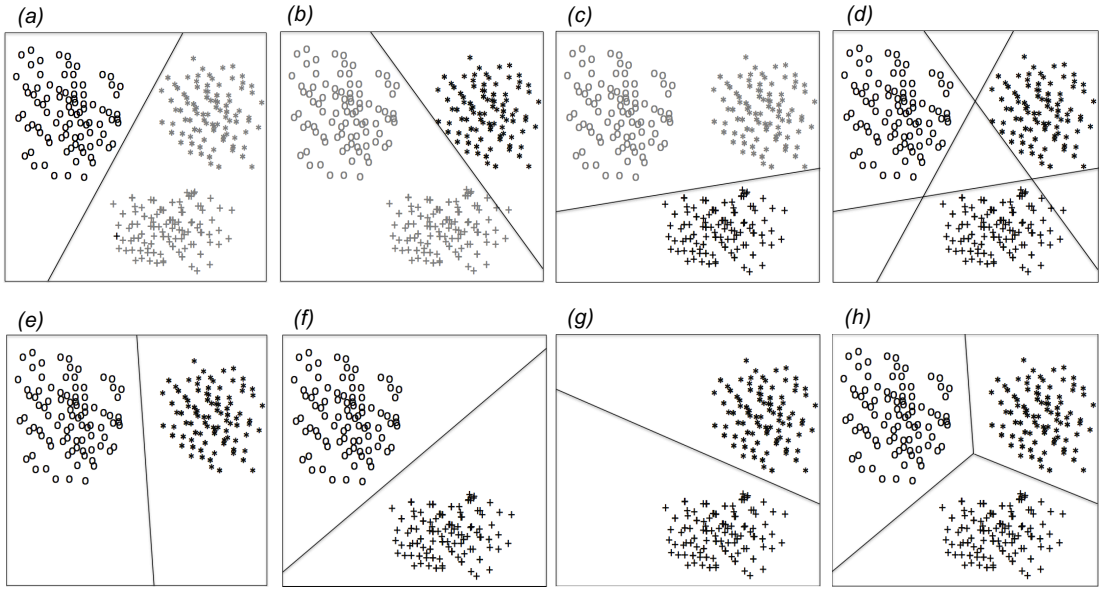



Figure 4.1: Binary decomposition of multi-class problem. The frames represent the scatter plot in the feature space of three different classes: circles, stars and crosses. Top row: OVR strategy. Bottom row: OVO strategy.

The OVR strategy breaks the multi-class classification down into a series of binary classification problems where each class is in turn compared with all others [129]. In this way, a $N$-class classification problem is decomposed into $N$ binary classification problems. For example, in a three-class classification problem, a first classifier is developed to discriminate the class in black (frame $(a)$) from all other classes are combined into a single class, in grey. The process is then repeated for the other two classes (frames $(b)$ and $(c)$). The final step is then performed either by assigning the class with positive outcome or by selecting the class with the largest decision value [123] (frame $(d)$). However, if the label-assigning rule is not based on the decision value directly, some data points may not be classified, because it is possible for a point to be rejected from all classes [129]. The OVR strategy is may be susceptible to class imbalances even if the training set is balanced, since the negative class is effectively composed by all other classes combined into one large class [9].

The OVO strategy is also known as all-pairs strategy, as it consists in enumerating all possible pairs of classes (frames $(e)$, $(f)$ and $(g)$) and then to develop a binary classifier for each pair of classes [129]. Classification is then done by inputting the data point into each particular binary classifier and labelling by majority voting. In this way, if there are $N$ classes, the number of binary classifiers is then $\frac{1}{2}N(N-1)$ [129] (frame $(h)$). Although the number of binary classification problems is of the order $N^2$ and may represent a significant memory requirement this solution, it may also provide simpler models (less support vectors), and thus improve generalisation [30]. Which strategy is the best is a still an on-going debate [20, 57].

Table 4.1: Binary confusion matrix.

| | Actual positive | Actual negative |
|---|---|---|
| Predicted positive | $TP$ | $FP$ |
| Predicted negative | $FN$ | $TN$ |

### 4.2.3 Comparison and evaluation of classifiers

The design and implementation of a learning algorithm require the use of accuracy metrics to assess the quality and compare the performance of alternative classifiers. For example, when fine-tuning a classification algorithm, it is often necessary to compute an accuracy metric to determine the parameterisation that yields on average the highest accuracy value. Although commonly used, the overall classification accuracy (the proportion of correctly classified data points) may not a reliable metric when the training set is imbalanced. This is because the majority class dominates the behaviour of this metric, and thus it gives optimistically biased results [145]. Indeed, the definition of the accuracy metric is particular important for binary classification, since the performance of the classifiers can be particularly sensitive to the classes' relative size [129, 145]. In this conditions, the results of the fine-tuning process may be unreliable not because of the process but rather because of the accuracy metric employed in the process. If the training data set is unbalanced and the classification accuracy is utilised, the outcome of the fine-tuning process will indicate that a particular parameterisation is the one with the highest classification accuracy but may indeed biased towards the majority class, since that parameterisation may yield a classifier that classifies very accurately the majority class in detriment of the minority class [67]. There are better alternative accuracy metrics to the classification accuracy specially when the data set is imbalanced, for example sensitivity and specificity [61]. At the basis of this analysis is the binary confusion matrix (table 4.1).

In table 4.1, True Positive (TP) represents the number of actual positive cases correctly classified, True Negative (TN) the number of actual negative cases correctly classified, False Positive (FP) the number of actual negatives predicted as positives, and False Negative (FN) the number of actual positives predicted as negatives. The classification accuracy is then the proportion of true positives and true negatives which is commonly used to metric classification performance in multi-class problems [145]. But in binary classification, classification accuracy may not be a reliable indicator particularly if the data set is imbalanced, since the influence of the majority class is much higher than that of the minority class [67]. Alternatively, other quality metrics can be used, such as sensitivity and specificity [145]. Sensitivity is the proportion of true positives correctly classified while specificity is the proportion of true negatives correctly classified [61]. Effectively, sensitivity is the producer's accuracy of the positive class and specificity is the producer's class of the negative class. In this way, sensitivity

indicates how good the classifier is recognising positive cases and specificity indicates how good the classifier is recognising negative cases [145].

Often sensitivity and specificity are combined in one metric for better analysis and comparison [135]. In particular, the geometric mean between sensitivity ($s$) and specificity ($S$) [84] in Equation 6.6 is particularly useful:

$$G = \sqrt{sS} \tag{4.5}$$

The geometric mean ($G$) indicates the balance between classification performances on the positive and negative class. High misclassification rate in the positive class will lead to a low geometric mean value, even if all negative data points are correctly classified [67]. This is a desirable feature specially when the testing sample is asymmetric. Indeed, it can be prove that, in a binary classification scenario, classification accuracy is the weighted average between sensitivity and specificity, where the weights are the proportion of each class in the sample. For example, if 10% of the sample is in the positive class and 90% is located in the negative class, a classifier that simply classifies every data point as belonging to the negative class yields an overall accuracy of 0.90. However, its sensitivity is 0.0 and specificity 1.0, and thus geometric mean $G$ is 0.0. In this way, if both sensitivity and specificity are high, the geometric mean $G$ is also a high value; but if one of the component accuracies, sensitivity or specificity, is low, the geometric mean $G$ is affected by it. Note that in some cases a testing sample has to be asymmetric, that is, one class has more testing data points than the other, simply due to its variability. This is the case in a class specific mapping problem, where the majority of the study area is typically outside the class of interest and thus contains all other classes. Thus, the geometric mean can be an important accuracy metric for class specific mapping, since it is particularly sensitive to the over-fitting to the negative class (i.e. others class) and to the degree in which the positive class (i.e. class of interest) is neglected [110].

## 4.3 Data and methods

The study area is located in Saloum river delta in Senegal, Africa (figure 6.1). The area is predominantly flat with altitudes ranging from below sea level in the estuarine zone to about 40 m above mean sea level inland. The climate is Sudano-Sahelian type with a long dry season from November to June and a 4-month rainy season from July to October [31, 38]. The regional annual precipitation, which is the main source of freshwater recharge to the superficial aquifer, increases southward from 600 to 1000 mm. The hydrologic system of the region is dominated by the river Saloum, its two tributaries (Bandiala and Diomboss), and numerous small streams locally called “bolons”. Downstream, it forms a large low-lying estuary bearing tidal wetlands, a mangrove ecosystem, and vast areas of denuded saline soils locally called “tan” [31]. The largest

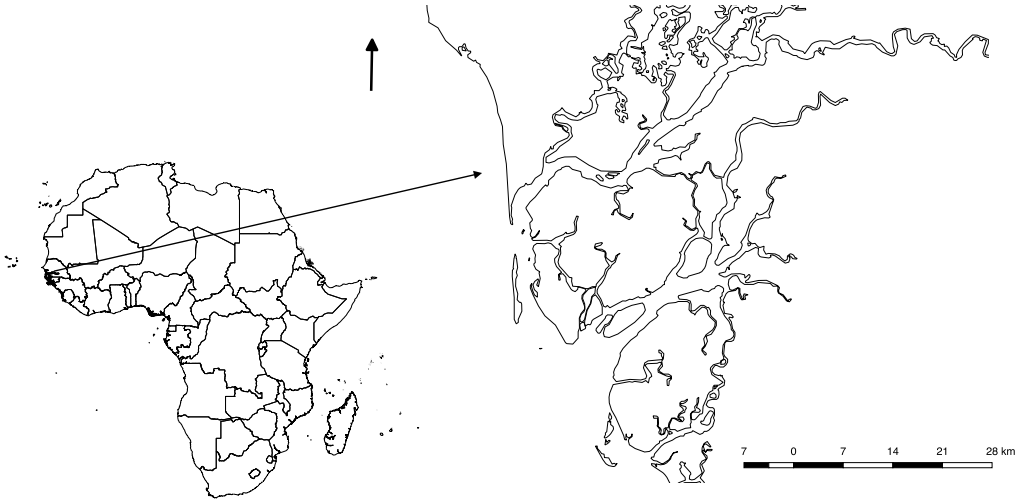



Figure 4.2: Saloum river delta in Senegal.

land cover classes present in the study area are water, mangrove species, shrubs, savannah and bare soil. The main crop is millet and the urban settlements are usually small and sparse. Saltpans develop to the north because of excessive salinity [106]. In this paper interest is focused on two types of mangrove, High Mangrove (HM) and Low Mangrove (LM). HM is generally characterised by a dense and tall canopy, while LM tends to show less dense and decayed canopy. In this study area, HM class is composed by species like *Rhizophora racemose*, *Rhizophora mangle* and *Avicennia Africana* [32], and LM by *Sesuvium portulacastrum*, *Sporobolus robustus*, *Paspalum vaginatum*, and *Philoxerus vermicularis* [32].

The Saloum river delta was designated a United Nations Educational, Scientific and Cultural Organization (UNESCO) World Heritage site for its remarkable natural environment and extensive biodiversity and is listed in the Ramsar List of Wetlands of International Importance [106]. Particularly important is Saloum's mangrove system, occupying roughly 180 000 ha supporting a wide variety of fauna and flora, and the local economy [106].

Remotely sensed data of the study area were acquired on 26 November 2010 by Landsat 5 Thematic Mapper (TM) and downloaded from United States Geological Survey (USGS) Global Visualisation web site. In this study all non-thermal bands (bands 1 to 5 and 7) have been used. Since only one image was utilised for analysis and the atmosphere may be considered to be homogeneous within the study area, atmospheric correction was not necessary [132] and, thus, the classification was performed using the original image digital numbers. In the same year of the image acquisition, fieldwork and aerial imagery interpretation were undertaken to derive ground-reference data. This analysis showed that the study area is composed by six large land cover classes: water, high-mangrove, low-mangrove, bare soil, savannah and shrubs. The training set comprised of 180 pixels per class (= 30 times the number of discriminatory

Table 4.2: Summary of the different experiments: experiments with (*) indicate benchmark. Exper. stands for experiment, Imbal. for Imbalanced, Cost-sens. for Cost-sensitive and Strat. for strategy. SSVM represents the standard use of SVM; FSVM represents the focused approach with SVM; FOVO represents the focused approach with cost-sensitive and OVO; and FOVR the focus approach with cost-sensitive and OVR.

| Exper. | Training set | Fine-tuning | Imbal. | Cost-sens. | Strat. |
|---|---|---|---|---|---|
| SSVM(*) | All classes | General | No | No | OVO |
| FSVM(*) | HM, LM, O | Specific | Yes | No | OVR |
| FOVO | HM, LM, O | Specific | Yes | Yes | OVO |
| FOVR | HM, LM, O | Specific | Yes | Yes | OVR |

variables) for each of the six land cover classes.

Four experiments were conducted to demonstrate the effects of data imbalance and the use of cost-sensitive learning. The first two experiments were used as benchmark. Table 4.2 summarises the different experiments carried out in this study.

The first benchmark classification constitutes the conventional approach to supervised classification, when interest in on a sub-set of classes present in the study area. In other words, a multi-class supervised classification is performed to obtain a land cover map with all classes, and then only the classes of interest are used. A Standard Support Vector Machines (SSVM) was trained using all six classes and fine-tuned for general class discrimination. The training set was balanced over all six classes and thus cost-sensitive methodology was not applied. The radial-basis function was chosen as kernel and it was used in all the tested approaches. The free-parameters $C$ and $\gamma$ of the radial-basis function were determined using a 5-fold cross-validation grid-search with overall accuracy as performance metric. In this way the fine-tuning process is effectively searching for the parameterisation with the highest overall accuracy regardless of the classes. From this analysis the parameters were set as $\gamma = 0.00097$ and $C = 64$. The experiment was conducted using LibSVM-3.12 [20] software interfaced with MATLAB®. This software package implements the standard SVM, unweighted analysis, with the OVO strategy for multi-class problems [20].

The second benchmark constitutes the focused approach to map specific classes without taking into account the data imbalances present in the training set. This benchmark classification used the standard SVM [e.g. 13]. For this reason this approach is named where as Focused Support Vector Machines (FSVM). All non-mangrove classes were combined into a large class called "others" for use in the training stage. The training set in the analysis is thus composed of three classes: HM, LM and others (O) class. In this way three binary classifiers were developed, each one focusing in the discrimination of one particular class. The geometric mean between sensitivity and specificity was applied in 5-fold cross-validation trials for fine-tuning. Table 4.3 summarises the parameterisations derived from the fine-tuning analysis and shows

Table 4.3: Parameterisation using focused approach.

| Positive | Negative | $\gamma$ | $C$ | Balance ratio |
|---|---|---|---|---|
| HM | LM + O | 0.03125 | 4 | 1:5 |
| LM | HM + O | 0.00195 | 0.0625 | 1:5 |
| O | LM + HM | 0.00391 | 0.125 | 2:1 |

the balance ratios to quantify the size difference present in each pair of classes. The balance ratio is the ratio between the sizes of each of the pair. For example, the balance ratio between high-mangrove (180 data points) class and rest of the training set (900 data points) is 1:5. To combine the different outcomes of each classifier, and to avoid non-labelled data points, the assigned label was that of the class with maximum decision value [129]. These experiments were conducted with the same software package as in previous experiment.

In contrast with the previous experiment, the fine-tuning does not take the overall classification accuracy as metric but rather the geometric mean between sensitivity and specificity, which is specific of each target class. In this paper, and for clarity, any fine-tuning process that takes into account the overall classification accuracy and not the classification of specific classes will be qualified as general, and specific otherwise.

It is important to note that the two benchmarks have their own specific limitations. The first benchmark, although widely used, is not optimised for the discrimination of the classes of interest, since the learning algorithm is evaluated on a different class composition than that with was tuned and trained. The second benchmark is an improvement over the first, suggested in previous studies. But, this leads to an classifier developed with an imbalanced data set, which may bias the analysis to the larger classes. Thus the first benchmark, while developed with a balanced training data set, was neither tuned nor trained to discriminate the classes of interest; and the second benchmark, while trained and tuned to discriminate the classes of interest, suffers the effects of training data imbalances. The remain approaches tackle these two problems. In other words, they tackle the class specific mapping while avoiding possible data imbalances issues using cost-sensitive learning.

To that end, data point weights were defined as the inverse of its training set size, similar to what has been applied in other studies, such as [145]. In this way by assigning more weight to the data points in the smaller classes, the training set weight distribution shifts from the largest class to the smallest classes minimising the bias towards larger classes. Two approaches were then analysed, one using OVO strategy and another using OVR (table 4.4).

The approach using OVO was named Focused One-vs-One (FOVO) and the one using OVR was named Focused One-vs-Rest (FOVR). In these approaches, all classes of no interest were combined into a large one, and thus the training set consisted in only three classes, HM, LM and others class. Fine-tuning was specific to each binary

Table 4.4: Parameterisation and weights for each pair of classes: using OVO strategy and using OVR strategy.

| Method | Positive | Negative | Weights (+, -) | $\gamma$ | $C$ | Balance ratio |
|---|---|---|---|---|---|---|
| OVO | HM | LM | 0.0056, 0.0056 | 0.00012 | 1024 | 1:1 |
| | LM | O | 0.0056, 0.0014 | 0.00012 | 128 | 1:4 |
| | O | HM | 0.0014, 0.0056 | 0.00098 | 2 | 4:1 |
| OVR | HM | LM + O | 0.0056, 0.0011 | 0.06250 | 256 | 1:5 |
| | LM | HM + O | 0.0056, 0.0011 | 0.00391 | 8 | 1:5 |
| | O | HM + LM | 0.0014, 0.0028 | 0.00098 | 4 | 2:1 |

classifier and the training data set was imbalanced and cost-sensitive analysis was employed.

HM and LM classes have the same amount of data points (table 4.4 – Balance ratio), thus the weights associated to their data points is equal, 0.0056. The others class is the majority class, and the weight associated to its data points is thus comparatively smaller to those of high-mangrove and low-mangrove, 0.0014. The free parameters were fine tuned using 5-fold cross-validation trials and the experiments were conducted with LibSVM-weights-3.12 [20] interfaced with MATLAB®.

Classification accuracy was estimated using an independent testing set of 100 random pixels per land cover class comprising a total of 600 pixels. An image analyst visually classified each pixel in the same year as the image acquisition with support of Google Earth and fieldwork data. The accuracy of each classification was expressed in terms of the proportion of correctly classified testing data points. Since a single testing set was used for each test site, the statistical significance of the difference in overall accuracy between different classification approaches will be assessed using the McNemar test [49].

The McNemar test is based on a binary contingency table in which pixels are classified as correctly or incorrectly allocated by the two classifiers under comparison. The main diagonal of this table shows the number of pixels on which both classifiers were correct and on which both classifiers were incorrect. The McNemar test however focus on proportion of pixels where one classifier was correct but the other was incorrect. The analysis will be based upon the evaluation of the $100(1-\alpha)\%$ confidence interval, where $\alpha$ is the level of significance, for the difference between two accuracy values expressed as proportions (say $p_1$ and $p_2$) expressed as [42]:

$$p_2 - p_1 \pm z_\alpha s \tag{4.6}$$

where the term $s$ is the standard error derived of the difference between the proportions, which can be determined by [42]:

$$s = \sqrt{\frac{p_{01} + p_{10} - (p_{01} - p_{10})^2}{n}} \tag{4.7}$$

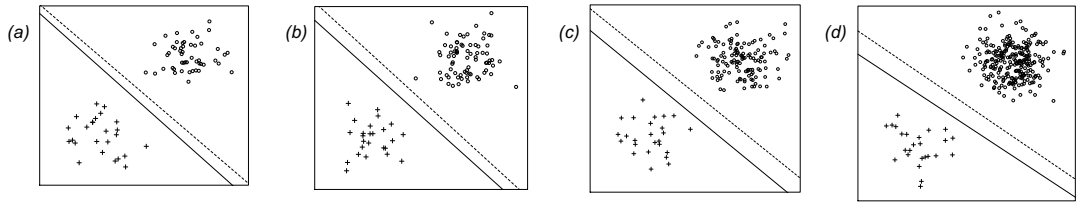



Figure 4.3: Illustration of the effects of data imbalance in the training data set with different degrees of balance ratios. The data set was generated artificially and represents a purposely simple classification problem, projected in the feature space. The minority class represented with crosses and the majority class represented with circles. Straight line is the discrimination plane generated with the non-weighted approach. Dashed line is the discrimination plane generated with the weighted approach. In frame ($a$) balance ratio is 1:2, in frame ($b$) is 1:3, in frame ($c$) is 1:5 and in frame ($d$) is 1:10.

here $p_{10}$ the proportion of testing pixels where the first classifier was correct and the second was incorrect and $p_{01}$ the proportion of testing pixels where the first classifier was incorrect and the second was correct. In this way, the statistical assessment of the differences was conducted to determine if these were significantly different or not [49].

## 4.4 Results and discussion

A sequence of experiments with synthetic data were performed to illustrate the effects of data imbalance in the resulted classifiers. For this purpose two normal distributed classes, the circles and crosses, were artificially generated with the same variability but different class sizes. The classes are linearly separable and thus a linear SVM classification algorithm is capable of developing a classifier without errors in the training data set. That is, it is able to find the optimal discrimination plane. The example is purposely simple to illustrate the effects of data imbalances in the training set. In other words, in real-world applications the relative size between classes is not the only factor contributing to the classification algorithm. The class mean location, variance and overlapping for example are also important informing the learning algorithm.

In figure 4.3, the effects of data imbalance in the training data set are observed with different balance ratios. The minority class represented with crosses and the majority class represented with circles. In frame ($a$) balance ratio is 1:2, in frame ($b$) is 1:3, in frame ($c$) is 1:5 and in frame ($d$) is 1:10. Straight line is the discrimination plane generated with the non-weighted approach and dashed line is the discrimination plane generated with the weighted approach. When data sets are not balanced, the discrimination boundary (straight line) is pushed away from the majority class. This gives more room to the majority class to accommodate atypical pixels, that is pixels with low frequency of occurrence or that were not represented in the training data set. However,

Table 4.5: Summary of the accuracy results in percentage obtained with each experiment. OA stands for overall accuracy, Ss. for sensitivity and Sp. for specificity for each class of interest.

| | | High-mangrove | | Low-mangrove | |
|---|---|---|---|---|---|
| Method | OA (%) | Ss (%) | Sp (%) | Ss (%) | Sp (%) |
| SSVM | 94.3 | 88.0 | 95.6 | 85.0 | 96.2 |
| FSVM | 91.0 | 86.0 | 92.9 | 72.0 | 94.8 |
| FOVO | 97.3 | 95.0 | 97.8 | 93.9 | 98.2 |
| FOVR | 96.7 | 93.0 | 97.4 | 91.0 | 97.8 |

the decision boundary is closer to the minority class, providing less room to accommodate pixels that deviate from the training data set distribution. Thus, the classifier is overfitted around the minority class. In other words, a point belonging to the minority class that deviates from the training data set distribution may be misclassified, because the discrimination boundary is too close to its true class. Thus, a classifier developed with an imbalanced data set may induce a classification with high number of false negatives in the minority class. That is, the minority class may be underestimated. This explains the findings of previous studies, like [59], that have shown a trend where classes with more samples were consistently over-predicted while classes with fewer samples were under-predicted. With the discrimination plane induced by the weighted approach, the effects of data imbalanced are mitigated. The training data points were weighted according to its class, using the same rule as presented in section 4.2.1. Here the decision boundary is further from the minority class compared to the plane induced by the non-weighted approach (straight line). This provides enough room to include atypical pixels, thus mitigating the effects of the overabundance of data points belonging to the majority class. In this way, by controlling the weight of the minority class data points, it is possible to inform the learning algorithm to push away the decision boundary to avoid over-fitting around the minority class.

The overall accuracy yielded by the two benchmarks was 94.3% and 91.0% for SSVM and FSVM, respectively (table 4.5). The difference in overall accuracy between these two approaches can be attributed mainly to the data imbalance present in the training set used in the FSVM experiment. Indeed, the training set used for SSVM is balanced since the six classes have precisely the same number of data points, in contrast with the FSVM where roughly 67% of the training consists in one class (others class), with the rest being equally distributed by high-mangrove and low-mangrove. Then when the binarisation process in FSVM is applied, the binary classifiers used to discriminate the target classes are developed with an imbalanced training set. The imbalance ratio in the training data set for the classes of interest is 1:9.

FSVM yielded lower sensitivity and specificity values in the classes of interest than SSVM. For high-mangrove, the difference in sensitivity between SSVM and FSVM is 2% while specificity differs 3.6%. For low-mangrove, on the other hand, sensitivity differs

13.0% while specificity differs 1.4%. With lower sensitivity, FSVM omits more positive cases than SSVM, which are precisely the pixels belonging to the classes of interest. On the other hand, with lower specificity FSVM commits more negative cases (elements of the class of non-interest) to the class of interest. These errors led to a decrease of the geometric mean of 2.8% and 7.8% in the discrimination of high-mangrove and low-mangrove respectively.

It is also important to notice that, although FSVM was tuned using the geometric mean, specific for each class of interest, that was not sufficient to overcome the effects of imbalance data. The determination of the parameters is an important factor in the sense that provides more sensibility to the learning algorithm about the boundaries of the classes of interest. However the issue introduced by the data imbalance remains, since the decision boundary will still be pushed way from the majority class.

The FOVO and FOVR experiments were conducted with the same training set as that of FSVM, but the data imbalance was mitigated with the use of data point weights. Overall accuracies were 97.3% and 96.7% for FOVO and FOVR, respectively, 6.3% and 5.7% higher than the FSVM. Sensitivity and specificity were higher in both classes of interest. The geometric mean yielded by FOVO and FOVR were 7.5% and 6.3% higher, respectively, for high-mangrove and 13.0% and 11.7% higher for low-mangrove. These results show how the use of weighted observations can be used to mitigate the effects of data imbalance in the training set for specific class mapping. In fact, the cost-sensitive approaches (FOVO and FOVR) yielded the highest geometric mean values in the discrimination of the classes of interest.

The main difference between SSVM and the cost-sensitive approaches is on the fine-tuning process, since both data sets are balanced, the first by design and the second by application of data point weights. The fine-tuning process in SSVM is generic; in other words, a single set of parameters was determined as the best set of parameters for the discrimination of each possible pair of classes since the utilised software implements OVO strategy to deal with multi-class problems. Thus, the fine-tuning process is effectively estimating the parameterisation yielding the maximum overall discrimination accuracy for the discrimination of the six land cover classes and not the best parameterisation for the particular discrimination of the classes of interest.

On the other hand, in FOVO and FOVR, the fine-tuning process is specific, that is it was applied to each particular pair of classes, and thus instead of determining the parameters that best fit the discrimination of all classes, each pair of classes had its own particular parameterisation. In contrast with SSVM, the training set and the fine-tuning process applied in FSVM is the same as those utilised in the two cost-sensitive approaches. In FSVM, although the fine-tuning process was specific to each particular binary classification problem, and not global as in SSVM, the imbalances present in the training set were not addressed.

To illustrate how the two best approaches compare regarding mapping the classes

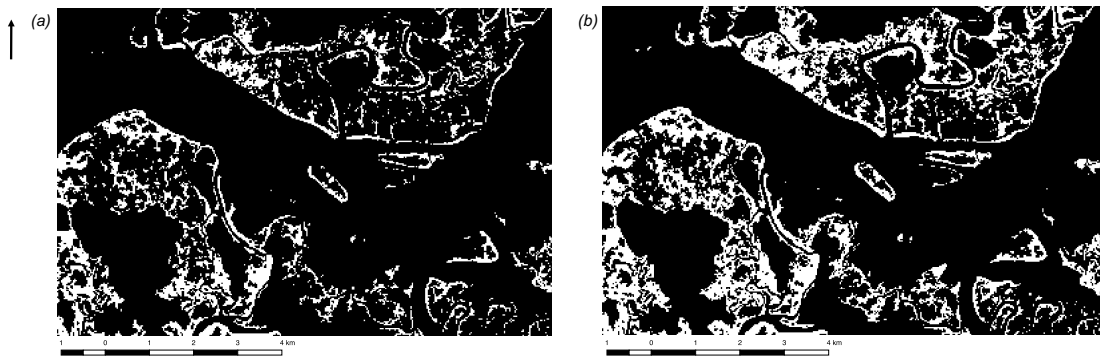



Figure 4.4: Binary map showing the areas of high-mangrove (white) and no-high-mangrove (black) classified by the SSVM frame (*a*) and the FOVO frame (*b*).

of interest, in figure 4.4, two binary classifications extracted from SSVM and FOVO are presented. For brevity sake and simplicity only the classifications of high-mangrove are presented, since similar observations can be done for low-mangrove. In general, the classifications show patches with similar geometrical structure, however the FOVO classification appears to be an expanded version of the SSVM classification. That is, FOVO classifier appears in general to add positive classifications around the positive classifications of SSVM. Although, data imbalance cannot be used to explain this effect, since SSVM was developed with a balanced data set, a similar effect to that observed in figure 4.3 may occur. That is, the decision boundary being located too close to the class of interest class. This may be caused by class composition of the training data set and in the way the learning algorithm parameters were fixed. Concretely, since the class of interest is only one small class in a larger group of six, a set of parameters inducing a classifier that correctly predicts the majority of the classes but neglecting the small class of interest, scores high in fine-tuning process. Such model ultimately defines a decision boundary closer to the class of interest, which may lead to a model that under-predicts this class. In other words, the classifier that is less sensitive to the class of interest. Note that the pixels added by FOVO are located near the interface between the class of interest and its negative. This suggests that these are pixels localised on edge of the class distribution, and thus are more likely to be misclassified by a classifier with low sensitivity to the class of interest, such SSVM. In other words, the classification errors committed by SSVM tend to be localised in such regions. This led the FOVO approach to predict roughly 7% more pixels of the classes of interest than the SSVM.

Table 4.6 summarises the statistical test results based on 95% confidence interval on the estimated difference in overall accuracy derived from different experiments. The 95% confidence interval for the estimated difference between the accuracies derived from FOVO and FOVR spanned from 0.3% to 0.9%, with centre at 0.6%, and lay within zone of indifference, indicating that FOVR classification was non-inferior to that of FOVO at 5% level of significance. The 95% confidence level for the difference

Table 4.6: 95% confidence interval (CI) on the estimated difference in overall accuracy obtained between the approaches. Results are presented in percentage and decision is done at 5% level of significance

| Method | Acc. diff. | 95% CI for diff. | Decision |
|---|---|---|---|
| SSVM vs. FSVM | 3.3 | [1.7 , 4.9] | Different |
| FOVO vs. SSVM | 3.0 | [1.5 , 4.5] | Different |
| FOVR vs. SSVM | 2.4 | [1.2 , 3.6] | Different |
| FOVO vs. FOVR | 0.6 | [0.3 , 0.9] | Equivalent |

between the classification accuracies yielded by the cost-sensitive approaches, FOVO and FOVR, spanned from 1.5% to 4.5% for FOVO and 1.2% to 3.6% for FOVR. The lower extremes of both intervals did not cross the zone of indifference, thus indicating that the classifications derived from FOVO and FOVR were significantly different from those derived from SSVM at 5% level of significance.

## 4.5 Summary and conclusions

Often users' interest is on a small sub-set of land cover classes present in the study area and not in a complete characterisation of the landscape. In these cases, conventional supervised classification techniques may not be appropriate for the derivation of information about these classes. Previous studies have shown that by combining the classes of no interest into a large single class and by decomposing the multi-class problem into a series of binary classification problems is sometimes a better approach than the conventional supervised classification method. However, this approach may suffer from data imbalance issues, since the classes of interest are usually a small component of the training set. In this article, cost-sensitive learning was applied to overcome data imbalances problems present in the training data. Experiments were conducted with Landsat 5 Thematic Mapper in Saloum, Senegal, where the classes of interest were high-mangrove and low-mangrove. The cost-sensitive learning outperformed the conventional multi-class approach and the focused approach in the discrimination of each class of interest. Classification accuracies derived from cost-sensitive approaches were significantly different from those derived from the standard multi-class and the focused approaches. Cost-sensitive approach also improved class specific discrimination. Indeed, for high-mangrove, the cost-sensitive learning approach yielded sensitivity and specificity geometric mean of 96.4% against 91.7% yielded by the multi-class approach and 88.9% yielded by the focused approach. And for low-mangrove, the cost-sensitive learning approach yielded a geometric mean of 95.6% against 90.4% yielded by the multi-class approach and 82.6% yielded by the focused approach. The cost-sensitive approaches as predicted roughly 7% more pixels of the classes of interest than the conventional supervised classification. Since interest was on more that one class, it is necessary to combine the outcomes of several binary classifiers. The

two most common approaches, the one-vs-one (OVO) and the one-vs-rest (OVR), were compared. The differences between the accuracies derived from OVO and OVR were not statistically significant. Indeed, although OVO show higher classification accuracy than OVR (97.3% against 96.7%), OVR accuracy was non-inferior to that of OVO at 5% significance level and using a 1% zone of indifference. From an operational point of view, the effort to apply OVO or OVR was the same, because the number of classes of interest was small. Since that is the case in most practical cases, the use of OVO or OVR may then be of little if any relevance. In summary, the study results suggest that the cost-sensitive learning is an effective solution to overcome data imbalances present in the training set and thus contribute to improve the classification accuracy of specific mapping of classes of interest.

Chapter 5

# Combined use one-class classifiers for specific class mapping: An experiment with forest classification

**Abstract** In land cover mapping with remotely sensed data the interest is often on a subset of classes present in the study area. The literature describes several supervised algorithms for one-class classification that are particularly attractive due to its reduced training effort. Nevertheless, there is no option if the interest is to identify a subset of the classes, instead of just one. In this paper it is proposed three combining methodologies to use one-class classifiers to map subsets of land cover classes. This is illustrated with the classification of deciduous and coniferous forest from Quickbird imagery. Three combination approaches were tested to take advantage of the one-class support vector machines non-exhaustive training set. These approaches were compared with conventional multi-class support vector machine developed with an exhaustive training set. Accuracy ranged from 80.00% to 87.33% showing any of the three combining approaches yield accurate results statistically similar to the multi-class classification. Thus the results suggest that an intelligent combination of single class classifiers can be used to achieve accurate results, statistically non-inferior to the standard multi-class classification, without the need of an exhaustive sample, saving resources that can be allocated to other steps of the data analysis process.

## 5.1 Introduction

Remote sensing data is an important source of land cover information and has been extensively used to map and monitor land cover classes over time to fulfil a variety of of scientific and managerial purposes [107]. Supervised classification, in particular,

has been frequently used to derive thematic maps depicting the land cover classes present in the study area from remotely sensed data. However, users are often not interested in all classes present in the study area but just on a small subset. This is evident in studies where major land cover transformations are object of study, such as deforestation [104] and urbanisation [24, 39], or where specific classes are of interest, such as abandoned agriculture [3], specific tree species [5, 54], invasive wetland species [87], and mangrove ecosystems [90, 140]. In such applications, the use of multi-class image classification methods can be inappropriate [55].

One problem in using multi-class approach for specific class mapping is that the training sample has to contain all classes present in the study area regardless of its importance for the analysis [9]. In other words, supervised multi-class classifiers are trained with an exhaustive training sample containing all classes present in the classification space. If a classifier is developed with a non-exhaustive training data set, the classifier may commit pixels of untrained classes into the set of classes in which the classifier was training [53]. This may originate classification error that are not identified in the accuracy assessment process [44]. For example, areas of untrained forest may consistently be committed into a particular crop or shrub. As result, the outputted map that overestimates the extension of that crop or shrub. Thus the analyst has to ensure all classes present in the study area are sampled in order to avoid such errors.

Another concern is that multi-class classifier is often developed to maximise classification accuracy over all land cover classes rather than focus on the specific classes of interest [88]. That is, the classification algorithm seeks to output a classifier where the overall classification accuracy, measured over all classes, is maximised. Since the classes of interest are typically only a small part of the training set, the algorithm may end up neglecting these small but important classes. Thus the analysis may not be optimal for the discrimination of these classes.

Therefore, when interest is on a subset of classes present in the study area, it may be preferable to follow an alternative approach to the conventional multi-class supervised classification method [108, 125]. Building a classifier capable of handling effectively only a subset of classes may be a better alternative for specific class mapping. There are essentially two general ways to implement such a classifier. One is to decompose the multi-class problem in a series of binary classification problems to separate the classes of interest from all the rest. Binary approaches tend to define simpler decision boundaries which reduce the competence areas of each classifier producing locally specialised models [82]. From these small binary problems, the original multi-class problem can be solved using combining strategies such as one-vs-one and one-vs-all [129]. Although studies have shown that binary decomposition performs well in most multi-class problems, it has nevertheless limitations such as being dependent of the combination method and being susceptible to data imbalance and sparse distributions

[82]. From the operational point of view, the binary classification approach still requires the sampling of land cover classes of no interest at training collection stage, since it is necessary to sample the classification space outside the classes of interest. And like the multi-class supervised method, if this space is ill sampled, that is some classes are omitted from the training or under-sampled, it is possible for a classifier to commit some of those areas into a class of interest [52], over-estimating the true extension of these important classes [55].

An alternative to the binary classification is the one-class classification. With this approach, each class of interest is treated independently and a one-class classifier may be developed for each. An attractive feature from the operational point of view is that the training set is composed exclusively by training data points of the classes of interest, which represents a significant reduction in the training sampling effort. Although these classifiers do not use all information available about the classification space, one-class classifiers display several desirable properties, since they are robust to many difficulties embedded in the data such as noise, imbalanced or complex distributions [79]. Previous studies [55, 101, 125] have shown that when interest is focused on just one class of interest, one-class classifiers are an efficient alternative since they require only training data for the target class yielding classifiers with classification accuracy non-inferior to that of standard multi-class methods. However, these studies have not explored the use of combining one-class classifiers to map a small set of classes of interest. This is relevant because in some applications, such as mangrove forest classification, interest is on a small subset of classes of interest and not just in one [31, 140]. In these cases, the direct use of one-class classifiers is not possible and it requires the combination of multiple classifier decisions.

In this paper, the aim and novelty is to show that the combination of multiple One-Class Support Vector Machines (OCSVM) can provide an accurate classification of the classes of interest non-inferior to that of the multi-class approach without the need to sample all classes present in the study area. In other words, if interest is only on a subset of the classes present in the study area, the combination of multiple OCSVM can be an efficient alternative to multi-class classification. To demonstrate this, a series of experiments are presented in three study areas located in Portugal mainland. Two land cover classes were defined as the classes of interest. These were classes of forest: deciduous forest and coniferous forest. Three combination strategies, inspired in the binary classifier combination schemes, were tested and their classification accuracy was compared with that derived from a conventional multi-class Support Vector Machines (SVM) classifier.

## 5.2 Background

### 5.2.1 One-class classification

The SVM is a popular supervised classification algorithm that has been successfully applied in many domains [108, 125]. In particular, in the classification of remotely sensed imagery, the study and application of SVM is extensive and well known [107]. In its origin, the SVM was developed to solve binary classification problems with linearly separable classes. However, the same principles can be applied to solve one-class problems, also know as novelty detection problems [128], that consist in detecting objects from a particular class. This class is often called target class or class of interest. These problems differ greatly from the standard supervised classification in the sense that the training set is composed exclusively by data points from the target class and thus there are no counterexamples to define the classification space outside the class of interest. One-class classification has been utilised in a variety of applications [126] and has great potential in remotely sensed data processing. There are two approaches to one-class classification based on SVM principles, the OCSVM [128] and the Support Vector Data Description (SVDD) [137]. In this paper, however, focus is in the use of OCSVM.

The basic idea behind the OCSVM is to determine a function that signals positive if the given data point belongs to the target class and negative otherwise. To achieve that the classification space origin is treated as the only available member of the non-target class. The problem is then solved by finding a hyperplane with maximum margin separation from the origin. Non-linear problems are dealt with a kernel function as in the binary SVM. The OCSVM optimisation problem is formulated as follows [128]:

$$min_{w,\xi,\rho} \frac{1}{2} \boldsymbol{w}^{\mathrm{T}} \boldsymbol{w} - \rho + \frac{1}{\nu m} \sum_{i} \xi_i \tag{5.1}$$

Subject to $\boldsymbol{w}^{\mathrm{T}} \boldsymbol{\phi}(\boldsymbol{x}_i) \geq \rho - \xi_i$ and $\xi_i \geq 0$. Here, $m$ is the number of training data points, $\boldsymbol{w}$ is the vector perpendicular to the hyperplane that defines the target class boundaries and $\rho$ is the distance to the origin. The function $\boldsymbol{\phi}$ is related with the kernel function [128]. The use of slack variables $\xi_i$ used in the OCSVM to allow the presence of class outliers, similar to binary SVM. The parameter $\nu$ ranges from 0 to 1 and controls the trade-off between the number of data points of the training set labelled as positive by the OCSVM decision function:

$$f(x) = sign(\boldsymbol{w}^{\mathrm{T}} \boldsymbol{\phi}(x) - \rho) \tag{5.2}$$

Applying the Karush-Kuhn-Tucker (KKT) [30] conditions to the original OCSVM problem, this can be rewritten as depending of the Lagrange multipliers $\alpha$:

$$min_{\boldsymbol{\alpha}} \sum_{i,j} \alpha_i \alpha_j K(\boldsymbol{x}_i, \boldsymbol{x}_j) \tag{5.3}$$

Subject to $\sum_i \alpha_i = 1$ and $0 \leq \alpha_i \leq \frac{1}{\nu m}$ for all training data points, where $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{\phi}^{\mathrm{T}}(\boldsymbol{x}_i)\boldsymbol{\phi}(\boldsymbol{x}_j)$ is the kernel matrix defined by the kernel function $\boldsymbol{\phi}$. From this, the decision function can be rewritten depending only on the non-null Lagrange multipliers and on the kernel matrix values:

$$f(x) = sign(\sum_i \alpha_i K(\boldsymbol{x}_i, \boldsymbol{x}) - \rho) \tag{5.4}$$

The data points with non-null Lagrange multipliers are effectively the support vectors of the one-class classifier. The classification rule is based on the signal of the decision value, positive if the data point is located inside the target class, negative otherwise. The absolute value of the decision value is directly related with the distance of the data point to the separating hyperplane in the transformed classification space [128].

### 5.2.2 Fine tuning one-class classifiers

Like SVM, OCSVM algorithm depends on free parameters that need to be set to develop the classifier. These free-parameters consist in kernel parameters, for example the radial factor of the radial-basis kernel function and the degree of the polynomial kernel, and regularisation parameters. In the case of OCSVM this parameter is $\nu$, ranging from 0 to 1, that defines the upper bound of the fraction of training data points regarded as outliers and and the lower bound of the fraction of training data points regarded as support vectors [128]. The determination of these free-parameters is important. In binary and multi-class classification, the determination of the free-parameters is often done by grid-search cross-validation [9, 30]. However, the training set in this study does not contain data points outside the class of interest, and thus it is not possible to assess the specificity of the classifier (the proportion of data points outside the class of interest that were correctly classified) in the cross-validation process [94, 109]. Thus only sensitivity can be assessed. Using only the sensitivity to parameterise a classification algorithm may result in a classifier with high sensitivity and low specificity, overestimating the extension of the classes of interest.

$$\hat{\theta} = argmax_\theta \frac{\frac{1}{n}\sum_k I(f_\theta(\boldsymbol{x}_k) = +1)}{N_{sv}(f_\theta)} \tag{5.5}$$

To minimise the effects of this limitation, the cross-validation process can be carried out using the ratio between the sensitivity and the number of support vectors as metric [7, 109] (equation 6.7), where $n$ is number of testing data points, $I$ is the characteristic function, $f_\theta$ is the decision function (equation 5.4) parametrised with $\theta$ and $N_{sv}$ is the number of support vectors in $f_\theta$. This ratio enforces high sensibility while limiting model complexity (the number of support vectors) which usually indicates good model generalisation ability [7].

### 5.2.3 Combining decisions

The combination of one-class classifiers is not as well studied as the combination of binary classifiers [80]. Indeed, binary classification combination has gained significant of the machine learning community since it is at the base of many multi-class classifiers, such as SVM, and has been proven to perform well in most multi-class problems [57]. Although less study, one-class classifiers combination has been explored and shown its usefulness [80]. Most one-class classifiers combination strategies are based on strategies applied to binary classifiers, namely One-vs-One (OVO), One-vs-All (OVA) and Decision Directed Acyclic Graph (DDAG).

In OVO the multi-class problem is decomposed in a pairwise way; that is, all possible pairs of classes are enumerated and a binary classifier is developed for each one. Thus if there are N classes, the OVO strategy implies $\frac{1}{2}N(N-1)$ binary classifiers [129]. The final classification is then performed by majority voting [80]. In the OVA strategy, one binary classifier is developed for each class where all other classes are agglomerated into a single large class. Although this approach implies less classifiers it may be susceptible to data imbalances issues [9]. To combine the different outcomes of each classifier, and to avoid non-labelled data points, the assigned label is that of the class with maximum decision value [129]. The DDAG constructs a rooted binary acyclic graph where each node is a classifier (not necessary a binary one) that redirects the decision. The final classification, like in a decision tree, is found once decision reaches a terminal node [95].

## 5.3 Data and methods

### 5.3.1 Study sites

The study areas are located in continental Portugal (Fig. 5.1). Three locations were selected based on image availability, landscape variability and how well represent the different land cover makeups present in continental Portugal.

Study site 1 is located near Salvaterra-de-magos and is composed mostly by agriculture (rainfed agriculture, irrigated agriculture and rice fields) with patches of forestry. The classes of interest here consist in almost 29% of the study site (table 5.1). Study site 2 is located in Lourinhã, near the seashore, and consists in pastures, bare soil, rainfed agriculture and shrubs. Here the classes of interest compose roughly a quarter of the site (table 5.1), with deciduous occupying 9.98% of the site and coniferous 15.54%. Study site 3 is located at North, in the region of Minho. This area is heavily composed by small patches to agriculture, such as vineyards and fruit trees, shrubs and forestry. The classes of interest compose almost two-thirds of the site (table 5.1), with deciduous forest composing 34.45% and coniferous forest composing 33.16%. Each study site contains also small villages and roads.

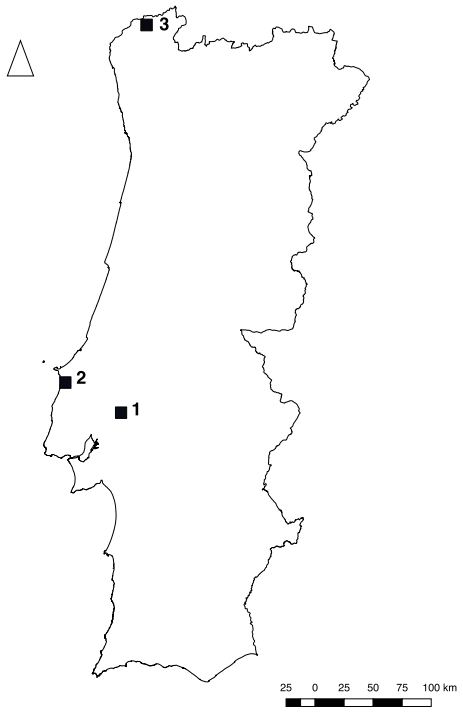



Figure 5.1: The three study areas located in continental Portugal.

Table 5.1: Composition of the each study site. Results are presented in percentage. Others represent the class of land cover types that are neither deciduous nor coniferous forest.

| Classes | Site 1 | Site 2 | Site 3 |
|---|---|---|---|
| Deciduous | 16.18 | 9.98 | 34.45 |
| Coniferous | 12.69 | 15.54 | 33.16 |
| Others | 71.13 | 74.47 | 32.38 |

### 5.3.2 Data

Remotely sensed data was acquired in September 2004 by Quickbird with 2.4 m geometric resolution for each study area. All four bands have been used, and since only one image per study area was utilised for analysis and the atmosphere may be considered to be homogeneous within each scene, atmospheric correction was not necessary [132]. The Normalised Difference Vegetation Index (NDVI) was computed as well the mean and standard deviation filters using a 3x3 moving window for each discriminating variable and then rescaled to $[0, 1]$ range [94]. Thus the data set is composed by 15 dimensions ranging in $[0, 1]$.

Training data was manually collected by an image analyst with the support of Google Earth and aerial imagery. For the purpose of this study, the multi-class SVM was utilised as benchmark and thus training data for all classes present in the study sites are necessary. Since the study sites are different, not just in location but also

Table 5.2: Parameterisation of the multi-class SVM for each study site.

| Site | $\gamma$ | $C$ |
|---|---|---|
| 1 | 0.03125 | 32 |
| 2 | 0.03125 | 1024 |
| 3 | 0.125 | 32 |

in land cover makeup, three training sets were collected independently. In each one of them, 100 pixels per land cover was collected. In site 1 and site 2, the training set consisted in a total of 800 pixels, composed by water, roof-tops, roads, bare-soil, irrigated agriculture, rainfed agriculture and forestry (deciduous and coniferous). In site 3, on the other hand, 1000 pixels were collected for training, covering burnt areas, roof-tops, roads, shadows, bare-soil, shrubs, deciduous and coniferous forest, and two types of agriculture.

### 5.3.3 Experiments

Four experiments were conducted to demonstrate that the intelligent use of multiple OCSVM can be used to map a set of specific land cover classes without the need to collect training samples for all classes present in the study site. The first consisted in the benchmark and the other three demonstrate the goal of this research paper.

The first experiment consisted in a conventional multi-class SVM classification. Here the classifier was developed with a training set as defined in section 5.3.2 and with radial-basis kernel. The SVM kernel free-parameters $C$ and $\gamma$ were determined using a 10-fold cross-validation grid-search as described in [20]. Table 5.2 summarises the parameterisations of the multi-class SVM for each study site. These experiments were performed with LibSVM software version 3.21 [20].

The next three experiments show three different approaches to the combination of OCSVM. Although not all relay exclusively on one-class classifiers, all approaches require only training samples from the classes of interest. In some cases, a binary classifier can be developed to discriminate between two classes of interest, which does not require additional information besides the training data of these classes.

In the second experiment, in figure 5.2 (*a*), a OCSVM is developed for each class of interest. Here the decision values obtained by each classifier where used instead of the output labels. The reason for this is that the label in these cases is the signal of the decision value, as defined in equation 5.4, and provides no additional information about the classification. This can be problematic when the two classifiers output positive decision values, since then there is no additional information to break the tie. Note that when all classifiers output a negative decision value, they are effectively agreeing that the pixel being classifier does not belong to any class of interest and thus there is tie to be solved. To combine the decision values of the two classifiers the following rule was applied: if both values are negative, then the pixel was classified as

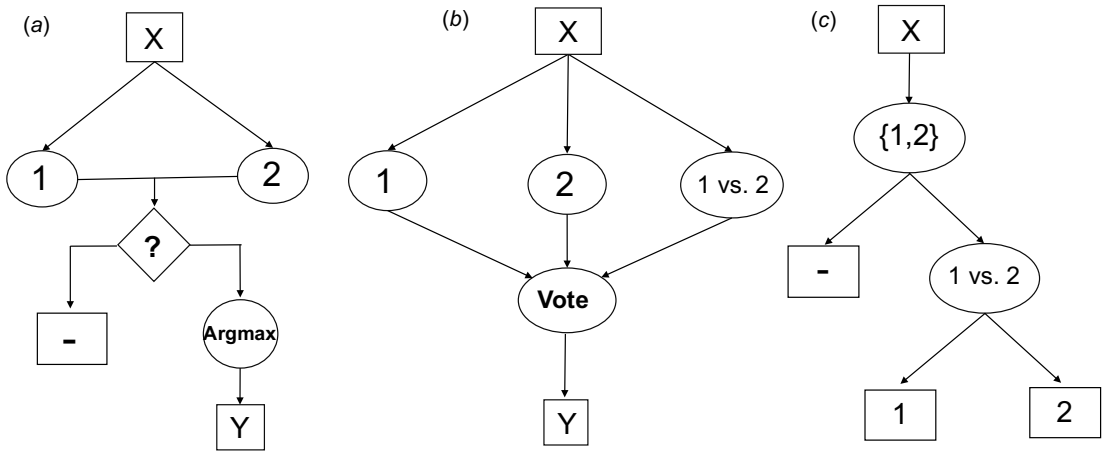



Figure 5.2: (*a*) second experiment based on OVA strategy; (*b*) third experiment based on OVO strategy; (*c*) fourth experiment based on DDAG. X represents a generic pixel and Y the output of the process. 1 represents the first class (deciduous) of interest and 2 represents the second class of interest (coniferous) and 1 vs 2 represents a binary classifier discriminating class of interest 1 from class of interest 2. The minus signal represents the set of pixels that are neither class 1 nor class 2.

Table 5.3: Parameterisation of the OCSVM for each study site.

| | Deciduous | | Coniferous | |
|---|---|---|---|---|
| Site | $\gamma$ | $\nu$ | $\gamma$ | $\nu$ |
| 1 | 0.250 | 0.25 | 0.125 | 0.25 |
| 2 | 0.250 | 0.50 | 1.000 | 0.50 |
| 3 | 0.125 | 0.50 | 0.500 | 0.50 |

outside of the classes of interest; otherwise the pixel was classified as belonging to the class with maximum decision value. Since the decision values may range in different intervals, the positive decision values were normalised using the sigmoid function [129]. This approach mimics that of the OVA strategy, where one binary classifier is developed for each class and the decision is based on the highest decision value [129]. Note that the method can easily adapted if the user is interested in more than two classes of interest.

The third approach mimics the OVO strategy, figure 5.2 (*b*). Here one OCSVM is developed for each class and a binary SVM is developed to discriminate one class of interest from the other. Effectively, the first classifier, a OCSVM, aims to identify deciduous; the second, also a OCSVM, aims to identify coniferous; and the third, a binary SVM, aims to discriminate deciduous from coniferous which is a standard binary classification problem. The classification was then performed by majority voting. Note that the binary classifier here is developed using only data from the classes of interest; when the two OCSVM classifiers tie, the binary classifier is then used to break the tie. A pixel then is neither deciduous nor coniferous only when the OCSVMs agree. The

Table 5.4: Parameterisation of the SVM for methods 2 (deciduous vs. coniferous) and OCSVM for method 3 (interest vs. no-interest).

| | Deciduous vs. Coniferous | | Interest vs. No-interest | |
|---|---|---|---|---|
| Site | $\gamma$ | $C$ | $\gamma$ | $\nu$ |
| 1 | 4.8828e-4 | 128 | 9.7656e-4 | 0.01 |
| 2 | 1.9531e-3 | 512 | 2.4414e-4 | 0.01 |
| 3 | 3.9063e-3 | 64 | 1.9531e-3 | 0.01 |

method can be adapted for cases with more than two classes of interest. Indeed, for $N$ classes of interest, the method develops $\frac{1}{2}N(N+1)$ classifies: $N$ single classifiers and $\frac{1}{2}N(N-1)$ binary classifiers. The parameters of the OCSVM classifiers were the same as in the first approach; the binary SVM parameters were determined with 10-fold cross-validation trails. Table 5.4 summarises the parameters utilised for each site.

The fourth experiment is based on DDAG where at root node is a OCSVM developed to identify belonging to the classes of interest, figure 5.2 (*c*). This node effectively divides the classification space into interest and non-interest pixels. A second node is built under the interest pixels set to identify deciduous from coniferous pixels, where a standard binary classifier is utilised. Note that if the user is interested in more than two classes of interest, this second node is effectively a multi-class problem where any standard multi-classification approach, such as OVO or OVA for example, can be used. Indeed the DDAG approach is similar to the OVO but dissimilar enough to induce different classifications. In the first step of DDAG, a single OCSVM is used to identify pixels of interest from pixels of no interest using all the data of the classes of interest, whereas in the OVO multiple OCSVM classifiers are used that contribute to the final decision. Table 5.4 summarises the parameterisations of the OCSVM for each study site. The parameterisation employed in the classifier in the second node was the same as in table 5.4.

### 5.3.4 Accuracy assessment and comparison

Classification accuracy was estimated using three independent testing set of 300 random pixels for each study area, with 100 for each class of interest, deciduous and coniferous, and 100 for the remain. The number of testing samples was determined by the trade-off between the operational implementation effort and the expected precision of the estimated accuracy metrics. Note that in this paper interest is the classification accuracy of maps composed by only three classes, the two classes of interest and the remain without class specification. In other words, classification errors between classes of no interest are disregarded. An image analyst visually classified each pixel in the same year as the image acquisition with support of 50 cm aerial imagery. The accuracy of each classification was expressed in terms of the proportion of correctly classified

testing data points. Since a single testing set was used for each test site, the statistical significance of the difference in overall accuracy between different classification approaches will be assessed using the McNemar test [49].

The McNemar test is based on a binary contingency table in which pixels are classified as correctly or incorrectly allocated by the two classifiers under comparison. The main diagonal of this table shows the number of pixels on which both classifiers were correct and on which both classifiers were incorrect. The McNemar test however focus on proportion of pixels where one classifier was correct but the other was incorrect. The analysis will be based upon the evaluation of the $100(1-\alpha)\%$ confidence interval, where $\alpha$ is the level of significance, for the difference between two accuracy values expressed as proportions (say $p_1$ and $p_2$) expressed as [42]:

$$p_2 - p_1 \pm z_\alpha s \tag{5.6}$$

Where the term $s$ is the standard error derived of the difference between the proportions, which can be determined by [42, 50]:

$$s = \sqrt{\frac{p_{01} + p_{10} - (p_{01} - p_{10})^2}{n}} \tag{5.7}$$

where $p_{10}$ the proportion of testing pixels where the first classifier was correct and the second was incorrect and $p_{01}$ the proportion of testing pixels, where the first classifier was incorrect and the second was correct. In this way, the statistical assessment of the differences was conducted to determine if these were significantly different or not [49].

Additionally, to the classification accuracy, classification sensitivity and specificity were also calculated for each target class. Sensitivity defined as the proportion data points belonging to the class of interest correctly classified and specificity as the proportion of data points outside the class of interest correctly classified [145]. In this sense, sensitivity and specificity complement each other, since each metric evaluates the quality of the classification from the two sides of the classification space, inside and outside the classes of interest. The sensitivity and specificity analysis can also be useful identify asymmetric classifiers, that is classifiers high number of false positives and low number of false negatives, or vice-versa. In these cases, sensitivity and specificity yield asymmetric results with one of them being smaller than the other.

Since the goal of the analysis is to show that combination of one-class classifiers can be as accurate as multi-class classification when interest is only on a subset of the classes present in the study area but requiring less training, the statistical analysis aims to show that the classification derived from the combination of OCSVM is non-inferior to that of multi-class SVM. It was assumed for the purposed of this paper that any decline in accuracy smaller than 2% was irrelevant and this value was used to define the region of indifference [114].

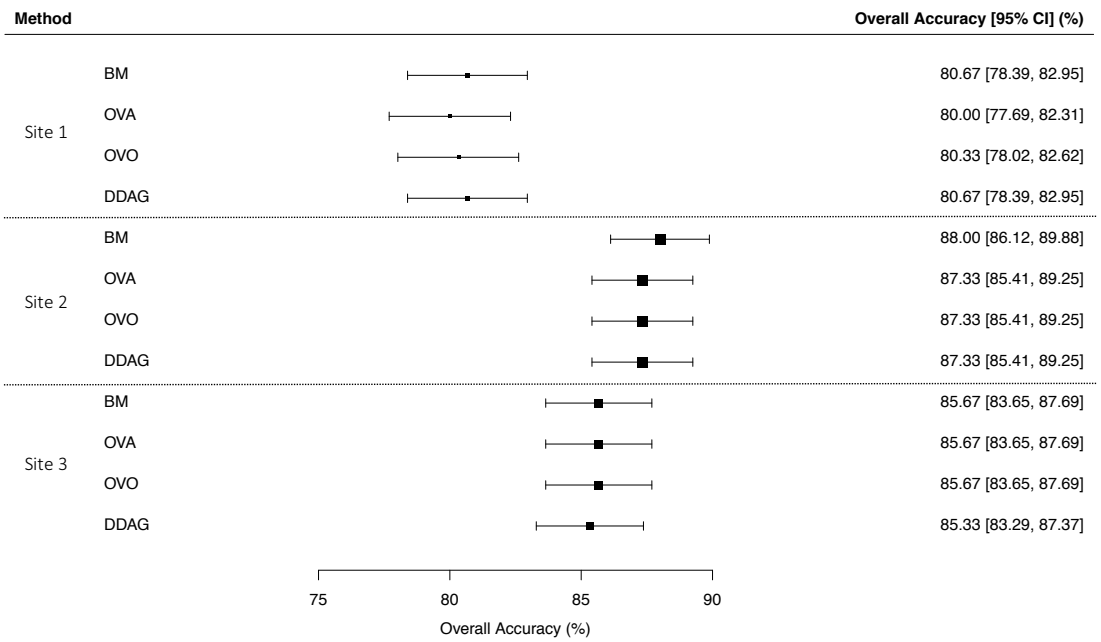



Figure 5.3: The overall accuracy and the respective 95% confidence interval of each method in each study site.

## 5.4 Results and discussion

For the purpose of this discussion, the benchmark method is referenced as BM, the first method of combination (figure 5.2 - a) is referenced as OVA, the second method (figure 5.2 - b) is referenced OVO and the third method (figure 5.2 - c) is referenced as DDAG.

In figure 5.3 is summarised the classifications accuracies of each method in each study site and their respective 95% confidence interval. It is possible to observe that the methods show similar performances within each study site, and that accuracies in sites 2 and 3 are very similar and slightly higher than in site 1. In study site 1, Benchmark (BM) and DDAG achieved the larger classification accuracy with 80.67% followed by OVO with 80.33% and by OVA with 80.00%. In site 2, BM achieved the larger classification accuracy with 88.00%, followed by the remain methods with 87.33%. And in site 3, BM and OVA achieved the larger classification accuracy with 85.67%, followed by OVO and DDAG with 85.67% and 85.33%, respectively. Although small differences in accuracy between classifications are present for the same site, the larger differences exist when comparing different sites. For example, BM yield 80.67% in site 1, 88.00% in site 2 and 85.67% in site 3.

In figure 5.4 is summarised the difference in classification accuracies results based on 95% confidence interval on the estimated difference in classification accuracy from the benchmark. Per site, the classification methods present small differences to the benchmark that are contained in the region on interest. All difference intervals in all study sites are within the pre-defined region of indifference. That indicates that all

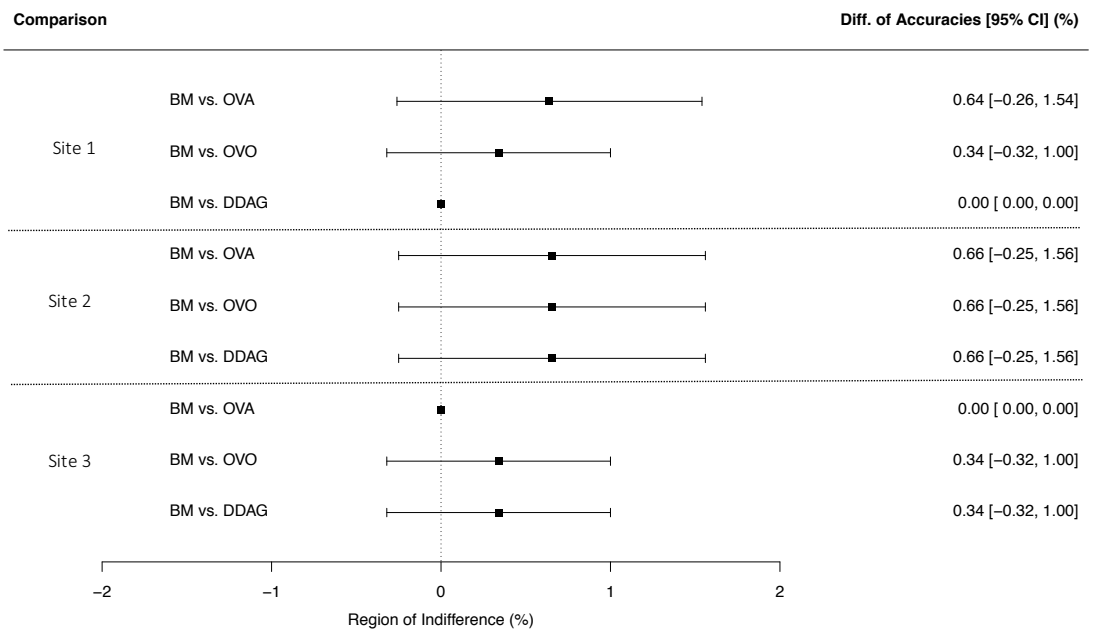



Figure 5.4: Difference test results based on 95% confidence interval on the estimated difference in classification accuracy from the benchmark. Note that the region of interest ranges from -2% to +2%. These are the maximum allowed differences between the tested methods and the benchmark. Intervals contained in the region of interest indicates, at 0.025 level of significance, that the proposed methods are non-inferior to the benchmark.

methods are non-inferior to the benchmark at 2.5% level of significance. Note however that the analysis of the results of DDAG in site 1 and of OVA in site 3 is trivial since the difference is null. This does not mean, however, the maps are equal. Indeed, these two approaches induce different errors as highlighted by the results of the sensitivity and specificity in figure 5.5.

In figure 5.5 is summarised the sensitivity and specificity of each method in each study site for both classes of interest. In the left, frame (*a*), deciduous class and in the right, frame (*b*), coniferous class. The dashed line represent the 1:1 straight line. That is, the points in this line hold the condition that the sensitivity is equal to the specificity. Thus a point above the 1:1 line indicates a method with more specificity accuracy than sensitivity (number of false positive is larger than the number of false negative), while a point below the 1:1 line indicates a method with more sensitivity accuracy than specificity (the number of false negatives is larger than the number of false positive)

In site 1, the sensitivity analysis for the deciduous class show BM and DDAG with 81.00% followed by OVO with 80.00% and OVA with 79.00%. The specificity analysis reveals that any of the methods were equally accurate with 80.50%. For the coniferous class, sensitivity analysis show that any of the methods performed equally with 79.00% and the specificity analysis show BM and DDAG with the higher accuracy,

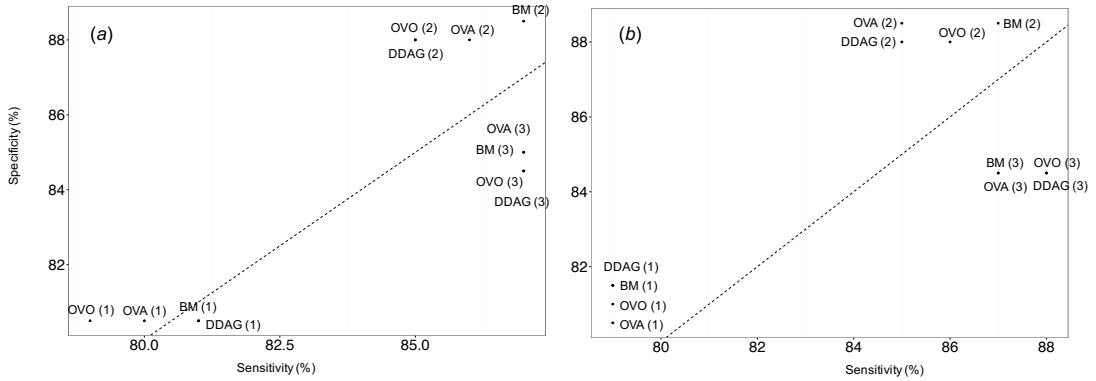



Figure 5.5: Sensitivity and specificity of each method in each study site (in parenthesis). Frame (*a*) represents the sensitivity versus specificity of all methods regarding the classification of deciduous class. Frame (*b*) represents the same thing but regarding the classification of coniferous class. The dashed line represents the 1:1 straight line. Thus the more a method is close to the top-right corner (more sensitivity and more specificity), the better it is classifying the class of interest.

81.50%, followed by OVO with 81.00% and OVA with 80.50%. In general, the classification accuracies in study site 1 were smaller than those of the site 2 and 3. A possible reason for this is the site land cover and their spectral confusion. Indeed, this study area is composed mostly by agriculture and small elements of natural vegetation, such as shrubs and grassland, occupying roughly 73%. These natural elements are intermingled with the classes of interest and present high spectral similarity to the target classes, deciduous and coniferous forest. This may lead to an increase of overlapping regions in the classification space and thus ambiguity in the classification process.

In site 2, the specificity all methods show equal performances, with 88.00%, differencing only 0.5% from the benchmark. For the coniferous forest, the analysis of the specificity shows BM and OVA with the highest results, 88.50%, followed OVO and DDAG only with 88.00%. For the coniferous forest the most sensitive method was the benchmark, with 87.00%, with the remained methods with sensitivity values differencing no more than 2%. The specificity analysis show BM and OVA with the highest values (88.50%) with the remain methods only 0.50% smaller. Thus, although small differences were present in the sensitivity and specificity, the overall show the methods performing equally well with highest overall accuracies (> 87.00%). It is important also to notice that in this site sensitivity is consistently smaller than specificity. This suggests that the extension of the classes of interest is being underestimated. This can be explained by the landscape of site 2, where the deciduous and the coniferous forest are present only in small and concentrated pockets, for forestry management purposes, composing roughly 25% of the study site and everything else, mostly rainfed agriculture grassland and urban, being spectrally very different. These two factors may lead to better classifications, since spectral dissimilarity reduce the overlapping regions between classes, and as consequence the ambiguity in the classification process. The

confinement of the classes of interest to very specific and evident locations, on the other hand, may lead the image analysts to collect training pixels with low variability. This misrepresentation of the classes of interest may lead the definition of a shrunken decision boundary increasing the number of false negatives.

In site 3, for the deciduous forest, any of the methods is equally sensitive yielding 87.00%. The specificity the methods differ, with BM and OVA achieving 85.00% and OVO and DDAG achieving 0.5% less. For the coniferous class, again BM and OVA achieve the larger sensitivity values with 88.00% and OVO and DDAG 87.00%. At the specificity level all methods perform equally, with 88.50%. Thus in general the classification accuracies were high (> 85.00%), although not as high as in site 2. It is important to notice however that differently from site 2 here the sensitivity is consistently larger than specificity, which suggests that the extension of the classes of interest are being overestimated by the classifiers. Here also the landscape composition can be used to explain this difference. Indeed, in this study are each class of interest represents roughly 33% and thus almost two thirds of the site is occupied by one of these classes. The remain third is composed by small patches of agriculture, such as vineyards, fruit trees and olive plantations, and natural vegetation such as grassland and shrubs in many classes intermingled with forestry (deciduous and coniferous). This leads to overlapping regions in the classification space but also hinders the image analysts training collection, leading them to mislabelled data points.

On the combining classifications, OVO and DDAG show equal classification accuracies in site 3 and also the lowest. OVA show the lowest classification accuracy in site 1 and the highest in site 3 matching that of the benchmark. And in site 2, all combining approaches show equal classification accuracy. In any of the study sites, the BM has achieved the larger classification accuracies, which is expected and supported by literature. Indeed, previous studies have shown that when data points outside the classes of interest are present in the training set, binary classifiers tend to outperform single-class classifiers [80]. The reason for this is that information about the distribution of the classification space outside the class of interest helps the learning algorithm to define more accurate decision boundaries [137]. However, from the operational point of view, definition of training pixels for the classes of no-interest may require too much effort [142]. Indeed, in a site composed by, say, ten land cover classes, from which only two are relevant for the purpose of the analysis, a large portion of the time and analysts is allocated to the data collection of classes of no importance for the purpose of the analysis, and the resulted classifier may be sub-optimal for the discrimination of those classes of interest [125]. The results of this study, however, have shown that an intelligent combination of single class classifiers can be used to achieve accurate results, statistically non-inferior to those of the standard multi-class classification, without the need to sample all classes present in the study site, saving resources that can be allocated to other steps of the data analysis process.

## 5.5 Conclusion

This work is focused on the classification problem of a subset of classes present in the study site. This is what often users intend, since in many research focus is on sub-set of classes and not on all classes that make up the study site. Conventional multi-class classification can be used to derive a map depicting the classes of interest, but considerable effort is allocated in the definition of an exhaustive training set containing all classes present in the study area. Additionally the classification process may be sub-optimal for the discrimination of the particular classes of interest. Alternatively, with one-class classification, only training data of the classes of interest is utilised which, from the operational point of view, represents a considerable reduction in sampling time and effort, and the analysis is focused on the classification of particular classes of interest. In this paper three one-class classifiers combination approaches were used to efficiently map a set of classes interest present in the study site. This is illustrated with the classification of deciduous and coniferous forest in three different study sites, from Quickbird imagery, using one-class support vector machines. These approaches were compared with conventional multi-class support vector machine, which was developed using a training set containing all classes present in the study area. Classification accuracy ranged from 80.00% to 87.33%, showing that any of the three combining approaches yield accurate results similar to the multi-class classification method, used as benchmark, suggesting them to be non-inferior to the benchmark at 2.5% level of significance. From this study, thus, results that an intelligent combination of one-class classifiers can be used to achieve accurate results, statistically non-inferior to those of the standard multi-class classification, without the need to sample all classes present in the study site, saving resources that can be allocated to other steps of the remotely sensed data analysis process.

Chapter 6

# Specific land cover class mapping by semi-supervised weighted support vector machines

**Abstract** In many remote sensing projects on land cover mapping the interest is often on a subset of classes presented in the study area. Conventional multi-class classification may lead to a considerable training effort and to the underestimation of the classes of interest. On the other hand, one-class classifiers require much less training but may overestimate the real extension of the class of interest. This paper illustrates the combined use of cost-sensitive and semi-supervised learning to overcome these difficulties. This method utilises manually collected set of pixels of the class of interest and a random sample of pixels, keeping the training effort low. Each data point is then weighted according to its distance to its near positive data point to inform the learning algorithm. The proposed approach was compared with a conventional multi-class classifier, a one-class classifier and a semi-supervised classifier in the discrimination of high-mangrove in Saloum estuary, Senegal, from Landsat imagery. The derived classification accuracies were high, 93.90% for the multi-class supervised classifier, 90.75% for the semi-supervised classifier, 88.75% for the one-class classifier and 93.75% for the proposed method. The results show that accuracy achieved with the proposed method is statistically non-inferior to that achieved with standard binary classification requiring however much less training effort.

## 6.1 Introduction

Remote sensing is today an integral part of many research activities related with Earth monitoring [26]. And in particular supervised classification of remotely sensed data

has become a fundamental tool for the derivation of land cover maps [107]. Indeed, users are often not interested in a complete characterisation of the landscape but rather on a subset of the classes that occur in the region to be mapped. For example, users may be focused on mapping urban classes [24, 39], abandoned agriculture [3], specific tree species [5, 59], invasive wetland species [87], or mangrove ecosystems [90, 140]. Fundamentally, depending on the application, the accurate discrimination of some classes is more important than the discrimination of others [88]. In this paper, it is assumed that interest is in a single land cover class but the discussion can be adapted if focus is on a subset of the classes composing the study area.

When interest is focused on a single class the use of conventional supervised classification process may be inappropriate [55]. Indeed, this approach assumes that the set of classes has been exhaustively defined [44, 53]. Thus, the correct application of this analysis require that all classes that occur in the study area be included in the training set [45, 103]. Therefore, when mapping a region for a user interested in urban land cover it will be necessary to collect training data points not only on the urban classes of interest but also on secondary classes with no interest to the user, such as crops, forest, water if these are present in the area of study. If these classes are not included in the training data set, the classifier will commit pixels of untrained classes into trained classes. For example, if the land cover class forest was not incorporated in the training data set, pixels of forest my be systematically classified as a type of shrub or crop, which greatly overestimates the real extent of those shrubs and crops classes. The user, therefore, must seek to ensure that all classes occurring in the region of interest are sampled to fulfil this requirement. In other words, the users have to allocate time and effort in training classes that are of no interest for their goals.

In addition, conventional supervised classification algorithms often are not optimised for the discrimination of a particular class [55, 88]. The classification algorithm seeks a classifier where the overall classification accuracy, measured over all classes, is maximum [16]. The class of interest, that is typically just one and often a small part of the set of classes, may be neglected in the process, and thus the resulted model may not be optimised for the discrimination of that particular class that may underestimate the class of interest [6]. In other words, the classifier may accurately discriminate secondary classes to the detriment of the class of interest [45, 55]. Hence, in both training and allocation stages, conventional supervised classification approaches are not focused on the class of interest. This take users wastefully directed training effort on classes of no interest and leads to an analysis that may not be optimal in terms of the discrimination of the important class. Therefore, when interest is on a class of interest, it may be preferable to follow an alternative approach to the conventional multi-class supervised classification method [125].

Literature shows that there are essentially two alternatives to the standard multi-class supervised approach: the binarisation strategy and one-class learning algorithms [57, 79, 136]. With binarisation strategy, users decompose the multi-class problem in

a series of small binary classification problems where one seeks to separate the classes of interest from all irrelevant classes [13, 41, 57, 82]. As binary classification is well-studied, binary decomposition of multi-class classification problems have attracted significant attention in machine learning research and has been shown to perform well in most multi-class problems [82]. Indeed, binary decomposition has been widely used to develop multi-class Support Vector Machines (SVM) showing better generalisation ability than other multi-class approaches [65]. The possibility to parallelise the training and testing of the component binary classifiers is also a big advantage in favour of binarisation apart of their good performance [57]. In particular, binarisation can be achieved by combining all land cover classes of no interest into a large nominal class, called for example "others" [47]. In this way the class of interest can be regarded as the positive class and all others as the negative class in the binary classification scenario. Previous studies [13, 47, 59, 90] have shown it to be possible to decompose the multi-class classification problem in a series of small binary classification problems and achieve results that are more suitable for the particular users' requests, namely the improvement of the discrimination of particular land cover classes of interest. Although specific class mapping can potentially be a better approach compared to the multi-class supervised classification, it has some particular difficulties, namely data imbalance in the training set [6, 9]. This is because often the classes of interest are only on a small component of the study area [88]. In fact, applying directly a binary decomposition to the classification problem may result in a highly unproportional allocation of training points to the negative class, leading to imbalance in the training data set [9]. In addition, the binarisation approach also requires the users to collect training on classes of no interest, similarly to what happens in the multi-class approach.

With the one-class classifier these problems are not present, since the user has only to collect training from the class of interest. However that is also its major limitation, since only data about one class is available and thus only one side of the discriminative boundary can be determined [136]. It can then be difficult to determine how tightly the boundary should fit in all directions around the class of interest in feature space. To overcome this difficulty some one-class classifiers (e.g. support vector data description) assume that the non-interest classes have a particular distribution around the class of interest. When the true distribution deviates from the assumption, the method may underperform [81]. Indeed, since the classifier is not able to bind the class distribution, the classifier may lead to over-expanded decision boundaries [136].That deviation however can only be assessed with training points outside of the class of interest [136]. Literature also shows that when information about the classification space outside the class of interest is available, binary classifiers tend to develop more accurate classifiers then the one-class approach [8, 79].

In the specific class mapping context, since the class of interest in typically only a small component of the study site [88], the number of negative pixels are much larger compared to the number of positives pixels, that is the pixels of interest. That is, there

is typically an over abundance of negative pixels in such that the probability of an arbitrary unlabelled pixel to be negative is much higher than the probability of being positive. In this context, the use of randomly selected data points can be an option to improve specific class mapping. This approach is typically known as semi-supervised learning.

Semi-supervised learning, also known as positive and unlabelled learning, is meant the use of unlabelled data points to inform the learning algorithm [21]. Here, semi-supervised learning can be possible approach to the classification process, since *a priori* there is bias toward the negative class [134].

Previous works have used semi-supervised learning approaches to map land cover classes, for example [6, 109]. In these studies, the Biased Support Vector Machines (BSVM) algorithm has been utilised with success to map classes like urban and tree tops from aerial imagery [94], and to classify single tropical species [6]. With BSVM the unlabelled set is regarded the negative class and the cost associated to the positive class and the negative class are asymmetrical, so that an error occurring in the positive class is more costlier than an error on the negative (unlabelled) class [96]. However, it is not clear how to set up the weights and the trial-and-error approach usually takes long computation time [34].

In this paper a similar approach is presented, however a different cost-sensitive approach is employed. A set of unlabelled data points, randomly defined, is used as an approximation to the negative class and only the positive class (the class of interest) is manually sampled. However, the costs associated to each class are not asymmetrically defined, but rather the individual data points are weighted differently according to its similitude to a positive data point. This similitude metric is function of the euclidian distance to its nearest positive data point. The heuristics followed here is that the closer negative data point is to a known positive data point, the higher the likelihood of that negative point to be mislabelled. Note that the positive data points are manually defined by an human analyst and thus considered certain and correct. The weight distribution in the negative class is then used by a cost-sensitive learning algorithm to develop a binary classifier. In other words, the proposed method aims to develop a binary classifier to classify a particular class of interest with the same sampling effort that is required by the one-class classifiers but providing the same discriminating information of a binary classification. Here the proposed method was compared with three different alternative approaches: the conventional multi-class supervised approach, here a support vector machine classifier; a single-class classifier, the One-Class Support Vector Machines (OCSVM); and a semi-supervised method, the BSVM.

## 6.2 Background

The SVM algorithm is a popular supervised classification algorithm that has been successfully applied in many domains [125]. In particular, in the classification of remotely sensed imagery, the study and application of SVM is extensive and well known [107]. Most implementations of SVM require the solution of the following optimisation problem [130]:

$$\min_{w,\xi} \frac{1}{2} w^{\mathrm{T}} w + C e^{T} \xi \tag{6.1}$$

subject to $y_i(w^{\mathrm{T}}\phi(x_i) + b) \geq 1 - \xi_i$ for $i = 1 \ldots m$ where $m$ is the number of training data points, $w$ is the hyperplane normal vector, $\phi$ is the kernel function, $e$ is the all 1's vector and $\xi$ is the vector of slack variables. The parameter $C$ represents the magnitude of penalisation. If $C$ is a large value, the optimal solution defines narrower margins in order to accommodate the misclassified training data points; in contrast, smaller values of $C$ lead to wider margins [127]. The penalisation strategy here is uniform and thus equally applied regardless of the class and the data point being analysed.

This is not limited to SVM. Indeed, in conventional supervised classification methods the aim is to minimise the general misclassification rate and thus all types of misclassification are regarded equally severe [16]. A more general approach is to consider misclassifications as not equal. That is, some errors are regarded as more costly than others. This difference is then utilised to inform the learning algorithm during the classification induction stage, and drive the induction process in more sensitive way. There are essentially two ways to implement a cost-sensitive approach: the class-dependent and the instance-dependent. However, which approach is the more suitable dependents of the problem at hand. Next are presented two implementations of these approaches: the BSVM implements implementing of the class-dependent cost definition and Weighted Support Vector Machines (WSVM) implementing the instance-dependent.

### 6.2.1 Bias SVM and weighted SVM

The BSVM is an adaption of the classical formulation of the SVM to handle unlabelled data [96]. This is done by defining different cost values to the positive and to the negative (unlabelled) classes.

$$\min_{w,\xi} \frac{1}{2} w^{\mathrm{T}} w + C_p e^{\mathrm{T}} \xi_p + C_n e^{\mathrm{T}} \xi_n \tag{6.2}$$

subject to $y_i(w^{\mathrm{T}}\phi(x_i) + b) \geq 1 - \xi_i$ for $i = 1 \ldots m$. The vector $\xi_p$ is the vector of slack variables of the positive data points, and $\xi_n$ is the vector of slack variables of the negative data points. By varying $C_p$ and $C_n$ is possible to penalise the positive class and the negative class differently. Intuitively, the cost values are assigned such that

$C_p$ is a big value compared to $C_n$, because the positive class was defined by an human analyst, and thus assumed correct, and the negative class is originated from a random sample of pixels, and thus possibly containing positive data points [109]. However, there is no clear indication of how to define those parameters and trail-and-error is generally recommended [96].

Differently from the SVM and BSVM, the WSVM implements an instance-dependent cost scheme, that is instead to penalising classes, like with the BSVM, or all data points equally, like with the SVM, the goal is to penalise individual data points. A way to adapt the SVM approach to inform the optimisation problem that some points are more relevant than others is by incorporating a weight vector that assigns different cost values to different data points [67, 145]. The original SVM problem is thus reformulated in the following way:

$$\min_{\boldsymbol{w},\xi} \frac{1}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w} + C\boldsymbol{\sigma}^{\mathrm{T}}\xi \tag{6.3}$$

subject to $y_i(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\phi}(\boldsymbol{x}_i)+b) \geq 1-\xi_i$ for $i = 1\ldots m$, where $\boldsymbol{\sigma}$ is the vector of weights. The user can then set different weights to different data points according to a predetermined criterion. Applying the Karush-Kuhn-Tucker (KKT) conditions, the original WSVM problem can be reformulated in its dual form [145]:

$$\min_{\alpha} \frac{1}{2}\sum_{i,j}\alpha_i\alpha_j y_i y_j K(\boldsymbol{x}_i,\boldsymbol{x}_j) - \sum_i \alpha_i \tag{6.4}$$

subject to $\sum_i y_i\alpha_i = 0$ and $0 \leq \alpha_i \leq C\sigma_i$ for $i = 1\ldots m$. Note that, unlike problem (6.3), the Lagrange multipliers are now bounded according to its weight. This allows the learning process to penalise the misclassification of some points differently from other points.

### 6.2.2 One-class SVM

In its origin, the SVM was developed to solve binary classification problems with linearly separable classes. However, the same principles can be applied to solve one-class problems, also know as novelty detection problems [128], that consist in detecting objects from a particular class. This class is often called target class or class of interest. These problems differ greatly from the standard supervised classification in the sense that the training set is composed exclusively by data points from the target class and thus there are no counterexamples to define the classification space outside the class of interest. One-class classification has been utilised in a variety of applications [126] and has great potential in remotely sensed data processing. There are two approaches to one-class classification based on SVM principles, OCSVM [128] and the Support Vector Data Description (SVDD) [137]. In this paper, however, focus is in the use of OCSVM.

The idea behind the OCSVM is to determine a function that signals positive if the given data point belongs to the target class and negative otherwise. To achieve that the classification space origin is treated as the only available member of the non-target class. The problem is then solved by finding a hyperplane with maximum margin separation from the origin. Non-linear problems are dealt with a kernel function as in the binary SVM. The OCSVM optimisation problem is formulated as follows [128]:

$$min_{w,\xi,\rho} \frac{1}{2} w^{\mathrm{T}} w - \rho + \frac{1}{\nu m} \sum_{i} \xi_i \tag{6.5}$$

subject to $w^{\mathrm{T}} \phi(x_i) \geq \rho - \xi_i$ and $\xi_i \geq 0$. Here, $m$ is the number of training data points, $w$ is the vector perpendicular to the hyperplane that defines the target class boundaries and $\rho$ is the distance to the origin. The function $\phi$ is related with the kernel function [128]. The use of slack variables $\xi_i$ used in the OCSVM to allow the presence of class outliers, similar to binary SVM. The parameter $\nu$ ranges from 0 to 1 and controls the trade-off between the number of data points of the training set labelled as positive by the OCSVM decision function $f(x) = sign(w^{\mathrm{T}} \phi(x) - \rho)$.

### 6.2.3 Free-parameter tuning

The development of a learning algorithm requires the use of accuracy metrics to assess the quality and compare the performance of alternative classifiers. In particular, the determination of these free-parameters is an important step. Indeed, there are empirical evidence suggesting that parameter tuning is often more important than the choice of algorithm [17], SVM being particularly harder to tune than other classification procedures [89]. Thus the selection of the correct performance metric is a critical step.

For example, when fine-tuning a classification algorithm, it is often necessary to compute an accuracy metric to determine the parameterisation that yields on average the highest accuracy value. Although commonly used, the overall classification accuracy (the proportion of correctly classified data points) may not a reliable metric, if the training set is imbalance. That is if the training data of one the classes outnumbers the training data of the other class [141]. This is because the performance of the classifier on the larger class dominates the behaviour of this metric, and thus it gives optimistically biased results [145]. Indeed, the definition of the accuracy metric is particular important for binary classification, since the performance of the classifiers can be particularly sensitive to the classes' relative size [129, 145]. In this conditions, the result of the tuning process may be unreliable not because of the process but rather because of the accuracy metric employed in it. If the training data set is imbalanced and the classification accuracy is utilised, the outcome of the tuning process will indicate that a particular parameterisation is the one with the highest classification accuracy but may indeed be biased towards the majority class, since that parameterisation may yield a

classifier that classifies very accurately the majority class in detriment of the minority class [67]. Since often the class of interest is just a small component of the training, the classifier would then underestimate the true extension of this small but important class.

There are better alternative accuracy metrics to the classification accuracy, for example sensitivity and specificity [61]. Sensitivity is the proportion of true positives correctly classified, while specificity is the proportion of true negatives correctly classified [145]. Note that in binary classification, classification accuracy may not be a reliable indicator particularly if the data set is imbalanced, since the influence of the majority class is much higher than that of the minority class [67]. Alternatively, other quality metrics can be used, such as sensitivity and specificity [145]. Sensitivity is the proportion of true positives correctly classified and specificity is the proportion of true negatives correctly classified [61]. Effectively, sensitivity is the producer's accuracy of the positive class while specificity is the producer's class of the negative class. In this way, sensitivity indicates how good the classifier is recognising positive cases and specificity indicates how good the classifier is recognising negative cases [145].

Often sensitivity and specificity are combined in one metric for better analysis and comparison [135]. In particular, the geometric mean between sensitivity ($s$) and specificity ($S$) [16, 84], equation (6.6), is particularly useful:

$$G = \sqrt{sS} \tag{6.6}$$

The geometric mean ($G$) indicates the balance between classification performances on the positive and negative class. High misclassification rate in the positive class will lead to a low geometric mean value, even if all negative data points are correctly classified [67]. Similarly if the classifiers shows high misclassification in the negative class. In this way, if both sensitivity and specificity are high, the geometric mean $G$ is also a high value; but if one of the component accuracies, sensitivity or specificity, is low, the geometric mean $G$ is affected by it. Thus a classifier with high geometric mean is highly desirable for class specific mapping [110], and hence $G$ can be used to fine-tune binary algorithms of classification.

A particular observation is necessary for BSVM. Since the positive class and the negative class are penalised differently, the BSVM has effectively two penalisation variables, which complicates the grid-search optimisation process. In general, the penalisation cost of the positive class should have a large value compared with that of the negative class should have a small value, because it is unknown whether the unlabelled samples are actually positive or negatives. However there is no clear criterion to adjust these parameters and often the user has to resort to trial-and-error [96].

Like SVM, OCSVM algorithm depends on free-parameters that need to be set to develop the classifier. These free-parameters consist in kernel parameters, for example the radial factor of the radial-basis kernel function and the degree of the polynomial

kernel, and regularisation parameters. In the case of OCSVM, the regularisation parameter is $\nu$, ranging from 0 to 1, that defines the upper bound of the fraction of training data points regarded as outliers and and the lower bound of the fraction of training data points regarded as support vectors [128]. In binary and multi-class classification, the determination of the free-parameters is often done by grid-search cross-validation using the classification accuracy as metric [9, 30]. However, the training set used with these type of classifier does not contain data points outside the class of interest, and thus it is only possible to assess the sensitivity of the classifier in the cross-validation process [94, 109]. Using only the sensitivity to parameterise a classification algorithm may result in a classifier with high sensitivity but low specificity, overestimating the true extension of the class of interest.

$$\hat{\theta} = argmax_{\theta} \frac{\frac{1}{n}\sum_k I(f_\theta(\boldsymbol{x}_k) = +1)}{N_{sv}(f_\theta)} \tag{6.7}$$

To minimise the effects of this limitation, the cross-validation process can be carried out using the ratio between the sensitivity and the number of support vectors as metric [7, 109] (equation 6.7), where $n$ is number of testing data points, $I$ is the characteristic function, $f_\theta$ is the OCSVM decision function parametrised with $\theta$ and $N_{sv}$ is the number of support vectors in $f_\theta$. This ratio enforces high sensibility while limiting model complexity (the number of support vectors) which usually indicates good model generalisation ability [7].

## 6.3 Methods

### 6.3.1 Study area

The study area is located in Saloum river delta in Senegal, Africa (fig. 6.1). The area is predominantly flat with altitudes ranging from below sea level in the estuarine zone to about 40 m above mean sea level inland. The climate is Sudano-Sahelian type with a long dry season from November to June and a 4-month rainy season from July to October [31, 38]. The regional annual precipitation, which is the main source of freshwater recharge to the superficial aquifer, increases southward from 600 to 1000 mm. The hydrologic system of the region is dominated by the river Saloum, its two tributaries (Bandiala and Diomboss), and numerous small streams locally called "bolons". Downstream, it forms a large low-lying estuary bearing tidal wetlands, a mangrove ecosystem, and vast areas of denuded saline soils locally called "tan" [31]. The largest land cover classes present in the study area are water, mangrove species, shrubs, savannah and bare soil. The main crop is millet and the urban settlements are usually small and sparse. Saltpans develop to the north because of excessive salinity [106]. In this paper interest is focused on one type of mangrove, High Mangrove (HM). HM is generally characterised by a dense and tall canopy and is composed

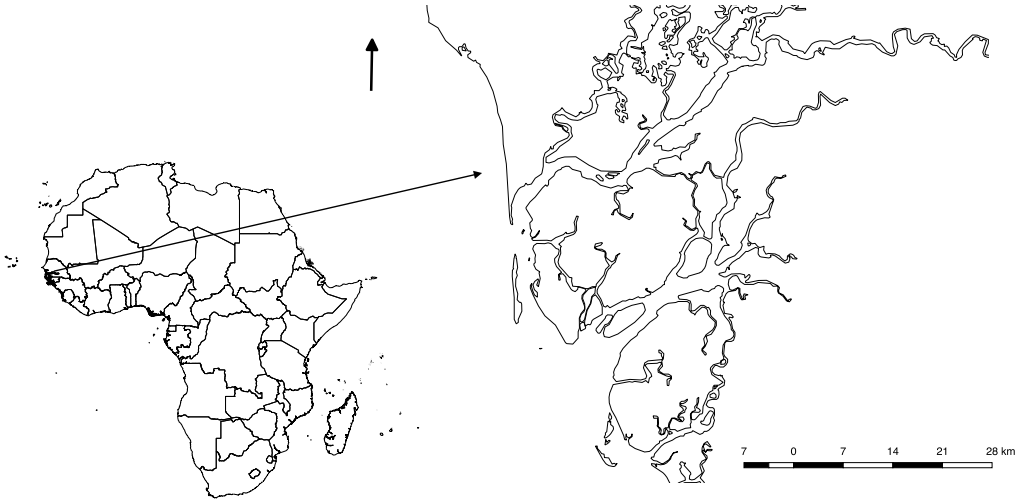



Figure 6.1: Saloum river delta in Senegal.

by species like *Rhizophora racemose*, *Rhizophora mangle* and *Avicennia Africana* [32]. The Saloum river delta was designated a United Nations Educational, Scientific and Cultural Organization (UNESCO) World Heritage site for its remarkable natural environment and extensive biodiversity and is listed in the Ramsar List of Wetlands of International Importance [106]. Particularly important is Saloum's mangrove system, occupying roughly 180 000 ha supporting a wide variety of fauna and flora, and the local economy [106].

### 6.3.2 Remotely sensed data and training set

Remotely sensed data of the study area were acquired on 9 February 2015 by Landsat 8 and downloaded from United States Geologic Survey (USGS) Global Visualization Viewer (GLOVIS). In this study all non-thermal bands (bands 2 to 7) have been used. Since only one image was utilised for analysis and the atmosphere may be considered to be homogeneous within the study area, atmospheric correction was not necessary [132]. The digital numbers were normalised using the max-min rule to range from 0 to 1. The study area is composed by eight large land cover classes: water, high-mangrove, low-mangrove, bare soil, savannah, shrubs humid areas and burnt areas. The training set comprised of 100 pixels per class for each of the eight land cover classes.

### 6.3.3 Experiments

Four experiments were conducted where the first two experiments were used as benchmark. In the first, a OCSVM classifier was developed using only training data of the class of interest, HM. This experiment is thus labeled as OCSVM. The kernel function utilised in the analysis was the radial-basis function and its free-parameters were fine-tuned using 10-fold cross-validation, as described in section 6.2.3. From this analysis

the free-parameters were set as $\gamma = 0.0000610$ and $\nu = 0.025$.

In the second experiment, an image analyst collected 100 points per land cover present in the study area, comprising a total of 800 data points. The data points labeled as high-mangrove were reclassified as positive (class of interest) and the remain training points were reclassified as negative (class of no interest). Next a SVM was developed to discriminate exclusively the class of interest. This experiment is then labeled as SVM. The kernel function utilised for analysis was the radial-basis and its free-parameters were set with 10-fold cross-validation as $\gamma = 2$ and $C = 0.125$, using the geometric mean as described in section 6.2.3. This approach consists in a common binarisation process of the classification problem and has been successfully used in previous studies, for example [13], and extensively studied by machine learning researchers [e.g. 41, 57, 82].

The third and fourth experiments were conducted in a semi-supervised way. Thus a simple random of pixels was utilised to collect random pixels throughout the study scene, composed by 1000 pixels. Following the semi-supervised approach, these were then labelled negative without individual verification [21, 94]. Next, only the class of interest was sampled by an analyst, similar to what happened with the one-class approach. In the third experiment, a BSVM classifier was trained. The kernel function utilised for analysis was the radial-basis and its parameters, $\gamma$, $C_p$ and $C_n$ were set by trail-and-error ensuring $C_n < C_p$, since negative class may contain mislabelled data points. From this analysis, $\gamma = 2$, $C_p = 256$ and $C_n = 0.03125$.

In the fourth experiment, the proposed approach was developed. That is, utilise a WSVM classifier trained in a semi-supervised fashion. Here the same sample that was utilised to train the BSVM was also used to train the WSVM. However, differently from the BSVM, with the WSVM a instance-dependent cost-sensitive approach was implemented to minimise the effect of the mislabeled data points in the learning process. To this end, the following heuristics was applied: negative data points that are spectrally close to known positive training points are likely to be mislabeled, and thus must have to have less impact in the learning process. On the other hand, random points dissimilar to known positive points are likely to be correctly labelled and thus are important for a correct learning algorithm. Note that the labels of the positive points, that were collected manually by the user, are considered certain and thus correct. But negative points, which were randomly selected and blindly labelled as negative, may be misclassified. The number of negative training data points that are mislabelled is expected to be small, since the area occupied by the class of interest is also expected to be small (roughly 10% from previous studies such as [31]).

The function utilised to relate the spectral distance with the nearest positive point was the exponential function in equation 6.8:

$$w_i = 1 - exp(-\sigma d_i^2) \tag{6.8}$$

where $w_i$ is the weight of the $i^{th}$ random point and $d_i^2$ is the squared euclidean distance of the $i^{th}$ random point to its nearest positive point in the feature space. The free-parameter $\sigma > 0$ is utilised as a smoothing parameter; large values of $\sigma$ increases the average weight of the points, while small values reduces it. Note that the maximum assigned weight is 1 and the smallest is asymptotically 0. Thus the misclassification of data points with big weights (close to 1) are more costly that the misclassification of points with less weight (close to 0). In this way the learning process is informed of which training points are more important to define the decision boundaries. Since it is necessary to assign a weight value to all training data points, the positive points were assigned the maximum weight 1 because these are considered certain and correctly labeled.

Note that this weighting model is not necessarily unique. Indeed, any function assigning distances to the interval $]0, 1]$ may be used, such as the inverse of the squared distance. However, the purpose of this study is not determine which weight assigning functions are the most suitable under given conditions, but rather to show the general effectiveness of the method. Thus only exponential function was utilised.

Similar to the SVM, kernel function that was used was the radial-basis function and its free-parameters were defined using a 10-fold cross-validation process with the geometric mean as metric. From this analysis these were set as $\gamma = 2$ and $C = 512$. The values of $\sigma$ were set by trail-and-error. A range of values were tested ranging from very small (0.01) to large (10); at the end the best value for $\sigma$ was 1. All weight values were then normalised using the maximum weight. From this analysis, the weights of the negative data points ranged from 0.001 to 0.86.

All experiments where conducted with LibSVM version 3.21 and LibSVM-weights version 3.20.

### 6.3.4 Classification accuracy and comparison

Classification accuracy was estimated using an independent testing set of 2000 simple random pixels. An image analyst visually classified each pixel, labelling the point as positive (belonging to high-mangrove) or negative (not belonging to high-mangrove) in the same year as the image acquisition with support of Google Earth. From this analysis, 107 pixels were labeled as positive and 1893 where labeled as negatives. The accuracy of each classification was expressed in terms of the proportion of correctly classified testing data points, and also using sensitivity and specificity. Sensitivity is the proportion of positive pixels correctly classified, while specificity is the proportion of negative pixels correctly classified [9]. Since a single testing set was used for each test site, the statistical significance of the difference in overall accuracy between different classification approaches will be assessed using the McNemar test [49].

The McNemar test is based on a binary contingency table in which pixels are classified as correctly or incorrectly allocated by the two classifiers under comparison. The

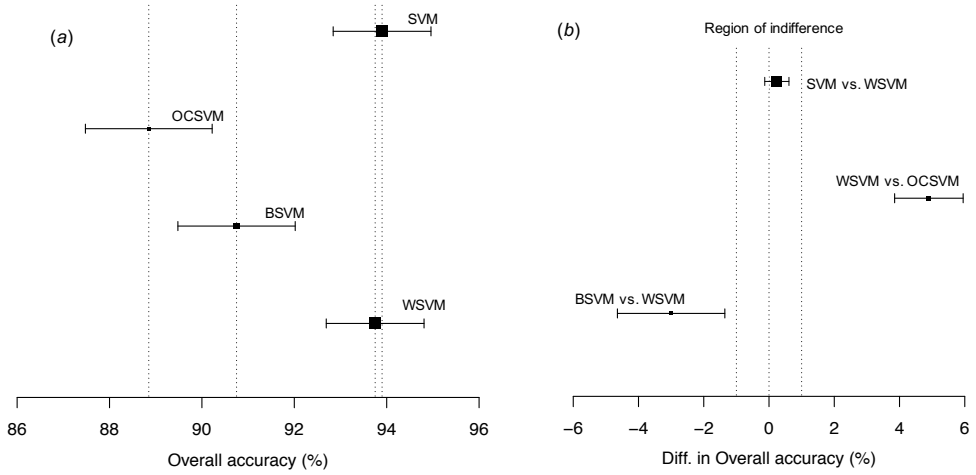



Figure 6.2: The overall accuracies of each method and their respective 95% confidence interval.

main diagonal of this table shows the number of pixels on which both classifiers were correct and on which both classifiers were incorrect. The McNemar test however focus on proportion of pixels where one classifier was correct but the other was incorrect. The analysis will be based upon the evaluation of the $100(1-\alpha)\%$ confidence interval, where $\alpha$ is the level of significance, for the difference between two accuracy values expressed as proportions (say $p_1$ and $p_2$) expressed as [42]:

$$p_2 - p_1 \pm z_\alpha SE \tag{6.9}$$

where the term $SE$ is the standard error derived of the difference between the proportions, which can be determined by [42]:

$$SE = \sqrt{\frac{p_{01} + p_{10} - (p_{01} - p_{10})^2}{n}} \tag{6.10}$$

where $p_{10}$ the proportion of testing pixels where the first classifier was correct and the second was incorrect and $p_{01}$ the proportion of testing pixels where the first classifier was incorrect and the second was correct. In this way, the statistical assessment of the differences was conducted to determine if these were significantly different or not [49]. To perform this analysis is necessary to define the zone of indifference [49]. This is, the largest amount of allowable difference that determine if the methods are considered equivalent or non-inferior [10]. In this evaluation it was assumed that the zone of indifference was 1.00%. Although this value was selected arbitrarily, ensures that small differences in accuracy are inconsequential [113].

## 6.4 Results and discussion

Figure 6.2 frame (*a*) presents the overall accuracy obtained with each method and their respective 95% confidence interval. All methods yielded high classification accuracy:

SVM achieved 93.90% with confidence interval at 95% confidence level of [92.84%, 94.96%], OCSVM achieved 88.85% with [87.48%, 90.22%], BSVM achieved 90.75 with [89.48%, 92.02%] and the WSVM 93.75% with [92.69%, 94.81%]. In frame (*b*) it is summarised the statistical comparison between the classifications. The difference between the classification accuracies yielded by SVM and WSVM was 0.15% ranging from -0.09% to 0.39% at 95% confidence interval. The confidence interval for the difference in accuracy is within the region of indifference ([-1%, 1%]) and thus provides evidence for the non-inferiority of WSVM. In other words, the statistical analysis shows that the classification accuracy derived from WSVM is non-inferior to that of SVM at 5% level of significance. However, WSVM was developed without the need to collect training data points in secondary classes, which contrast with SVM where all classes present in the study area were incorporated in the training set. Indeed, the sampling effort was similar to that of OCSVM. The difference between the classification accuracies yielded by WSVM and OCSVM was 4.90% ranging from 3.85% to 5.95% at 95% confidence interval. The confidence interval is outside and above the region of indifference without intersecting it. This provides evidences for the difference between the classifications at 5% significance level. The difference between the classification accuracies yielded by BSVM and WSVM was -3.00% ranging from -4.55% to -1.35% at 95% confidence interval. The confidence interval is outside and bellow the region of indifference without intersecting it, however the difference is smaller to that with OCSVM. Note that if the region of indifference were increased to [-2%, 2%], the conclusion would not change, since the interval defining the difference between BSVM and WSVM would not cross zero, although there would be an overlapping region. The main difference between BSVM and WSVM is in the way the learning algorithms deal with the negative class. BSVM penalises all points of the negative class in the same. The WSVM, on the other hand, particularises the penalisation. This leads WSVM to trust some negative training data points in the same way as a positive training data point and disregard some negative points as blunders.

In figure 6.3, Sensitivity and specificity of each method is under analysis. The dashed line represents the 1:1 line. Thus, any point on the line indicates a method with equal sensitivity and specificity. However, if a method is localised above the line, this indicates higher specificity values than sensitivity, which indicates a method that underestimates the extension of the class of interest. But if a method is bellow the 1:1 line, this indicates a method with higher sensitivity and lower specificity, which suggests the method is overestimating the extension of the class of interest.

Particular informative is the specificity that quantifies how good each method is discriminating negative data points. SVM, BSVM and WSVM yield specificity accuracies above 90% while OCSVM yield a respective value roughly 5% lower. This indicates that OCSVM is committing more pixels of the classes of no interest to the class of interest, that is the number of false positives in larger in this method. Geometrically this suggests an over-expansion of the true extension of the class of interest.

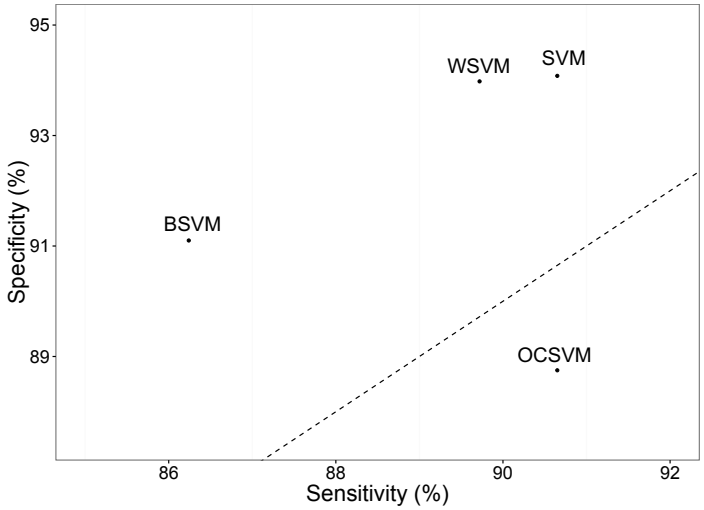



Figure 6.3: Sensitivity and specificity of each method under analysis. The dashed line represents the 1:1 line.

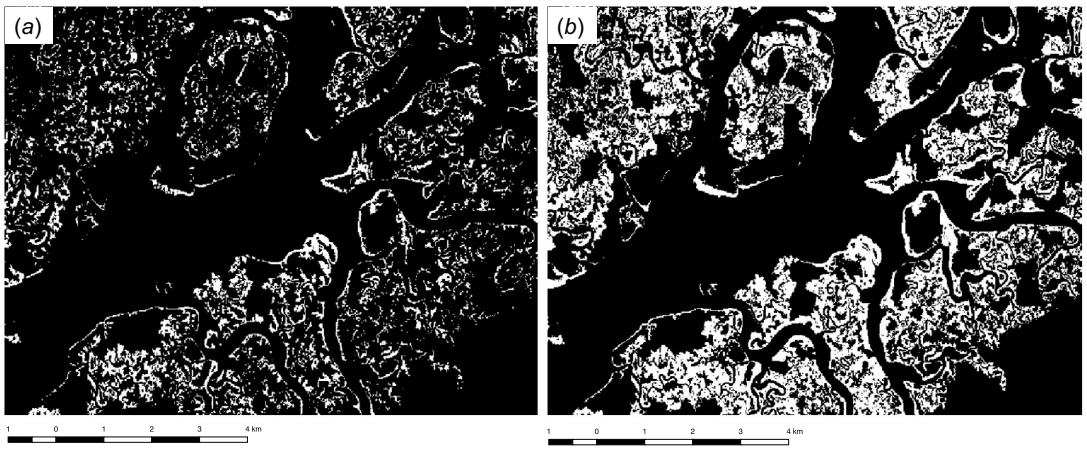



Figure 6.4: Two excerpts of (*a*) the WSVM map and (*b*) the OCSVM map.

A visual inspection of the outputted maps, in figure 6.4, shows that OCSVM is overestimating the extension of the the class of interest (). This can be explained by the fact the one-class classifier does not have access to information of the classification space outside the class of interest and, thus, this may lead the learning algorithm to overextend the decision boundary [137], resulting in an overestimation of the positive class.

Sensitivity values were also high: 90.65% for SVM and OCSVM, 86.24% for BSVM and 89.72% for WSVM. The high value yield by OCSVM can be explained by the over-extension of the decision boundary, which is extended enough to accommodate a large number of positive testing data points. The lower value yield by WSVM and BSVM, when compared to SVM, can be a consequence of the way the negative data set was sampled. With training set used by SVM, where all classes present in the study site were sample, the negative class with well characterised. In other words, all

regions of the negative classification space are represented in the training set. That may not happen with the WSVM and BSVM, where this training data was randomly generated. In other words, some regions of the classification space may have not been sample and thus the resulted classifier may be committing untrained areas to the class of interest. Indeed, WSVM and BSVM errors occur mostly in forest and shrub class in areas spectrally similar to the class of interest, high-mangrove.

## 6.5 Conclusions

This paper proposes and tests a method that aims to reduce the training sampling effort in class specific mapping. The motivation for the development of this method comes from the fact that although one-class classification requires the user to collect only training data from the class of interest, which represents a great reduction in training effort, these methods may overestimate the class of interest. Typically if information about the class of interest and the classes of no interest is available, binary classifiers tend to achieve higher classification accuracy. However, these methods require the user to collect training data from classes of no interest. The proposed method combines the sampling effort required by the one-class classifier with the discrimination capability of a binary class using a semi-supervised approach with cost-sensitive learning. The results indicate that although the four methods under analysis achieved high overall classification accuracy, the one-class classification achieved the lowest classification accuracy (88.85%) due to the overestimation the extension of the class of interest, and the proposed method (93.75%) was non-inferior to the binary classification (93.90%) at 5% level of significance, requiring however less training effort.

Chapter 7

# Final remarks

The technology associated to remote sensing has evolved considerably giving raise to a vast group of sensors operating at a wide range of scales, temporal frequencies and spectral resolutions. Combined with ancillary data, such as past cartography and a large catalogue of historical remote sensing imagery, the volume of data available for a given user is vast, accommodating a long list of possible uses and thus user requirements. With this ever grow number of users, the need of maps that specifically answer to specific user needs becomes fundamental. This was the starting point of this dissertation.

In general it was shown that adapting the classification process to the user needs yields better results when compared with conventional multi-class approach. This improvements happens either by an increase in the classification accuracy of the classes of interest or by minimising the training requirements. In particular, in the first case, a data set that was utilised to produce an exhaustive land cover map was adapted to the need of specific class mapping and yielded better results regarding the discrimination of the classes of interest. In the second and third case, the process was undertaken from the beginning and only the class of interest was sampled. This represented a considerable decrease in training requirements without, however, loose classification accuracy.

For future research, this dissertation presents three possible research lines not necessarily related with specific class mapping but with the automatic classification of satellite images.

The first concerns the model selection. Model selection is an important step in specific class mapping but also in the classification process in general. To find a good parameterisation is not uncommon for a cross-validation process to train hundreds of classifiers and once assessed are discarded. Can those classifiers be used instead of

rejected? One hypothesis is the design of an ensemble with all those classifiers. This could be profitable in the sense that often errors committed by a particular classifier are not committed by a similar classifier with a different parameterisation. If this question is answered, it will save time (because there will be no need for model selection) and resources (because the user no longer needs to be an expert to correctly fine tune the classifier).

The second is concerned with a fundamental assumption in automatic classification. For mathematical purposes, the classification problem is assumed to be inserted in an infinite space where data points are infinite. In other domains, like fraud detection, engine anomaly identification, the classification space, that is the set comprehending all data points that are to be classified, is large enough that is reasonable to assume infinite. Unlike these domains, in the classification of remotely sensed data for the purpose of producing land cover maps the classification space finite and well known. Although images can be large, users' concerns are with a small region of land, the set of pixels that need to be classified is small when compared with other domains. With a ever grow computing power, how can that finitude be leveraged to produce better land cover maps?

The third is concerned with the training data. More work could be done to understand the important factors and characteristics that make a data set a good training set. The idea is not new, and literature provides some works investigating this issue. However, it has apparently been forgotten. Thus, exploring what makes a data set a good training set is perhaps the most important of these three suggestions for future research.

This dissertation ends with two recommendations or observations about the automatic classification of remotely sensed data with the aim of producing a land cover map. These recommendations are the outcome of not only a comprehensive analysis of the literature but above all of the author's experience in the production of land cover maps by automatic means.

The first point concerns the importance that is typically attributed to the classifier. The classifier is only one component in the mapping process and is often regarded as the reason a particular map is good or bad. The classifier itself is no more important than any other step in the mapping process. There are a multitude of different approaches to supervised classifier, each one with its own particularities and, one can assume fairly certain that there is no such thing as the ultimate best learning algorithm. Indeed, the same learning algorithm will render different classifier under different conditions (training data, parameterisation, etc). For the spotlight to move to the learning algorithm, other components of the classification process are often forgotten. For example, the pre-processing phase (feature reduction, scaling, etc) and model selection are often omitted and their effects are regarded inconsequential. Thus this recommendation focus on the importance of the process; all steps are relevant and are equally contributors to the final outcome.

The second point concerns the disassociation of the classification process from the data domain. The classification itself is pointless if it is not incorporate in an applicational domain. In the present case that domain is the classification of remotely sensed data to produce a land cover maps. More time could/must be put on the characterisation of the problem itself. Effectively, it is fundamental to perform a comprehensive analysis of the landscape of the region of interest and select the most adequate land cover nomenclature to represent them and understand the limits of what can be extracted from the data: it is impossible to discriminate beyond the geometrical resolution provided by the imagery and it is very difficult to discriminate extremely similar classes. In other words, one must be reasonable and not expect to automatically distinguish between land cover classes that are not separable within the spatial, spectral and temporal resolutions provided by the data set. In other words, one must recognise that classification process depends on numerous factors that do not depend only on the classifier.

To end this document is important to highlight that this dissertation was never intended to be an ending point. Numerous research directions were always possible during the course of this investigation and some are still open. Which branch to follow? was the most difficult question that had to be answered in this research. There are unopened doors and unturned stones throughout this investigation which indicates that this inquiry is not the end but only a starting point for new lines of inquiry.

# Bibliography


[1] S. S. A. Guerrero-Curieses A. Biasiotto and G. Moser. "Supervised classification of remote sensing images with unknown classes". In: *Geoscience and Remote Sensing Symposium, 2002. IGARSS '02. 2002 IEEE International* (2002).

[2] R. Akbani, S. Kwek, and N. Japkowicz. "Applying Support Vector Machines to Imbalanced Datasets". In: *In Proceedings of the 15th European Conference on Machine Learning (ECML)*. 2004, pp. 39–50.

[3] C. Alcantara, T. Kuemmerle, A. V. Prishchepov, and V. C. Radeloff. "Mapping abandoned agriculture with multi-temporal MODIS satellite data". In: *Remote Sensing of Environment* 124 (2012), pp. 334–347. ISSN: 00344257. DOI: 10.1016/j.rse.2012.05.019.

[4] M. Amer, M. Goldstein, and S. Abdennadher. "Enhancing One-class Support Vector Machines for Unsupervised Anomaly Detection". In: *Proceedings of the ACM SIGKDD Workshop on Outlier Detection and Description*. ODD '13. Chicago, Illinois: ACM, 2013, pp. 8–15. ISBN: 978-1-4503-2335-2. DOI: 10.1145/2500853.2500857.

[5] P. M. Atkinson, G. M. Foody, P. W. Gething, A. Mathur, and C. K. Kelly. "Investigating spatial structure in specific tree species in ancient semi-natural woodland using remote sensing and marked point pattern analysis". In: *Ecography* 30.1 (2007), pp. 88–104. ISSN: 09067590. DOI: 10.1111/j.2006.0906-7590.04726.x.

[6] C. A. Baldeck, G. P. Asner, R. E. Martin, C. B. Anderson, D. E. Knapp, J. R. Kellner, and S. J. Wright. "Operational tree species mapping in a diverse tropical forest with airborne imaging spectroscopy". In: *PLoS ONE* 10.7 (2015). ISSN: 19326203. DOI: 10.1371/journal.pone.0118403.

[7] A. Banerjee, P. Burlina, and C. Diehl. "A support vector method for anomaly detection in hyperspectral imagery". In: *IEEE Trans. Geosci. Re- mote Sens., vol. 44, no* 44.8 (Aug. 2006), pp. 2282–2291.

[8] C. Bellinger, S. Sharma, and N. Japkowicz. "11th International Conference on Machine Learning and Applications (ICMLA)". In: *One-class versus binary classification: which and when?* 2012, pp. 102–104.

[9] C. M. Bishop. *Pattern recognition and machine learning, information science and statistics*. Berlin: Springer, 2006.

[10] W. C. Blackwelder. "Proving the null hypothesis in clinical trials". In: *Crontrol Clinic Trails* 3.4 (1982), pp. 345–353.

[11] B. E. Boser, I. M. Guyon, and V. N. Vapnik. "A Training Algorithm for Optimal Margin Classifiers". In: *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*. COLT '92. Pittsburgh, Pennsylvania, USA: ACM, 1992, pp. 144–152. ISBN: 0-89791-497-X. DOI: 10.1145/130385.130401. URL: http://doi.acm.org/10.1145/130385.130401.

[12] L. Bottou, C. Cortes, J. S. Denker, H. Drucker, I. Guyon, L. D. Jackel, Y. Lecun, U. A. Müller, Säckinger, P. Simard, and V. Vapnik. "Comparison of classifier methods: a case study in handwritten digit recognition". In: *Proceedings of the 12th International Conference on Pattern Recognition and Neural Networks, Jerusalem*. IEEE Computer Society Press, 1994, pp. 77–87.

[13] D. Boyd, C. Sanchez-Hernandez, and G. Foody. "Mapping a specific class for priority habitats monitoring from satellite sensor data". In: *International Journal of Remote Sensing* 27.March 2015 (2006), pp. 37–41. ISSN: 0143-1161. DOI: 10.1080/01431160600554348.

[14] A. P. Bradley. "The Use of the Area Under the ROC Curve in the Evaluation of Machine Learning Algorithms". In: *Pattern Recogn.* 30.7 (July 1997), pp. 1145–1159. ISSN: 0031-3203. DOI: 10.1016/S0031-3203(96)00142-2. URL: http://dx.doi.org/10.1016/S0031-3203(96)00142-2.

[15] J. Calpe-maravilla. *Combination of One-Class Remote Sensing Image Classifiers*. 2007.

[16] P. Cao, D. Zhao, and O. Zaiane. "An Optimized Cost-Sensitive SVM for Imbalanced Data Learning". In: *Advances in knowledge discovery and data mining*. 2013, pp. 280–292. ISBN: 978-3-642-37455-5. DOI: 10.1007/978-3-642-37456-2{_}24.

[17] E. Carrizosa and D. Romero Morales. "Supervised classification and mathematical optimization". In: *Computers and Operations Research* 40.1 (2013), pp. 150–165. ISSN: 03050548. DOI: 10.1016/j.cor.2012.05.015.

[18] E. Carrizosa, B. Martín-Barragán, and D. Romero Morales. "A nested heuristic for parameter tuning in Support Vector Machines". In: *Computers and Operations Research* 43 (2014), pp. 328–334. ISSN: 03050548. DOI: 10.1016/j.cor.2013.10.002.

[19] C.-C. Chang and C.-J. Lin. "LIBSVM: A library for support vector machines". In: *ACM Transactions on Intelligent Systems and Technology* 2 (3 2011). Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm, 27:1–27:27.

[20] C.-C. Chang and C.-L. Lin. "LIBSVM: A Library of Support Vector Machines". In: *ACM Transactions on Intelligent Systems and Technology* 2 (3 2011), pp. 1–27.

[21] e. . C. f. . U. i. . . p. . . t. . S. v. . . y. . . Chapelle O., Scholkopf B. and Zien A. edition = The MIT Pr.

[22] N. V. Chawla. "Data Mining for Imbalanced Datasets: An Overview". In: *Data Mining and Knowledge Discovery Handbook* (2005), pp. 853–867. ISSN: 978-0-387-24435-8. DOI: 10.1007/0-387-25465-X{_}40. arXiv: 978-0-387-09823-4{_}45 [10.1007].

[23] W. G. Cochran. *Sampling techniques*. Wiley series in probability and mathematical statistics-applied. New York, Chichester, Brisbane: J. Wiley & sons, 1977. ISBN: 0-471-02939-4. URL: http://opac.inria.fr/record=b1077492.

[24] K. Cockx, T. Van de Voorde, and F. Canters. "Quantifying uncertainty in remote sensing-based urban land-use mapping". In: *International Journal of Applied Earth Observation and Geoinformation* 31.1 (2014), pp. 154–166. ISSN: 15698432. DOI: 10.1016/j.jag.2014.03.016.

[25] A. J. Comber. "Land Cover or Land Use?" In: *Journal of Land Use Science* 3 (2008), 199–201.

[26] C. Corbane, S. Lang, K. Pipkins, S. Alleaume, M. Deshayes, V. E. Garcia Millan, T. Strasser, J. Vanden Borre, S. Toon, and F. Michael. "Remote sensing for mapping natural habitats and their conservation status - New opportunities and challenges". In: *International Journal of Applied Earth Observation and Geoinformation* 37 (2015), pp. 7–16. ISSN: 1872826X. DOI: 10.1016/j.jag.2014.11.005.

[27] K. Crammer and Y. Singer. "On the Learnability and Design of Output Codes for Multiclass Problems". In: *In Proceedings of the Thirteenth Annual Conference on Computational Learning Theory*. 2000, pp. 35–46.

[28] M. Dalponte, L. Bruzzone, and D. Gianelle. "Tree species classification in the Southern Alps based on the fusion of very high geometrical resolution multispectral/hyperspectral images and LiDAR data". In: *Remote Sensing of Environment* 123 (2012), pp. 258–270. ISSN: 00344257. DOI: 10.1016/j.rse.2012.03.013. URL: http://dx.doi.org/10.1016/j.rse.2012.03.013.

[29] M. Dalponte, L. T. Ene, M. Marconcini, T. Gobakken, and E. Næsset. "Semi-supervised SVM for individual tree crown species classification". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 110 (2015), pp. 77–87. ISSN: 09242716. DOI: 10.1016/j.isprsjprs.2015.10.010.

[30] N. Deng, Y. Tian, and C. Zhang. *Support Vector Machines: Optimization Based Theory, Algorithms, and Extensions*. CRC Press, 2012.

[31] M. Dieng, J. Silva, M. Goncalves, S. Faye, and M. Caetano. *The Land/Ocean Interactions in the Coastal Zone of West and Central Africa, Estuaries of the World. Estuaries of the World*. 2014. DOI: 10.1007/978-3-319-06388-1_5.

[32] E. S. Diop. "Estuaires holocènes tropicaux. Etude géographique physique comparée des rivières du Sud du Saloum (Sénégal) à la Mellcorée (République de Guinée)". PhD thesis. 1986, p. 522.

[33] S. Du and S. Chen. "Weighted support vector machine for classification". In: *Systems, Man and Cybernetics, 2005 IEEE* 2 (2005), pp. 859–864. DOI: 10.1109/ICSMC.2005.1571749.

[34] C. Elkan and K. Noto. "Learning classifiers from only positive and unlabeled data". In: *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2008, pp. 213–220. ISBN: 9781605581934. DOI: 10.1145/1401890.1401920.

[35] M. Ergul, N. Sen, and O. E. Okman. *Effective training set sampling strategy for SVDD anomaly detection in hyperspectral imagery*. 2014. DOI: 10.1117/12.2051040. URL: http://dx.doi.org/10.1117/12.2051040.

[36] F. E. Fassnacht, H. Latifi, K. Stereczak, A. Modzelewska, M. Lefsky, L. T. Waser, C. Straub, and A. Ghosh. "Review of studies on tree species classification from remotely sensed data". In: *Remote Sensing of Environment* 186 (2016), pp. 64–87. ISSN: 00344257. DOI: 10.1016/j.rse.2016.08.013. arXiv: arXiv:1011.1669v3.

[37] T. Fawcett. "An Introduction to ROC Analysis". In: *Pattern Recogn. Lett.* 27.8 (June 2006), pp. 861–874. ISSN: 0167-8655. DOI: 10.1016/j.patrec.2005.10.010. URL: http://dx.doi.org/10.1016/j.patrec.2005.10.010.

[38] S. Faye, M. Diaw, R. Malou, and A. Faye. "Impacts of climate change on groundwater recharge and salinization of groundwater resources in Senegal". In: *Groundwater and climate in Africa proceeding of the Kampala conference* (2008).

[39] X. Feng, G. Foody, P. Aplin, and S. N. Gosling. "Enhancing the spatial resolution of satellite-derived land surface temperature mapping for urban areas". In: *Sustainable Cities and Society* 19 (2015), pp. 341–348. ISSN: 22106707. DOI: 10.1016/j.scs.2015.04.007.

[40] J.-B. Feret and P. Asner. "Tree Species Discrimination in Tropical Forests Using Airborne Imaging Spectroscopy". In: *IEEE Trans. Geoscience Remote Sensing* 51.1 (2013), pp. 73–84. ISSN: 0196-2892. DOI: 10.1109/TGRS.2012.2199323.

[41] A. Fernandez, V. Lopez, M. Galar, M. J. Del Jesus, and F. Herrera. "Analysing the classification of imbalanced data-sets with multiple classes: Binarization techniques and ad-hoc approaches". In: *Knowledge-Based Systems* 42 (2013), pp. 97–110. ISSN: 09507051. DOI: 10.1016/j.knosys.2013.01.018.

[42] J. Fleiss, B Levin, and M Cho Paik. *Statistical Methods for Rates and Proportions*. 2003, p. 800. ISBN: 0471526290 (cloth : acid-free paper). DOI: 10.1198/tech.2004.s812.

[43] R. Fletcher. *Practical Methods of Optimization; (2Nd Ed.)* New York, NY, USA: Wiley-Interscience, 1987. ISBN: 0-471-91547-5.

[44] G. M. Foody. "Hard and soft classifications by a neural network with a non-exhaustively defined set of classes". In: *International Journal of Remote Sensing* 23.18 (2002), pp. 3853–3864. ISSN: 0143-1161. DOI: 10.1080/01431160110109570.

[45] G. M. Foody. "Supervised image classification by MLP and RBF neural networks with and without an exhaustively defined set of classes". In: *International Journal of Remote Sensing* 25.15 (2004), pp. 3091–3104. ISSN: 0143-1161. DOI: 10.1080/01431160310001648019.

[46] G. M. Foody. "Supervised image classification by MLP and RBF neural networks with and without an exhaustively defined set of classes". In: *International Journal of Remote Sensing* 25.15 (2004), pp. 3091–3104. ISSN: 0143-1161. DOI: 10.1080/01431160310001648019. URL: http://www.informaworld.com/openurl?genre=article{\&}doi=10.1080/01431160310001648019{\&}magic=crossref||D404A21C5BB053405B1A640AFFD44AE3.

[47] G. M. Foody, D. S. Boyd, and C. Sanchez-Hernandez. "Mapping a specific class with an ensemble of classifiers". In: *International Journal of Remote Sensing* 28.8 (2007), pp. 1733–1746. ISSN: 0143-1161. DOI: 10.1080/01431160600962566.

[48] G. M. Foody. "Estimation of sub-pixel land cover composition in the presence of untrained classes". In: *Computers and Geosciences* 26.4 (2000), pp. 469–478. ISSN: 00983004. DOI: 10.1016/S0098-3004(99)00125-9.

[49] G. M. Foody. "Classification accuracy comparison: Hypothesis tests and the use of confidence intervals in evaluations of difference, equivalence and non-inferiority". In: *Remote Sensing of Environment* 113.8 (2009), pp. 1658–1663. ISSN: 00344257. DOI: 10.1016/j.rse.2009.03.014.

[50] G. M. Foody. "Latent class modeling for site- and non-site-specific classification accuracy assessment without ground data". In: *IEEE Transactions on Geoscience and Remote Sensing* 50.7 PART 2 (2012), pp. 2827–2838. ISSN: 01962892. DOI: 10.1109/TGRS.2011.2174156.

[51] G. M. Foody and M. E. J. Cutler. "Mapping the species richness and composition of tropical forests from remotely sensed data with neural networks". In: *Ecological Modelling\rSelected Papers from the Third Conference of the International Society for Ecological Informatics (ISEI), August 26–30, 2002, Grottaferrata, Rome, Italy* 195.1-2 (2006), pp. 37–42. ISSN: 03043800. DOI: 10.1016/j.ecolmodel.2005.11.007.

[52] G. M. Foody and A. Mathur. "Toward intelligent training of supervised image classifications: directing training data acquisition for SVM classification". In: *Remote Sensing of Environment* 93.1-2 (2004), pp. 107–117. ISSN: 00344257. DOI: 10.1016/j.rse.2004.06.017.

[53] G. M. Foody and A. Mathur. "The use of small training sets containing mixed pixels for accurate hard image classification: Training on mixed spectral responses for classification by a SVM". In: *Remote Sensing of Environment* 103.2 (2006), pp. 179–189. ISSN: 00344257. DOI: 10.1016/j.rse.2006.04.001.

[54] G. M. Foody, P. M. Atkinson, P. W. Gething, N. A. Ravenhill, and C. K. Kelly. "Identification of specific tree species in ancient semi-natural woodland from digital aerial sensor imagery". In: *Ecological Applications* 15.4 (2005), pp. 1233–1244. ISSN: 10510761. DOI: Doi10.1890/04-1061.

[55] G. M. Foody, A. Mathur, C. Sanchez-Hernandez, and D. S. Boyd. "Training set size requirements for the classification of a specific class". In: *Remote Sensing of Environment* 104.1 (2006), pp. 1–14. ISSN: 00344257. DOI: 10.1016/j.rse.2006.03.004.

[56] G. Foody. "The effect of a non-exhaustively defined set of classes on neural network classifications". 2001. URL: http://eprints.soton.ac.uk/15216/.

[57] M. Galar, A. Fernandez, E. Barrenechea, H. Bustince, and F. Herrera. "An overview of ensemble methods for binary classifiers in multi-class problems: Experimental study on one-vs-one and one-vs-all schemes". In: *Pattern Recognition* 44.8 (2011), pp. 1761–1776. ISSN: 00313203. DOI: 10.1016/j.patcog.2011.01.017.

[58] B. Gorte and N. Gorte-Kroupnova. "Non-parametric classification algorithm with an unknown class". In: *Proceedings., International Symposium on Computer Vision* (1995).

[59] S. J. Graves, G. P. Asner, R. E. Martin, C. B. Anderson, M. S. Colgan, L. Kalantari, and S. A. Bohlman. "Tree species abundance predictions in a tropical agricultural landscape with a supervised classification model and imbalanced data". In: *Remote Sensing* In Review (2016), pp. 1–21. ISSN: 2072-4292. DOI: 10.3390/rs8020161.

[60] A. D. Gregorio and L. J. M. Jansen. *Land Cover Classification System*. Rome, Italy: FAO, 2000.

[61] T. Hastie, R. Tibshinari, and J. Friedman. *The elements of statistical learning*. Second Edition. Springer: Springer Series in Statistics, 2009.

[62] H. He and E. A. Garcia. "Learning from imbalanced data". In: *IEEE Transactions on Knowledge and Data Engineering* 21.9 (2009), pp. 1263–1284. ISSN: 10414347. DOI: 10.1109/TKDE.2008.239.

[63] H. He and M. Yunqian. *Imbalanced Learning: Foundation, Algorithms and Applications*. The Instit. John Wiley & Sons, Ltd, 2013, p. 216. ISBN: 9781118074626.

[64] K Hempstalk and E Frank. "Discriminating against new classes: one-class versus multi-class classification". In: *Lecture Notes in Artificial Intelligence of Lecture Notes in Computer Science*. 2008, pp. 325–336.

[65] C.-W. Hsu and C.-J. Lin. "A comparison of methods for multiclass support vector machines". In: *IEEE Transactions on Neural Networks* 13.2 (2002), pp. 415–425. ISSN: 1045-9227. DOI: 10.1109/72.991427.

[66] D. S.-X. Huang Yin-Min. "Weighted support vector machine for classification with uneven training class sizes". In: *2005 IEEE International Conference on Systems, Man and Cybernetics* 4.August (2005), pp. 3866–3871. DOI: 10.1109/ICSMC.2005.1571749.

[67] J. P. Hwang, S. Park, and E. Kim. "A new weighted approach to imbalanced data classification problem via support vector machine with quadratic cost function". In: *Expert Systems with Applications* 38.7 (2011), pp. 8580–8585. ISSN: 09574174. DOI: 10.1016/j.eswa.2011.01.061.

[68] L. R. Iverson and D. McKenzie. "Tree-species range shifts in a changing climate: detecting, modeling, assisting". In: *Landscape Ecology* 28.5 (2013), pp. 879–889. ISSN: 1572-9761. DOI: 10.1007/s10980-013-9885-x. URL: http://dx.doi.org/10.1007/s10980-013-9885-x.

[69] B. J. and D. Landgrebe. "Partially supervised classification using weighted unsupervised clustering". In: *IEEE Trans. Geosci. Remote Sens.* 37 (1999), 1073–1079.

[70] S. S. Japkowiciz N. "The class imbalance problem: a systematic study". In: *Intelligent Data Analysis* 6.5 (2002), pp. 1–39.

[71] B. Jeon and D. A. Landgrebe. "Partially supervised classification using weighted unsupervised clustering". In: *IEEE TRANS. ON GEOSCIENCE AND REMOTE SENSING* 37.2 (1999), pp. 1073–1079.

[72] S. S. Keerthi and C.-J. Lin. "Asymptotic Behaviors of Support Vector Machines with Gaussian Kernel". In: *Neural Comput.* 15.7 (July 2003), pp. 1667–1689. ISSN: 0899-7667. DOI: 10.1162/089976603321891855. URL: http://dx.doi.org/10.1162/089976603321891855.

[73] S. S. Khan and M. G. Madden. "One-Class Classification: Taxonomy of Study and Review of Techniques". In: *The Knowledge Engineering Review* 00.January (2004), pp. 1–24. ISSN: 0269-8889. DOI: 10.1017/S000000000000000. arXiv: arXiv:1312.0049v1.

[74] K. B. Kirui, J. G. Kairo, J. Bosire, K. M. Viergever, S. Rudra, M. Huxham, and R. A. Briers. "Mapping of mangrove forest land cover change along the Kenya coastline using Landsat imagery". In: *Ocean and Coastal Management* 83 (2013), pp. 19–24. ISSN: 09645691. DOI: 10.1016/j.ocecoaman.2011.12.004.

[75] S. Knerr, L. Personnaz, and G. Dreyfuss. "Single-layer learning revisited: a stepwise procedure for building and training a neural network". In: *Neurocomputing: Algorithm, Architectures and Applications*. Ed. by J. Fogelmann. Springer, 1990.

[76] R. Kohavi. "A Study of Cross-validation and Bootstrap for Accuracy Estimation and Model Selection". In: *Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 2*. IJCAI'95. Montreal, Quebec, Canada: Morgan Kaufmann Publishers Inc., 1995, pp. 1137–1143. ISBN: 1-55860-363-8. URL: http://dl.acm.org/citation.cfm?id=1643031.1643047.

[77] C. Kontgis, A. Schneider, and M. Ozdogan. "Mapping rice paddy extent and intensification in the Vietnamese Mekong River Delta with dense time stacks of Landsat data". In: *Remote Sensing of Environment* 169 (2015), pp. 255–269. ISSN: 00344257. DOI: 10.1016/j.rse.2015.08.004.

[78] S. Kotsiantis, D. Kanellopoulos, and P. Pintelas. *Handling imbalanced datasets : A review*. 2006. DOI: 10.1007/978-0-387-09823-4{_}45.

[79] B. Krawczyk. "One-class classifier ensemble pruning and weighting with firefly algorithm". In: *Neurocomputing* 150.PB (2015), pp. 490–500. ISSN: 18728286. DOI: 10.1016/j.neucom.2014.07.068.

[80] B. Krawczyk and M. Woźniak. "Wagging for Combining Weighted One-class Support Vector Machines". In: *Procedia Computer Science* 51 (2015), pp. 1565–1573. ISSN: 18770509. DOI: 10.1016/j.procs.2015.05.351.

[81] B. Krawczyk, G. Schaefer, and M. Woźniak. "Combining one-class classifiers for imbalanced classification of breast thermogram features". In: *Proceedings of the 2013 4th International Workshop on Computational Intelligence in Medical Imaging, CIMI 2013 - 2013 IEEE Symposium Series on Computational Intelligence, SSCI 2013*. 2013, pp. 36–41. ISBN: 9781467359191. DOI: 10.1109/CIMI.2013.6583855.

[82] B. Krawczyk, M. Woźniak, and F. Herrera. "On the usefulness of one-class classifier ensembles for decomposition of multi-class problems". In: *Pattern Recognition* 48.12 (2015), pp. 3969–3982. ISSN: 00313203. DOI: 10.1016/j.patcog.2015.06.001.

[83] U. H.-G. Kre. "Advances in Kernel Methods". In: ed. by B. Schölkopf, C. J. C. Burges, and A. J. Smola. Cambridge, MA, USA: MIT Press, 1999. Chap. Pairwise Classification and Support Vector Machines, pp. 255–268. ISBN: 0-262-19416-3. URL: http://dl.acm.org/citation.cfm?id=299094.299108.

[84] M. Kubat and S. Matwin. "Addressing the Curse of Imbalanced Training Sets: One Sided Selection". In: *In Proceedings of the Fourteenth International Conference on Machine Learning*. Vol. 4. 1. 1997, pp. 179–186. ISBN: 9780874216561. DOI: 10.1007/s13398-014-0173-7.2. arXiv: arXiv:1011.1669v3.

[85] L. I. Kuncheva. *Combining Pattern Classifiers: Methods and Algorithms*. Wiley-Interscience, 2004. ISBN: 0471210781.

[86] S. Y. Kung. *Kernel Methods and Machine Learning*. 2014, p. 572. ISBN: 9781107024960. DOI: 10.1017/CBO9781139176224. URL: http://www.amazon.com/Kernel-Methods-Machine-Learning-Kung/dp/110702496X/ref=pd{_}sim{_}sbs{_}b{_}5?ie=UTF8{\&}refRID=0M0FRPGY69WQH67H23TP.

[87] M. Laba, R. Downs, S. Smith, S. Welsh, C. Neider, S. White, M. Richmond, W. Philpot, and P. Baveye. "Mapping invasive wetland plants in the Hudson River National Estuarine Research Reserve using quickbird satellite imagery". In: *Remote Sensing of Environment* 112.1 (2008), pp. 286–300. ISSN: 00344257. DOI: 10.1016/j.rse.2007.05.003.

[88] R. M. Lark. "Components of accuracy of maps with special reference to discriminant analysis on remote sensor data". In: *International Journal of Remote Sensing* 16.8 (1995), pp. 1461–1480. ISSN: 0143-1161. DOI: 10.1080/01431169508954488.

[89] N. Lavesson and P. Davidsson. "Quantifying the Impact of Learning Algorithm Parameter Tuning". In: *Proceedings of the 21st National Conference on Artificial Intelligence - Volume 1*. AAAI'06. Boston, Massachusetts: AAAI Press, 2006, pp. 395–400. ISBN: 978-1-57735-281-5. URL: http://dl.acm.org/citation.cfm?id=1597538.1597602.

[90] T. M. Lee and H. C. Yeh. "Applying remote sensing techniques to monitor shifting wetland vegetation: A case study of Danshui River estuary mangrove communities, Taiwan". In: *Ecological Engineering* 35.4 (2009), pp. 487–496. ISSN: 09258574. DOI: 10.1016/j.ecoleng.2008.01.007.

[91] H. Li, F. Qi, and S. Wang. "A Comparison of Model Selection Methods for Multi-class Support Vector Machines". In: *Computational Science and Its Applications – ICCSA 2005: International Conference, Singapore, May 9-12, 2005, Proceedings, Part IV*. Ed. by O. Gervasi, M. L. Gavrilova, V. Kumar, A. Laganá, H. P. Lee, Y. Mun, D. Taniar, and C. J. K. Tan. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 1140–1148. ISBN: 978-3-540-32309-9. DOI: 10.1007/11424925_119. URL: http://dx.doi.org/10.1007/11424925_119.

[92] P. Li and H. Xu. "Land-cover change detection using one-class support vector machine". In: *Photogrammetry Engineering and Remote Sensing* 76 (2010), 255–263.

[93] W. Li and Q. Guo. "A maximum entropy approach to one-class classification of remote sensing". In: *International Journal of Remote Sensing* 31 (2010), 2227–2235.

[94] W. Li, Q. Guo, and C. Elkan. "A positive and unlabeled learning algorithm for one-class classification of remote-sensing data". In: *IEEE Transactions on Geoscience and Remote Sensing* 49.2 (2011), pp. 717–725. ISSN: 01962892. DOI: 10.1109/TGRS.2010.2058578.

[95] Y. Li, J. Kwok, and Z. Zhou. "Cost-Sensitive Semi-Supervised Support Vector Machine." In: *Proceedings of the 24th AAAI*. 2010, pp. 500–505. ISBN: 9781577354642.

[96] B. Liu, Y. Dai, X. Li, W. Lee, and P. Yu. "Building text classifiers using positive and unlabeled examples". In: *Third IEEE International Conference on Data Mining*. 2003. ISBN: 0-7695-1978-4. DOI: 10.1109/ICDM.2003.1250918.

[97] S. Liu, C. Jia, and H. Ma. "A new weighted support vector machine with GA-based parameter selection". In: *Machine Learning and Cybernetics, 2005. ...* August (2005), pp. 18–21.

[98] V. Lopez, A. Fernandez, J. G. Moreno-Torres, and F. Herrera. "Analysis of preprocessing vs. cost-sensitive learning for imbalanced classification. Open problems on intrinsic data characteristics". In: *Expert Systems with Applications* 39.7 (2012), pp. 6585–6608. ISSN: 09574174. DOI: 10.1016/j.eswa.2011.12.043.

[99] F. Löw, E. Fliemann, I. Abdullaev, C. Conrad, and J. P. Lamers. "Mapping abandoned agricultural land in Kyzyl-Orda, Kazakhstan using satellite remote sensing". In: *Applied Geography* 62 (2015), pp. 377–390. ISSN: 01436228. DOI: 10.1016/j.apgeog.2015.05.009.

[100] D. Lu and Q. Weng. "A survey of image classification methods and techniques for improving classification performance". In: *International Journal of Remote Sensing* 28.5 (2007), pp. 823–870. ISSN: 0143-1161. DOI: 10.1080/01431160600746456. URL: http://www.tandfonline.com/doi/abs/10.1080/01431160600746456.

[101] B. Mack, R. Roscher, and B. Waske. "Can i trust my one-class classification?" In: *Remote Sensing* 6.9 (2014), pp. 8779–8802. ISSN: 20724292. DOI: 10.3390/rs6098779.

[102] M. G. Mantero P. and S. B. Serpico. "Supervised classification of remote sensing images with unknown classes". In: *Proc. SPIE 5238, Image and Signal Processing for Remote Sensing IX, 386 (February 5, 2004)* (2004).

[103] M. Marconcini, D. Fernandez-Prieto, and T. Buchholz. "Targeted Land-Cover Classification". In: *IEEE Transactions on Geoscience and Remote Sensing* 52.7 (2014), pp. 4173–4193. ISSN: 0196-2892. DOI: 10.1109/TGRS.2013.2280150.

[104] J. G. Masek, D. J. Hayes, M. Joseph Hughes, S. P. Healey, and D. P. Turner. "The role of remote sensing in process-scaling studies of managed forest ecosystems". In: *Forest Ecology and Management* 355 (2015), pp. 109–123. ISSN: 03781127. DOI: 10.1016/j.foreco.2015.05.032.

[105] A. Mellor, S. Boukir, A. Haywood, and S. Jones. "Exploring issues of training data imbalance and mislabelling on random forest performance for large area land cover classification using the ensemble margin". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 105 (2015), pp. 155–168. ISSN: 09242716. DOI: 10.1016/j.isprsjprs.2015.03.014.

[106] W. Mitsch and J. Gosselink. *Wetlands*. Wiley, 2015. ISBN: 978-1-118-67682-0.

[107] G. Mountrakis, J. Im, and C. Ogole. "Support vector machines in remote sensing: A review". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 66.3 (2011), pp. 247–259. ISSN: 09242716. DOI: 10.1016/j.isprsjprs.2010.11.001.

[108] B. L. Munoz-Mari Jordi and G. Camps-Valls. "A support vector domain description approach to supervised classification of remote sensing images". In: *IEEE Transactions on Geoscience and Remote Sensing* 45.8 (2007), pp. 2683–2692. ISSN: 01962892. DOI: 10.1109/TGRS.2007.897425.

[109] F. G.-C. L. B. L. Munoz-Mari J. Bovolo and G. Camps-Valls. "Semisupervised One-Class Support Vector Machines for Classification of Remote Sensing Data". In: *IEEE Trans. Geosci. Remote Sens.* 48.8 (2010), pp. 3188–3197.

[110] G. H. Nguyen, S. L. Phung, and A. Bouzerdoum. "Efficient SVM training with reduced weighted samples". In: *Proceedings of the International Joint Conference on Neural Networks* (2010), pp. 1764–1768. ISSN: 1098-7576. DOI: 10.1109/IJCNN.2010.5596745.

[111] P. Olofsson, G. M. Foody, M. Herold, S. V. Stehman, C. E. Woodcock, and M. A. Wulder. "Good practices for estimating area and assessing accuracy of land change". In: *Remote Sensing of Environment* 148 (2014), pp. 42–57. ISSN: 00344257. DOI: 10.1016/j.rse.2014.02.015.

[112] M. Pal. "Factors influencing the accuracy of remote sensing classifications: a comparative study". PhD thesis. 2010, p. 278.

[113] M. Pal and G. M. Foody. "Feature selection for classification of hyperspectral data by SVM". In: *IEEE Transactions on Geoscience and Remote Sensing* 48.5 (2010), pp. 2297–2307. ISSN: 01962892. DOI: 10.1109/TGRS.2009.2039484. arXiv: arXiv:1011.1669v3.

[114] M. Pal and G. M. Foody. "Evaluation of SVM , RVM and SMLR for Accurate Image Classification With Limited Ground Data". In: 5.5 (2012), pp. 1344–1355.

[115] S. L. Phung, A. Bouzerdoum, and G. H. Nguyen. "Learning pattern classification tasks with imbalanced data sets". In: *Pattern Recognition* (2009), pp. 193–208.

[116] J. C. Platt. "Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods". In: *ADVANCES IN LARGE MARGIN CLASSIFIERS*. MIT Press, 1999, pp. 61–74.

[117] J. C. Platt, N. Cristianini, and J. S. Taylor. "Large Margin DAGs for Multiclass Classification". In: *Advances in Neural Information Processing Systems*. Ed. by S. A. Solla, T. K. Leen, and K. R. Mueller. 2000, pp. 547–553. URL: citeseer.ist.psu.edu/platt00large.html.

[118] K. Popper. *Conjectures and Refutations: The Growth of Scientific Knowledge*. second. Routledge, 2002, p. 608.

[119] A. V. Prishchepov, V. C. Radeloff, M. Dubinin, and C. Alcantara. "The effect of Landsat ETM/ETM+ image acquisition dates on the detection of agricultural land abandonment in Eastern Europe". In: *Remote Sensing of Environment* 126 (2012), pp. 195–209. ISSN: 00344257. DOI: 10.1016/j.rse.2012.08.017.

[120] X. Qiao and L. Zhang. *Distance-weighted Support Vector Machine*. 2013. arXiv: 1310.3003.

[121] M. M. Rahman and D. N. Davis. "Transactions on Engineering Technologies: Special Volume of the World Congress on Engineering 2013". In: ed. by G.-C. Yang, S.-I. Ao, and L. Gelman. Dordrecht: Springer Netherlands, 2014. Chap. Semi Supervised Under-Sampling: A Solution to the Class Imbalance Problem for Classification and Feature Selection, pp. 611–625. ISBN: 978-94-017-8832-8. DOI: 10.1007/978-94-017-8832-8_44.

[122] J. a. Richards and X. Jia. *Remote Sensing Digital Image Analysis: An Introduction*. 2006, p. 439. ISBN: 3540251286. DOI: 10.1007/978-3-642-30062-2. URL: http://books.google.com/books?id=4PB5vhPBdJ4C.

[123] R. Rifkin and A. Klautau. "In defense of one-vs-all classification". In: *Journal of Machine Learning Research* 5 (2004), pp. 101–141. ISSN: 15324435.

[124] C. Sanchez-Hernandez, D. S. Boyd, and G. M. Foody. "Mapping specific habitats from remotely sensed imagery: Support vector machine and support vector data description based classification of coastal saltmarsh habitats". In: *Ecological Informatics* 2.2 (2007), pp. 83–88. ISSN: 15749541. DOI: 10.1016/j.ecoinf.2007.04.003.

[125] C. Sanchez-Hernandez, D. S. Boyd, and G. M. Foody. "One-class classification for mapping a specific land-cover class: SVDD classification of fenland". In: *IEEE Transactions on Geoscience and Remote Sensing* 45.4 (2007), pp. 1061–1072. ISSN: 01962892. DOI: 10.1109/TGRS.2006.890414.

[126] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, a. J. Smola, and R. C. Williamson. "Estimating the support of a high-dimensional distribution." In: *Neural computation* 13.7 (2001), pp. 1443–1471. ISSN: 0899-7667. DOI: 10.1162/089976601750264965.

[127] B. Schölkopf, A. J. Smola, R. C. Williamson, and P. L. Bartlett. "New support vector algorithms". In: *Neural computation* 12.5 (2000), pp. 1207–1245.

[128] B. Schölkopf, R. Williamson, A. Smola, J. Shawe-Taylor, and J. Platt. *Support Vector Method for Novelty Detection*. 2000.

[129] S. Shalev-Shwartz and S. Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. New York, NY, USA: Cambridge University Press, 2014. ISBN: 1107057132, 9781107057135.

[130] J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. New York, NY, USA: Cambridge University Press, 2004. ISBN: 0521813972.

[131] D. Sheeren, M. Fauvel, V. Josipovi, M. Lopes, and C. Planque. "Tree Species Classification in Temperate Forests Using Formosat-2 Satellite Image Time Series". In: (2016), pp. 1–29. DOI: 10.3390/rs8090734.

[132] C. Song, C. E. Woodcock, K. C. Seto, M. P. Lenney, and S. A. Macomber. "Classification and Change Detection Using Landsat TM Data : When and How to Correct Atmospheric Effects ?" In: *Remote Sensing of Environment* 75.00 (2001), pp. 230–244.

[133] X. F. Song, H. S. Cui, and Z. H. Guo. "Remote sensing of mangrove wetlands identification". In: *Procedia Environmental Sciences* 10.PART C (2011), pp. 2287–2293. ISSN: 18780296. DOI: 10.1016/j.proenv.2011.09.357.

[134] K. Sriphaew, H. Takamura, and M. Okumura. "Cool blog classification from positive and unlabeled examples". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Vol. 5476. Springer, 2009, pp. 62–73. ISBN: 3642013066. DOI: 10.1007/978-3-642-01307-2{_}9.

[135] Y. Tang, Y. Q. Zhang, and N. V. Chawla. "SVMs modeling for highly imbalanced classification". In: *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics* 39.1 (2009), pp. 281–288. ISSN: 10834419. DOI: 10.1109/TSMCB.2008.2002909.

[136] D. M. J. Tax. "One-class classification". PhD thesis. 2001, p. 202.

[137] D. M. J. Tax and R. P. W. Duin. "Combining One-Class Classifiers". In: *Lecture Notes in Computer Science* 1032 (2001), pp. 299–308. ISSN: 00219673. DOI: 10.1016/j.chroma.2003.11.077.

[138] S. Theodoridis and K. Koutroumbas. *Pattern Recognition*. Vol. 53. 9. 2009, p. 1000. ISBN: 9788578110796. DOI: 10.1017/CBO9781107415324.004. arXiv: arXiv:1011.1669v3.

[139] V. N. Vapnik. *The Nature of Statistical Learning Theory*. Springer, 1995.

[140] T. Vo, C. Kuenzer, and N. Oppelt. "How remote sensing supports mangrove ecosystem service valuation: A case study in Ca Mau province, Vietnam". In: *Ecosystem Services* 14.MAY (2015), pp. 67–75. ISSN: 22120416. DOI: 10.1016/j.ecoser.2015.04.007.

[141] G. M. Weiss. "Mining with Rarity: A Unifying Framework". In: *SIGKDD Explorations* 6.1 (2004), pp. 7–19. ISSN: 19310145. DOI: 10.1145/1007730.1007734.

[142] G. M. Weiss and F. Provost. "Learning when training data are costly: The effect of class distribution on tree induction". In: *Journal of Artificial Intelligence Research* 19 (2003), pp. 315–354. ISSN: 10769757. DOI: 10.1613/jair.1199. arXiv: 1106.4557.

[143] J. Weston and C. Watkins. *Multi-class Support Vector Machines*. 1998.

[144] S. J. Wright. *Optimization for Machine Learning*. Vol. 1. 2011, p. 494. ISBN: 9780262016469. DOI: 10.1007/SpringerReference_302149.

[145] P. Xanthopoulos and T. Razzaghi. "A weighted support vector machine method for control chart pattern recognition". In: *Computers & Industrial Engineering* 70.October (2014), pp. 134–149. ISSN: 03608352. DOI: 10.1016/j.cie.2014.01.014.

[146] X. Yang, Q. Song, and Y. Wang. "A weighted support vector machine for data classification". In: *International Journal of pattern recognition and artificial inteligence* 2.5 (2007), pp. 859–864. ISSN: 0218-0014. DOI: 10.1142/S0218001407005703.

[147] S. Zhang, S. Sadaoui, and M. Mouhoub. "An Empirical Analysis of Imbalanced Data Classification". In: *Computer and Information Science* 8.1 (2015), pp. 151–162.

[148] J. Zhou, S. Pan, Q Mao, and I. Tsang. "Multi-view Positive and Unlabeled Learning." In: *Jmlr*. 2012, pp. 555–570.