Università degli Studi di Salerno

Facoltà di Scienze Matematiche Fisiche e Naturali

Dottorato di Ricerca in Informatica

IX ciclo Nuova Serie

Discovering hidden structures in high dimensional data space

Loredana Murino

November 2010

Chairman	Supervisor
Prof.	Prof.
Margherita Napoli	Roberto Tagliaferri

Acknowledgments

I would like to express my sincerest thanks to my thesis supervisor, Professor Roberto Tagliaferri, for his great guidance, support, patience and understanding throughout the course of this thesis. I also wish to express my appreciation to Professor Giancarlo Raiconi who made many valuable suggestions and gave constructive advice.

I gratefully thank Dr. Umberto Amato, Dr. Claudia Angelini and Dr. Italia De Feis of the IAC (Istituto per le Applicazioni del Calcolo 'Mauro Picone') CNR of Naples for kindly offering me their invaluable expertise and comments in the development of my thesis.

I would also like to extend my gratitude to all my colleagues of the NeuRoNe Lab, DMI, University of Salerno.

Contents

1 Introduction **11**

1.1 Motivation 11

1.2 Contribution 13

1.3 Thesis organization 14

2 Classification methods **15**

2.1 Supervised learning and classification methods 15

2.2 Discriminant analysis 19

2.2.1 Probability density function 19

2.2.2 Transforms 21

2.2.3 Discriminant analysis methods 22

3 Statistical cloud detection from SEVIRI multispectral images **23**

3.1 Cloud detection 23

3.2 Data 28

3.3 Experiments 35

3.3.1 Analysis of class density functions 37

3.3.2 Experiment 1 41

3.3.3 Experiment 2 56

4 Clustering and Consensus Clustering: Background **59**

4.1 Clustering: an open problem 59

4.2 Introduction to Consensus Clustering 64

4.3 Meta Clustering: does exist a unique solution? 69

5 Least Squares Consensus Clustering Algorithm **72**

5.1 Least-Squares Consensus Clustering 72

5.2 Least-Squares Error 74

5.3 Similarity measure and hierarchical clustering 75

5.4 Algorithm 79

5.5 Automatic cutoff selection 82

5.6 Pairwise matrix visualization 84

6 Results and Analysis **88**

6.1 Experimental setup 88

6.1.1 Simulated data 88

6.1.2 Real data-set 98

6.2 Method robustness 100

7 Conclusions and Future Work **104**

List of Figures

2.1 The process of supervised machine learning. 16

3.1 RGB image of European area, taken by SEVIRI sensor on June 30, 2004 11:27 (UTC) (data-set A). 31

3.2 Probability density function (pdf) of reflectance/radiance corresponding to the 11 SEVIRI channels. Plots refer to the data-set A (daytime) and land pixels. 38

3.3 Probability density function (pdf) of reflectance/radiance corresponding to the 11 SEVIRI channels. Plots refer to the data-set A (daytime) and water pixels. 39

3.4 Probability density function (pdf) of the first 4 principal components of reflectance/radiance.Left: land pixels; right: water pixels. Plots refer to the data-set A (daytime). 39

3.5 Probability density function (pdf) of reflectance/radiance corresponding to the 11 SEVIRI channels. Plots refer to the data-set E (nighttime) and water pixels. 40

3.6 Probability density function (pdf) of reflectance/radiance corresponding to the 11 SEVIRI channels. Plots refer to the data-set E (nighttime) and land pixels. 40

3.7 Probability density function (pdf) of the first 4 principal components of reflectance/radiance. Left: land pixels; right: water pixels. Plots refer to the data-set E (nighttime). 41
3.8 Success percentage, S, of the considered classification methods, when the data-set A is considered both as a training and test data-set. Plot refers to pixels over land. 43
3.9 Success percentage, S, of the considered classification methods, when the data-set E is considered both as a training and test data-set. Plot refers to pixels over land. 45
3.10 Success percentage, S, of the considered classification methods, when the data-set A is considered both as a training and test data-set. Plot refers to pixels over water 46
3.11 Success percentage, S, of the considered classification methods, when the data-set E is considered both as a training and test data-set. Plot refers to pixels over water. 47
3.12 Cloud mask obtained by PCDA for the data-set A when the same data-set is used to train discriminant analysis. Black: unprocessed pixels; blue: clear pixels over water; green: clear pixels over land; white: cloudy pixels over land or water. 49
3.13 RGB image of European area, taken by SEVIRI sensor on July 15, 2004 10:27 (UTC) (data-set B). 56
3.14 Cloud mask obtained by QDA for the data-set B when the data-set A is used to train discriminant analysis. Black: unprocessed pixels; blue: clear pixels over water; green: clear pixels over land; white: cloudy pixels over land or water. 57

5.1 Dendrogram example. 79
5.2 Global Least-Squares Error E_{LS} versus the number of steps. . . . 81

5.3 Heat-map visualization example. 86

6.1 Synthetic data-sets composed, respectively, by a mixture of 5, 6 and 7 Gaussians in 2 dimensions. 90

6.2 Synthetic data set: a three-level hierarchical structure with 3, 6 and 12 clusters in 2 dimensions. 91

6.3 Dendrogram of the clustering solutions for the synthetic data-sets. Different colors indicate different groups of aggregated clusterings. 92

6.4 Pairwise matrix visualization for the group of clusterings $\Gamma_7 = \{\gamma_{28}, \gamma_{29}, \gamma_{31}, \gamma_{32}, \gamma_{33}, \gamma_{34}, \gamma_{35}, \gamma_{36}, \gamma_{38}\}$ obtained applying the algorithm to the second synthetic data-set. 93

6.5 Least Square Error of the clusterings set at each step of the Algorithm for the synthetic data-set with a multi-level hierarchical structure. 95

6.6 Dendrogram of the clustering solutions for the synthetic data-set with the multi-level hierarchical structure. 96

6.7 Pairwise matrix visualization for the groups of clusterings Γ_1,Γ_2,Γ_3 and Γ_4 (starting from the upper panel) obtained applying the algorithm to the synthetic data-set with the multi-level hierarchical structure. 97

6.8 Pairwise matrix visualization for the groups of clusterings Γ_5, Γ_6 and Γ_7 (upper, central and lower panel respectively) obtained applying the algorithm to the synthetic data-set with the multi-level hierarchical structure. 99

6.9 Dendrogram of the clustering solutions for the real data-set. Different colors indicate different groups of aggregated clusterings. . 100

6.10 Pairwise matrix visualization for the 5 groups of clusterings obtained applying the algorithm to the real data-set. Starting from the upper panel on the left, the plots are referred to the clusterings groups with γ_3, γ_7, γ_4, γ_1 and γ_9 as consensus clustering respectively. 101

6.11 Upper panel: histogram of the number L of solutions obtained from the first set of experiments (K-means algorithm with fixed value of k); Central panel: Histogram of the number L of solutions obtained from the second set of experiments (K-means algorithm with variable values of k); Lower panel: Histogram of the number L of solutions obtained from the third set of experiments (EM algorithm with fixed value of k). 103

List of Tables

3.1 SEVIRI spectral characteristics. 29

3.2 SEVIRI data-sets. 30

3.3 MODIS data-sets. 33

3.4 Number of training data. 34

3.5 Success percentage, S, obtained by QDA and NPDA separately over land and water for data-sets A (daytime) and E (nighttime) when only one single spectral band is used for classification . . . 42

3.6 Bands chosen for the classification of the data-set A (daytime) over land pixels 48

3.7 Bands chosen for the classification of the data-set A (daytime) over water pixels. 49

3.8 Bands chosen for the classification of the data-set E (nighttime) over land pixels. 50

3.9 Bands chosen for the classification of the data-set E (nighttime) over water pixels. 50

3.10 Error indicators of LDA, QDA, NPDA and PCDA methods for classification on land pixels (Experiment 1). 51

3.11 Error indicators of LDA, QDA, NPDA and PCDA methods for classification on water pixels (Experiment 1). 52

3.12 Error indicators of LDA, QDA, NPDA and PCDA methods for classification of daytime data-sets on land pixels (Experiment 2). 53
3.13 Error indicators of LDA, QDA, NPDA and PCDA methods for classification of daytime data-sets on water pixels (Experiment 2). 54
3.14 Error indicators of LDA, QDA, NPDA and PCDA methods for classification of nighttime data-sets on land and water pixels (Experiment 2). 55

6.1 Groups of clustering solutions obtained by the dendrogram cut for the second synthetic data-set. 93
6.2 Groups of clustering solutions obtained by the dendrogram cut for the synthetic data-set with the multi-level hierarchical structure. 95

Chapter 1

Introduction

1.1 Motivation

Advances in sensing and storage technology have created many high-volume and high-dimensional data sets. The generation of multi-dimensional data has proceeded at an explosive rate in many disciplines: bioinformatics, finance, e-commerce, internet applications, geology, satellite detection and hyperspectral imaging are only few examples of this trend. Most of the data is stored digitally in electronic media, thus providing huge potential for the development of automatic data analysis.

The increase in both the volume and the variety of data requires advances in data mining which is the task of discovering interesting patterns from large amounts of data which can be stored in databases, data warehouses, or other information repositories. Data mining involves an integration of techniques from multiple disciplines such as database technology, statistics, machine learning, high-performance computing, pattern recognition, neural networks, data visualization, information retrieval, etc.

Data mining techniques can be broadly classified into two types [Tukey, 1977]: (i) exploratory or descriptive, meaning that the investigator does not have pre-specified models or hypotheses but wants to understand the general characteristics or structure of the data, and (ii) confirmatory or inferential, meaning that the investigator wants to confirm the validity of a hypothesis/model or a set of assumptions given the available data.

Machine learning provides the technical basis of data mining by extracting information from the raw data in the databases. The process usually consists of the following: transforming the data to a suitable format, cleaning it, and inferring or making conclusions regarding the data. Machine learning is divided into two primary sub-fields: supervised learning (classification) and unsupervised learning (clustering), the first involving only labeled data (training patterns with known category labels) while the latter involving only unlabeled data [Duda et al., 2001].

In traditional methodology, all these techniques assume many observations and a few, well chosen variables. The trend today is towards more observations but also to larger numbers of variables. There are a lot of examples where the observations gathered on individual instances are curves, or spectra, or images, or even movies, so that a single observation has dimensions in the thousands or billions, while there are only tens or hundreds of instances available for study. In a gene expression microarray data set, for instance, there could be tens or hundreds of dimensions, each of which corresponds to an experimental condition. As the classical methods are simply not designed to cope with this kind of explosive growth of dimensionality of the observation vector, new methods of high-dimensional data analysis could be developed [Donoho, 2000].

1.2 Contribution

The main purpose of this work of thesis is to find the most reasonable solutions for two data mining problems related to the management of high dimensional data. As mentioned in Section 1.1, the large volume of data that is currently collected in various fields of application can not be managed using data mining standard techniques: each technique is able to explore the solution space in a different way and it is often sensible to initial conditions. This thesis wants to emphasize the need to take a step forward in order to address the problems which arise from time to time and to use the correct data mining method for the problem at hand.

In particular two main applications of mining high dimensional data are considered in this work. The first one deals with cloud detection, a problem of multispectral satellite image classification, demonstrating the high reliability of the statistical techniques of discriminant analysis in classifying this type of images. Such classification technique has been compared with standard ones based on physical principles in order to benchmark the processing costs and the pass/fail rate [Amato et al., 2008]. The second application addresses the need to handle high dimensional data for which it is necessary to make assumptions rather than to have a confirmation (as in the previous application) . This naturally leads to the problem of clustering the data allowing to find significant structures within them. Instead of dwelling on one or more particular techniques of clustering, we chose to address the problem in a more comprehensive way by the so-called consensus clustering: rather than seek a single solution to the problem, the goal is to find all possible equivalently valid solutions. To this purpose an automatic procedure based on Least Squares Consensus Clustering has been developed.

The applications have been tested using both synthetic and real data-sets,

actually demonstrating the validity of the procedures. Strong emphasis has also been put on results validation through the use of "goodness" indicators in order to demonstrate the reliability of the techniques developed.

1.3 Thesis organization

The rest of the thesis is organized as follows: An overview of supervised learning and classification methods is presented in Chapter 2 with a particular reference to discriminant analysis techniques. Chapter 3 discusses in detail the methodology of cloud detection. Also the pre-processed data, the experimental results and analysis are provided at the end of this chapter. Chapter 4 contains an overview of clustering and consensus clustering techniques. The Least Squares Consensus clustering in presented in Chapter 5. Finally experimental results and analysis are provided in Chapter 6, followed by conclusions and future study in Chapter 7.

Chapter 2

Classification methods

2.1 Supervised learning and classification methods

Supervised machine learning is the process of learning a set of rules from instances (examples in a training set), or more generally speaking, creating a classifier that can be used to generalize from new instances. The process of applying supervised machine learning to a real-world problem is described in Figure 2.1 [Kotsiantis, 2007].

The first step is collecting the data-set. If a requisite expert is available, then he could suggest which fields (attributes, features) are the most informative. If not, then the simplest method is that of "brute-force," which means measuring everything available in the hope that the right (informative, relevant) features can be isolated. The second step is the data preparation and data preprocessing that cope with missing/noisy data and the infeasibility of learning from very large data-sets. During preprocessing, feature subset selection is used for identifying and removing as many irrelevant and redundant features as possi-

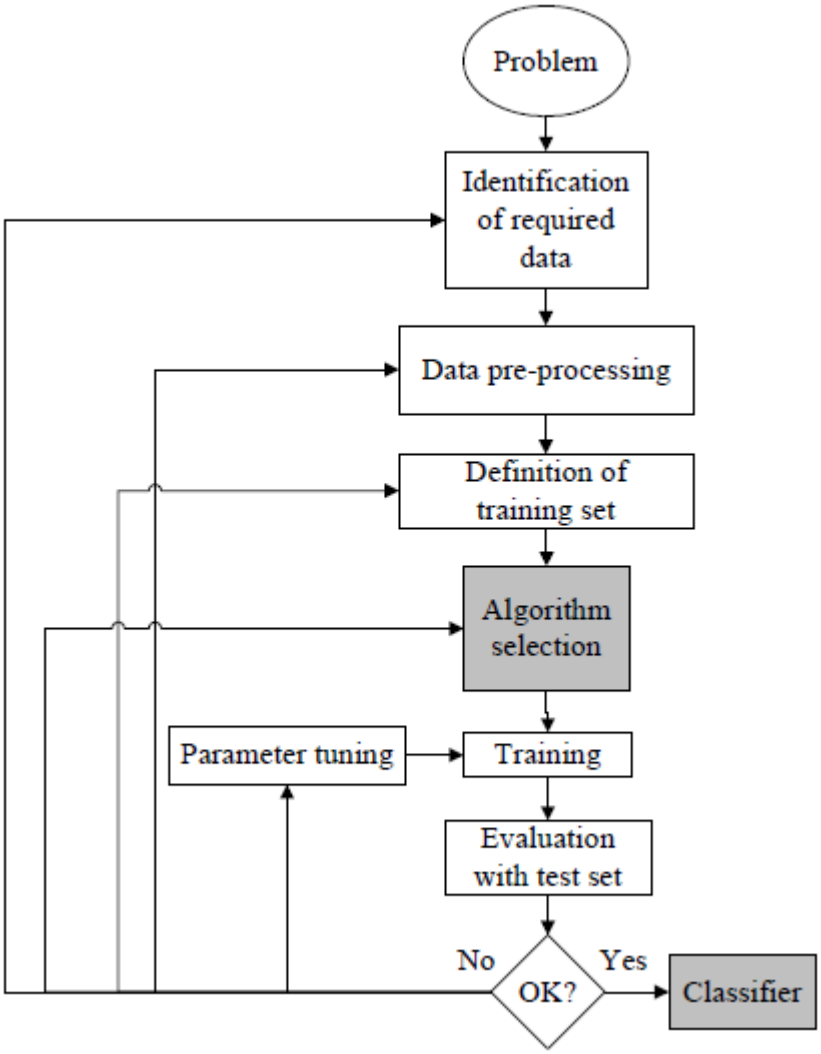

Figure 2.1: The process of supervised machine learning.

ble in order to reduce the dimensionality of the data and enable data mining algorithms to operate faster and more effectively. The fact that many features depend on one another often influences the accuracy of supervised classification models. This problem can be addressed by constructing new features from the basic feature set [Markovitch and Rosenstein, 2002]. This technique is called feature construction/transformation. These newly generated features may lead to the creation of more concise and accurate classifiers. In addition, the discovery of meaningful features contributes to better comprehensibility of the produced classifier, and a better understanding of the learned concept.

Algorithm selection is a critical step and depends on the data collection and preprocessing steps. A common method for comparing and choosing the supervised algorithms is to perform statistical comparisons of the accuracies of trained classifiers on specific data-sets. Classification algorithms aim at assigning a class label for each input example. Given a training data set of the form (x_i, y_i), where $x_i \in \mathbb{R}^n$ is the ith example and $y_i \in \{1, \ldots, K\}$ is the ith class label, we aim at finding a learning model H such that $H(x_i) = y_i$ for new unseen examples. The classification problem is simply formulated in the two class case, where the labels y_i are just $+1$ or -1 for the two classes involved. Several algorithms have been proposed to solve this problem in the two class case, some of which can be naturally extended to the multiclass case, and some that need special formulations to be able to solve the latter case.

The multiclass classification problem can be solved by naturally extending the binary classification technique for some algorithms [Aly, 2005]. These include:

1. Neural Networks and, in particular, Multilayer Feed forward Neural Networks provide a natural extension to the multiclass problem. Instead of just having one neuron in the output layer, with binary output, one could

have N binary neurons. Weightings are applied to the signals passing from one neuron to another, and it is these weightings which are tuned in the training phase to adapt a neural network to the particular classification at hand.

2. Decision Trees [Breiman et al., 1984] try to infer a split of the training data based on the values of the available features to produce a good generalization. The split at each node is based on the feature that gives the maximum information gain. Each leaf node corresponds to a class label.

3. k-Nearest Neighbors [Bay, 1998] is considered among the oldest non parametric classification algorithms. To classify an unknown example, the distance (using some distance measure e.g. Euclidean) from that example to every other training example is measured. The k smallest distances are identified, and the most represented class in these k classes is considered the output class label. The value of k is normally determined using a validation set or using cross-validation.

4. Naive Bayes [Rish, 2001] is a successful classifier based upon the principle of Maximum A Posteriori (MAP).

5. Support Vector Machines [Cortes and Vapnik, 1995] are based upon the idea of maximizing the margin i.e. maximizing the minimum distance from the separating hyperplane to the nearest example. The basic SVM supports only binary classification, but extensions have been proposed to handle the multiclass classification case as well. In these extensions, additional parameters and constraints are added to the optimization problem to handle the separation of the different classes.

2.2 Discriminant analysis

In the present work we have used discriminant analysis methods for the problem of images classification. The purpose of any discriminant analysis method is to assign a p-variate observation $\mathbf{x}$ (pixel) to one class from a set of K classes with the lowest possible error rate. In the standard setting, observations are described by multivariate random vectors coming from a certain class $k(k = 1 \dots K)$ characterized by a density function $f_k(\mathbf{x})$. An observation is decided to be drawn from one and only one class (Bayes rule) and error is incurred if it is assigned to a wrong one. The cost or loss associated with such an error is usually defined by $L(k, \tilde{k})$, where k is the correct class assignment and $\tilde{k}$ is the assignment that was actually made. A special but commonly occurring loss L is the $0-1$ loss defined by

$$L(k, \tilde{k}) = \begin{cases} 0, & if\ k = \tilde{k} \\ 1, & otherwise \end{cases} \tag{2.1}$$

In this case the Bayes decision rule actually used in the algorithm allocates $\mathbf{x}$ to the class $\tilde{k}$ such that $f_k(\mathbf{x})\pi_k$ is maximum, where $f_k(\mathbf{x})$ are k-class conditional density functions and π_k are unconditional class prior probabilities, assumed uniform in the present work. Discriminant analysis requires a training data-set that can be considered as a sample of feature vectors from each class used to learn the density functions of the classes.

2.2.1 Probability density function

The most popular classification rules are based on the normal theory, which assumes that the densities f_k are Gaussian. Such standard parametric rules include linear and quadratic discriminant analysis (e.g., [Anderson, 1984]), which have been shown to be quite useful in a wide variety of problems. However, in

practice, the form of the class-conditional densities is seldom known and hardly meets the hypothesis of gaussianity. Therefore a careful analysis of the density functions of the values for different samples is very important for the correct application of the discriminant analysis. First of all, one can evaluate the overall feasibility of the discriminant analysis in classifying samples: the more density functions are split away for different classes, the more the discriminant analysis will be able to classify samples correctly. If density functions are recognized as belonging to classic known types, simple parametric discriminant analysis methods can be applied; in the opposite case discriminant analysis naturally extends to the situation where nothing is known about the densities f_k except possibly for some assumptions about their general behavior. The suggested approach is to estimate the densities f_k from a training set using nonparametric density estimates and to substitute these estimates into the Bayes decision rule to give a nonparametric discriminant rule. The most popular procedure for nonparametric density estimation is kernel density estimation with appropriate smoothing parameter selection [Wand and Jones, 1995]:

$$f_k(\mathbf{x}) = \frac{1}{N_k} \sum_{\ell=1}^{N_k} H(\mathbf{x} - \mathbf{z}_\ell), \tag{2.2}$$

where N_k is the size of the training set for class k and $\mathbf{z} \equiv (\mathbf{z}_\ell)_{\ell=1,\ldots,N_k} \equiv (z_\ell^{(1)}, \ldots, z_\ell^{(p)})$. A popular choice of the kernel function H is the Gaussian one. Most often the p-variate density function is taken as the product of univariate functions as [Wand and Jones, 1995]:

$$H(\mathbf{x}) = \frac{1}{(2\pi)^{p/2}\sigma_1 \cdots \sigma_p} \prod_{j=1}^{p} \exp\left(-\frac{\left(x^{(j)}\right)^2}{2\sigma_j^2}\right), \tag{2.3}$$

where [Wand and Jones, 1995]

$$\sigma_j = 1.09\hat{\sigma}_j N_k^{-0.2} \tag{2.4}$$

and $\hat{\sigma}_j^2$ is an estimate of the variance of the multispectral component j for the (dropped) class k. However, it is known that while in one-dimensional density estimation it is not crucial to estimate the tails accurately, this is no longer true in high dimensional spaces where regions of relatively low density can still be extremely important parts of the multidimensional density.

For the purpose of the present analysis unidimensional Gaussian kernel density estimation and corresponding bandwidth were considered (Eqs. (2.2)-(2.4)).

2.2.2 Transforms

To estimate a density function as the product of univariate functions requires an assumption of independence of data that does not always hold in practice. In the case of Gaussian distributions this can be fixed by transforming original multivariate data into principal components by Principal Component Analysis (PCA) and then proceeding with classification. For general distributions, application of PCA to the data only decorrelates them, without yielding full independence. A possible remedy is to seek for a transform that makes data mutually independent irrespective of their distribution. Independent Component Analysis (ICA) achieves such a task. It is a statistical method for linearly transforming an observed multidimensional random vector into a random vector whose components are stochastically as independent from each other as possible. Several procedures to find such transformations have been recently developed in the signal processing literature relying either on Comon's information-theoretic approach [Comon, 1984] or Hyvärinen's maximum negentropy approach [Hyvärinen, 1997]. The present work considers this latter approach and relies on the Matlab package fastica [Hyvärinen, 1999], available at

the Website http://www.cis.hut.fi/projects/ica/fastica/, for its implementation. fastica looks for independent components maximizing an index of nongaussianity through some suitable contrast functions ($u^2, u^3, tanh(u), u\exp(-u^2/2)$, with u being seeked independent component). Details of the method (ICDA) are reported in [Amato et al., 2003].

2.2.3 Discriminant analysis methods

Three nonparametric and two parametric discriminant analysis methods for multispectral cloud classification have been considered:

1. LDA (Linear Discriminant Analysis), based on Gaussian density functions with common variance among classes;

2. QDA (Quadratic Discriminant Analysis), based on Gaussian density functions with general covariance of the multispectral radiance/reflectance for each class;

3. NPDA (NonParametric Discriminant Analysis), where a nonparametric estimate of the density functions is made for each component separately;

4. PCDA (Principal Component Discriminant Analysis [Amato et al., 2003]), where original components are transformed into principal components prior to nonparametric density estimation;

5. ICDA (Independent Component Discriminant Analysis [Amato et al., 2003]), where original components are transformed into independent components prior to nonparametric density estimation. To this purpose the fastica package was resorted to estimate independent components by using the contrast function u^3.

Chapter 3

Statistical cloud detection from SEVIRI multispectral images

3.1 Cloud detection

Remote sensing is leading an increasingly significant contribution to environmental monitoring and Earth observation. A lot of techniques have been developed to analyze and extract information from remotely sensed data for several applications as agriculture, deforestation, pollution, earthquakes, fire detection, oceans monitoring and risk management. The first step in remotely sensed image analysis for all these applications is the detection of the areas which can be actually supervised: a significant portion of land surface is obscured by clouds and this fact complicates the observations of surface processes and phenomena from space. So cloud detection is a preliminary important operation in most

algorithms for processing radiance data measured from sensors on board satellites.

Clouds are generally characterized by higher reflectance and lower temperature than the underlying earth surface. However, there are many surface conditions when this characterization of clouds is inappropriate. Additionally, some cloud types such as thin cirrus, low stratus at night, and small cumulus are difficult to be detected because of insufficient contrast with the surface radiance. Many of these concerns can be mitigated by multispectral approaches to cloud detection and, for this reason, the availability of multispectral sensors, able to measure radiance emitted by Earth surface at several and narrow spectral bands, represents an important improvement in this field.

Several algorithms devoted to cloud detection are available for multispectral data. Most of them are based upon the spectral behavior of clouds both in the emissive infrared and reflective bands. Generally some decision rules are set involving a few selected spectral bands; then thresholds on the value of radiances are empirically chosen to discriminate between the cloudy and clear sky conditions. Methods based on decision rules underwent a significant evolution during recent years, even permitting to retrieve not only the presence of clouds but also several related features, e.g., tracking, shape [Yang et al., 2006, Yang et al., 2007].

Physical methodologies suffer from some drawbacks as high variability of clouds, dependence of radiance on the emissivity of the surface, which is very difficult to estimate accurately over land, and the choice of suitable bands for the decision rules. For this reason there was in the recent years interest towards classification methods that approach the problem of cloud detection through statistical methods: classification methods learn the statistical features of the cloudy and clear sky conditions "on-field", that is starting from "truth" images

where the sky conditions are "certainly" known; then sky conditions on other "new" images are inferred from these by relying on some of the statistical properties learned. The idea of statistical classification is based on the fact that the spectral signature of each pixel contains information on the physical characteristics of the land underlying the pixel and/or the clouds eventually present in the atmosphere above. Therefore from this information we can infer, e.g., the statistical properties of the type of land cover or cloud associated to that pixel. It is clear that the use of medium or high resolution spectral data (i.e., multivariate data in the statistical terminology) opens new perspectives to applications: actually, coverage of a wider fraction of the electromagnetic spectrum at a better spectral resolution means to represent better the spectral signature corresponding to each pixel and then to pick the unique spectral features of clouds better.

In recent years a lot of works have been published on this topic; we mention methodologies based on Support Vector Machines (SVM) applied to MODIS data [Han et al., 2006, Lee et al., 2004]. [Lee et al., 2004] give an excellent example of statistical methods applied directly to the physical methodology underlying an operative product of cloud detection. We also recall "neural-network classifiers" which include multilayer Back- Propagation Neural Network (BPNN), Self Organizing Map (SOM) [Stephanidis et al., 1995], Probability Neural Network (PNN) [Tian et al., 1999], etc. They need a training phase to learn cloud features from "truth" images. A BPNN classification system was tested on MSG-SEVIRI images using MODIS cloud mask product both in training and testing phase [Falcone and Azimi-Sadjadi, 2005]. Moreover, as a fixed neural network may not be able to deal with a sequence of images obtained at different times of the day, temporal adaptive neural network-based cloud classification systems have been developed [Tian et al., 2000]. METEOSAT images were considered

in [Macias-Macias et al., 2004]. Also fuzzy rule based approaches have been proposed to estimate cloud cover [Baum et al., 1997, Ghosh et al., 2003].

Another technique that was used in the cloud classification field is based on textural features analysis which consists in distinguishing clouds by the spatial distribution characteristics of gray levels corresponding to a region in one specific channel [Ameur et al., 2004]. While the spectral features of clouds may change, their textural properties are often distinct and tend to be less sensitive to the effects of atmospheric attenuation or detector noise. Most of the texture-based cloud classification methods in the past had used statistical measures based on Gray Level Co-occurrence Matrix (GLCM) and its variants, such as Gray Level Difference Vector (GLDV). Several comparative studies on textural features have been conducted [Gu et al., 1989], however there is no consistent and optimal feature extraction scheme determined at this time. From the statistical point of view full use of multivariate data has intrinsic issues both from the theoretical and the numerical point of view that have to be carefully investigated. Here we stress that multivariate analysis is subject to the so called well known "curse of dimensionality", that expressed in a statistical sense accounts for the degradation of estimation accuracy with a growing number of dimensions. From the practical point of view this means that accuracy is worse when the same number of data points is spread over more dimensions or, as a counterpart, that a much higher number of points is required to get the same accuracy as the unidimensional case. A raw way to face the curse of dimensionality is to neglect correlations among variates (i.e., spectral bands) and to deal with all variates separately. A better solution is to take full account of the link among the variates by considering properly their dependence or, at least, covariance structure. As we have seen in Section 2.2.2 this can be accomplished, e.g., through a Principal Component Analysis, where new variates are consid-

ered with respect to the original ones, that are their linear combinations claimed to be fully decorrelated. We claim that statistical classification via discriminant analysis has a strong connection with the physical consolidated methodologies based on thresholds and decision rules on radiances/reflectances. Actually unidimensional discriminant analysis is the statistical counterpart of the thresholds on reflectances/radiances at single wavelengths; in the bidimensional case, discriminant analysis is the counterpart of physical decision rules involving couples of spectral bands, since it defines regions in the radiance/reflectance plane where pixels are classified as clear or cloudy. In addition we observe that these regions can have shapes with a growing generality (semi-planes in the case of Linear Discriminant Analysis, paraboloids for Quadratic Discriminant Analysis, and so on) according to the assumptions made by discriminant analysis; the key point is that these shapes include and are more general than the ones of the physical classification methods. When we increase the number of dimensions (i.e., spectral bands) the regions of the hyperplanes that discriminate between clear and cloudy pixels are even more general and complicated and there is no counterpart of any decision rule developed physically. In this work the link between statistical and physical classification is made definitely strong by a full plug-in of some physical classification methodology into discriminant analysis. This is accomplished in the training phase of discriminant analysis, where statistical properties of the cloudy and clear sky conditions have to be learned: a cloud mask produced by another sensor is used to this purpose. Of course this choice poses new questions to the cloud detection through discriminant analysis concerning reliability of the cloud mask product used for training and consequently accuracy of the produced final cloud mask.

We mention in advance that the product used for the training phase is MOD35 available from NOAA starting from MODIS sensor on board EOS se-

ries satellites [Salomonson et al., 1998]; MOD35 is claimed to yield an accurate cloud mask. We also point out that a cloud mask obtained directly from SEVIRI sensor is available from EUMETSAT [Derrien and Le Gleau, 2005]. We stress that only pixels classified as clear or cloudy with a good confidence are used in the training phase, which increases robustness of the method; in addition, provided that the number of erroneous training pixels is small, discriminant analysis is even able to correct such occurrences. The physical/statistical methodology proposed in the present work can reveal useful with new sensors, because it represents a not expensive and fast method to produce cloud masks during the commissioning phase immediately following the launch of the sensor.

3.2 Data

Meteosat Second Generation (MSG) is a significantly enhanced follow-on system to the previous generation of Meteosat. MSG consists of a series of four geostationary meteorological satellites, along with ground-based infrastructure, that will operate consecutively until 2018. The first MSG satellite to be launched was Meteosat-8, in 2002. The second satellite followed up in December 2005. MSG has been designed in response to user requirements and serves the needs of nowcasting applications and numerical weather prediction in addition to provision of important data for climate monitoring and research. The MSG system has brought major improvements in these services through its radiometer, the Spinning Enhanced Visible and InfraRed Imager (SEVIRI). SEVIRI is a 50 cm diameter aperture, line by line scanning radiometer which provides image data in four Visible and Near InfraRed (VNIR) channels and eight InfraRed (IR) channels. The VNIR channels include also a High Resolution Visible (HRV) channel to scan the Earth with a 1 km sampling distance at sub satellite point. All the other channels (including the IR channels) are designed to scan the Earth with

a 3 km sampling distance. The imaging is performed by combining satellite spin and rotation (stepping) of the scan mirror. The images are taken from South to North and East to West. The E-W scan is achieved through the rotation of the satellite with a nominal spin rate of 100 rpm. Spectral characteristics of SEVIRI are shown in Table 3.1.

Channel ID	Channel Type	Wavelengths (μm)		
		Central	Minimum	Maximum
VIS 0.6	VNIR	0.635	0.56	0.71
VIS 0.8	VNIR	0.81	0.74	0.88
IR 1.6	VNIR	1.64	1.50	1.78
IR 3.9	IR	3.92	3.48	4.36
IR 6.2	Water vapour	6.25	5.35	7.15
IR 7.3	Water vapour	7.35	6.85	7.85
IR 8.7	IR	8.70	8.30	9.10
IR 9.7	IR	9.66	9.38	9.94
IR 10.8	IR	10.80	9.80	11.80
IR 12.0	IR	12.00	11.00	13.00
IR 13.4	IR	13.40	12.40	14.40
HRV	Visible	Broadband (0.4-1.1)		

Table 3.1: SEVIRI spectral characteristics.

SEVIRI data are available at the EUMETSAT on-line archive, website http://archive.eumetsat.org/en/index.html.

Data, distributed in Level 1:5 BSQ format, are calibrated. Their format is described in [Damman and Mueller, 2006].

Five SEVIRI data-sets were used in our study: three of them were taken on daytime and the other ones on nighttime. Table 3.2 shows these data-sets and the corresponding data and time (in UTC format) of the SEVIRI acquisition; EUMETSAT (European Organization for the Exploitation of Meteorological Satellites) file names are also shown; a literal identification (ID) is assigned to each set of data for the purposes of the present work.

A geographic area extending on Europe, from Iberian Peninsula to Italy, was selected from the full disk. Figure 3.1 shows the RGB image of this area

ID	File name	Date	Time (UTC)
A	MSG15-0100-NA-20040630111237.433000000Z-132926	June 30, 2004	11:12
B	MSG15-0100-NA-20040715102736.486000000Z-136066	July 15, 2004	10:27
C	MSG15-0100-NA-20040815112737.020000000Z-135782	August 15, 2004	11:27
D	MSG15-0100-NA-20040706214237.472000000Z-139840	July 6, 2004	21:42
E	MSG15-0100-NA-20040818212737.141000000Z-133532	August 18, 2004	21:27

Table 3.2: SEVIRI data-sets.

obtained starting from SEVIRI channels at 0.635 μm, 0.81 μm and 1.64 μm for the June 30, 2004 data-set (ID=A).

As already mentioned in Section 2.1, classification methods require a training data-set from which statistical properties of the classes can be learned. To this purpose MODIS (Moderate Resolution Imaging Spectroradiometer) cloud mask was used. MODIS is the keystone instrument on board NASA EOS (Earth Observation System) Terra and Aqua satellites [Salomonson et al., 1998]. They view the entire Earth's surface every 1 to 2 days, acquiring data in 36 spectral bands from the short wave visible to the long wave infrared. A cloud mask obtained from MODIS radiance is available as product MOD35. It is a daily, global Level 2 product generated at 1 Km and 250 m (at nadir) spatial resolutions. MOD35 algorithm [Ackerman et al., 1998, Li et al., 2003, Platnick et al., 2003] identifies some conceptual domains according to some geographical parameters (surface type, illumination, daytime or nighttime). For each pixel belonging to a particular domain some tests try to infer contamination of clouds from the measured radiances/reflectances. Some tests involve single channels, others two channels through differences or ratios. Channels involved in the tests are a subset of the full channels of MODIS (14 out of 36). Suitable thresholds are defined for each test that discriminate between the status of cloudy or clear pixel. The answer of each test is not binary; rather a confidence indicator between 0 and 1 is yielded where 0 represents high confidence in cloudy conditions and 1 high confidence in clear conditions (intermediate values of course indicate

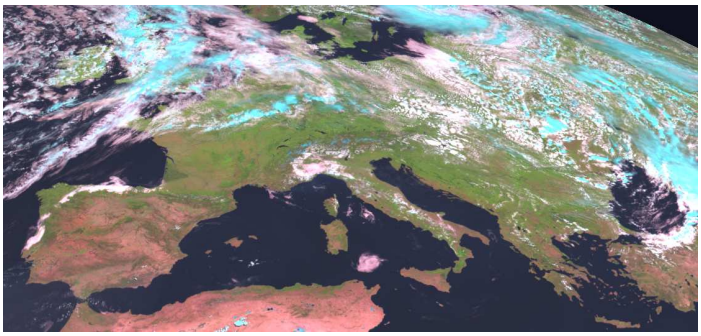

Figure 3.1: RGB image of European area, taken by SEVIRI sensor on June 30, 2004 11:27 (UTC) (data-set A).

less confidence about the two conditions). Actually, tests are combined in 5 groups aiming at detecting particular categories of clouds: Group I for thick high clouds; Group II detects thin clouds; Group III, relying on reflectance, is devoted to low clouds; Group IV is specialized for high thin clouds; finally Group V detects high thin cirrus. For each group a confidence indicator is defined as the smallest confidence indicator of the tests belonging to it.

Finally a cloud mask indicator, Q, is computed as the geometric mean of the confidence indicators of the 5 Groups. This approach is conservative in the estimation of clear sky: for example, if any of all tests is totally confident that the pixel is cloudy (confidence indicator equal to 0), then the pixel is classified as confidently cloudy. Actually, the cloud mask of MOD35 provides four levels of confidence according to the value assumed by Q: confidently clear ($Q > 0.99$), probably clear ($0.95 < Q \leq 0.99$), uncertain clear ($0.66 < Q \leq 0.95$), cloudy ($Q \leq 0.66$). MOD35 is claimed to yield a very robust cloud mask; its algorithm still undergoes changes to improve detection capability (see, e.g., [Liu et al., 2004], for nighttime polar region). Radiometrically accurate radiances are required, so holes in the cloud mask will appear wherever the input radiances are incomplete or of poor quality. As all official EOS data products, MOD35 is created in Hierarchical Data Format (HDF) and is available from the EDG (EOS Data Gateway) on-line catalog at the website http://edcimswww.cr.usgs.gov/pub/imswelcome/. Use of MODIS data as a training data-set for detecting arctic clouds has also been made in [Shi et al., 2007]. An estimate of the classification error for the MOD35 product is given in [Lee et al., 2004]: the value obtained on a purposely built data-set directly classified by a meteorologist is about 18%; this value is reported only for the whole training set (that is considering pixels classified probably clear or cloudy as confidently clear or cloudy, respectively), so that the misclassification

ID	File name	Date	Time (UTC)
A	MOD35-L2.A2004182.1110.004.2004182223742	June 30, 2004	11:10
B	MOD35-L2.A2004197.1025.004.2004197235042	July 15, 2004	10.25
C	MOD35-L2.A2004228.1120.004.2004256073242	August 15, 2004	11:20
D	MOD35-L2.A2004188.2140.004.2004189085034	July 6, 2004	21:40
E	MOD35-L2.A2004231.2120.004.2004258132428	August 18, 2004	21:20

Table 3.3: MODIS data-sets.

error for pixels confidently classified is somewhat smaller. [Shi et al., 2007] also give an estimate of the error affecting MOD35 cloud mask over arctic regions, based on a training data-set purposely defined by an expert (about 11%).

For each SEVIRI data-set one corresponding MOD35 product was selected. They are shown in Table 3.3, including date, acquisition time and corresponding file names. Of course SEVIRI and MODIS data-sets were chosen such that passage times over the analyzed zone were coinciding as much as possible. MOD35 products generated at 1 km spatial resolution were used.

As SEVIRI and MODIS sensors have different grids and, especially, spatial resolutions, MODIS data were resampled on the SEVIRI grid before using them to create the training data-sets.

To this purpose associated MOD03 product can be used to obtain geographical coordinates on the MODIS grid, avoiding the interpolation error of the reduced grid (5 Km) provided with MOD35. MOD03 product was not available with the scenes considered in this work, however no practical misalignment has been detected by interpolating the reduced resolution grid provided with MOD35 to the full 1 Km grid. In our experiments only pixels on the SEVIRI grid where all corresponding MODIS pixels were classified by MOD35 as confidently clear or confidently cloudy were selected to build the training data-sets. In Table 3.4 the number of training data for each MODIS data-set is shown separately for land and water.

Two classes are defined corresponding to “cloudy” and “clear sky” conditions

	ID	Total pixels	Confidently classified	Cloudy	Clear
Land	A	140.888	97.776 (69.4%)	36.416	61.360
	B	104.328	80.124 (76.8%)	46.925	33.199
	C	89.714	59.839 (66.7%)	27.028	32.811
	D	117.744	78.064 (66.3%)	49.163	28.901
	E	99.284	58.280 (58.7%)	22.065	36.215
Water	A	128.064	100.658 (78.6%)	47.049	53.609
	B	83.915	73.845 (88.0%)	26.980	46.865
	C	89.803	76.063 (84.7%)	51.334	24.729
	D	106.480	69.212 (65.0%)	66.335	2877
	E	82.399	34.937 (42.4%)	30.019	4918

Table 3.4: Number of training data.

and the classification is performed separately on land and water pixels. Of course classification on daytime involved all the 11 SEVIRI bands; for visible and near infrared channels (0.635 μm, 0.81 μm and 1.64 μm) a geometric conversion from radiance to reflectance was performed. For nighttime classification only the 8 infrared channels have been considered.

Discriminant analysis methods described in Section 2.2.3 are multivariate in the sense that the (multivariate) probability density functions of the populations are factorized into their (univariate) spectral components. In the case of LDA, QDA and NPDA this is only a raw approximation of the statistical properties of the population, whereas PCDA and ICDA provide a much better approximation. These methods could be applied to all the 11 spectral bands for daytime data-sets and all the 8 spectral bands for nighttime data-sets; however different bands have not the same quality for several reasons: instrumental arguments related with the lower signal to noise ratio of the infrared channels with respect to the visible ones; noise induced by atmosphere; ultimately, but most important, the information content is very different among the spectral bands. Furthermore there can be statistical reasons for selecting spectral bands to be used in the classification. These reasons are mainly related to the so called "curse of dimensionality" already discussed in Section 3.1. Indeed the method-

ologies described in the present work naturally circumvent this problem in that independence is guaranteed through proper transforms. Nevertheless, we have investigated a progressive inclusion of the bands for the classification analysis, with the main objective to assess the role of the bands for the classification in a quantitative way. In practice we have selected some bands and estimated their effectiveness in classifying data. Two different strategies were considered for selecting the bands:

- *simple forward*. The first band is selected by an exhaustive search over all N bands as the one that gives the best classification performance on the training data-set; the second band is selected by the same criterion among all the remaining N-1 bands; the other bands are chosen orderly with the same criterion by a recursive procedure.

- *forward-backward*. At each step of the procedure a check is made whether eliminating one of the already selected bands improves performance of the classification; this allows one to limit the bias eventually introduced by the forward recursive procedure. See also [Groves and Bajcsy, 2003] for an alternative band selection methodology based on a priori estimate of the information content of the bands.

3.3 Experiments

This section shows the experiments worked out on the data of Section 3.2. Two different numerical experiments have been conducted in order to evaluate accuracy of the classification methods and robustness with respect to the training data-set:

- *Experiment 1*: A data-set is defined starting from one of the MOD35 products of Table 3.3 for the training of the classification methods. Clas-

sification is performed on the data of the corresponding SEVIRI data-set (see Table 3.2) and performance indicators are evaluated for the training set (test data-set coinciding with the training data-set);

- *Experiment 2*: Again a data-set is defined starting from one of the MODIS data-sets of Table 3.3 for the training of the classification methods; however classification is performed on SEVIRI data-sets of different days (test data-set different from the training data-set).

From MOD35 product a water–land mask was extracted and each of the two experiments was performed on land pixels and water pixels separately. The results are shown in separate tables. In order to estimate the performance of the classification methods quantitatively, the following indicators have been considered:

- s_k (percentage of agreement), defined as the percentage of pixels belonging to the class k (cloud or clear sky) correctly classified;

- F_k^+(false positive rate of class k), defined as the percentage of pixels known to belong to the class different from k and erroneously classified as belonging to class k;

- F_k^-(false negative rate of class k), defined as the percentage of pixels known to belong to class k and erroneously classified as belonging to the other class;

- κ_k(kappa-statistic coefficient). It is the chance-corrected measure of agreement for each class, defined as $(s_k - p_c)/(1 - p_c)$, where s_k is the above mentioned observed percentage of agreement and p_c is the percentage agreement that would occur by chance alone; values of $\kappa > 0.7$ are claimed to indicate a good classification.

- S, global success percentage of correctly classified pixels for the whole set of data.
- κ (global kappa-statistic coefficient). It is the chance corrected measure of agreement for the whole set of data.

Numerical values of these indicators have been obtained comparing the classification results with the corresponding MOD35 products. We recall that for the purpose of training set definition and performance evaluation only SEVIRI pixels composed of MODIS pixels all classified confidently (both clear and cloudy) are considered. This is due to the fact that assignment of pixels classified by MOD35 as probably clear or cloudy to their respective confident classes would almost surely (in a probabilistic sense) produce several misclassifications of the discriminant analysis in correspondence of the MOD35 pixels erroneously classified. We are aware that this procedure could bias positively the performance indicators obtained, since many pixels not included in the analysis are probably more difficult to be classified correctly. However pixels classified confidently by MOD35 are majority of all pixels in an extent that depends on the particular scene.

3.3.1 Analysis of class density functions

Analysis of density function of spectral radiance for the two considered classes ("cloud" and "clear sky") is an important step to apply correctly the discriminant analysis. If the density functions belong to known families then simpler and more efficient parametric classification methods could be resorted; in the opposite case discriminant analysis should be based on a nonparametric estimate of the density functions. Figure 3.2 (for land) and Figure 3.3 (for water) show the density function of reflectance or radiance (where relevant) of the considered classes as obtained by Kernel density estimation for the data-set A.

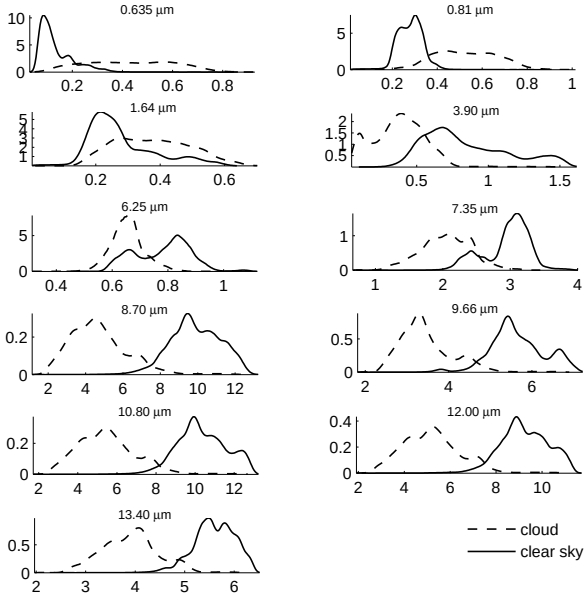

Figure 3.2: Probability density function (pdf) of reflectance/radiance corresponding to the 11 SEVIRI channels. Plots refer to the data-set A (daytime) and land pixels.

Figure 3.4 shows the density functions of the first 4 principal components in clear and cloudy sky conditions for the same data-set A (separately over land and over water). Of course principal components are now a mixing of the original reflectance/radiance at the wavelengths of the SEVIRI sensor. It is clear that density functions of radiance (or reflectance) hardly can be described well by Gaussians, while transform to principal components makes density functions unimodal and even more Gaussian-like. In addition overlap of the density functions is much lower in the latter case, which potentially improves the rate of classification.

Analysis conducted on the other daytime data-sets (not shown here for the sake of brevity) has shown similar results. Figures 3.6, 3.5 and 3.7 show the results of the same analysis conducted on the nighttime data-set E.

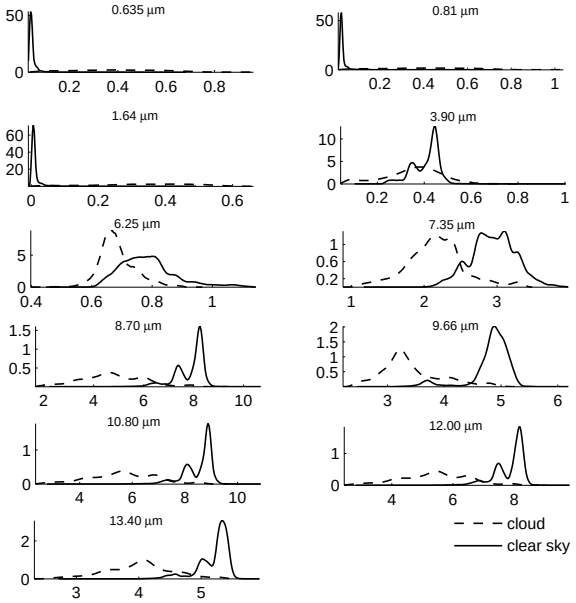

Figure 3.3: Probability density function (pdf) of reflectance/radiance corresponding to the 11 SEVIRI channels. Plots refer to the data-set A (daytime) and water pixels.

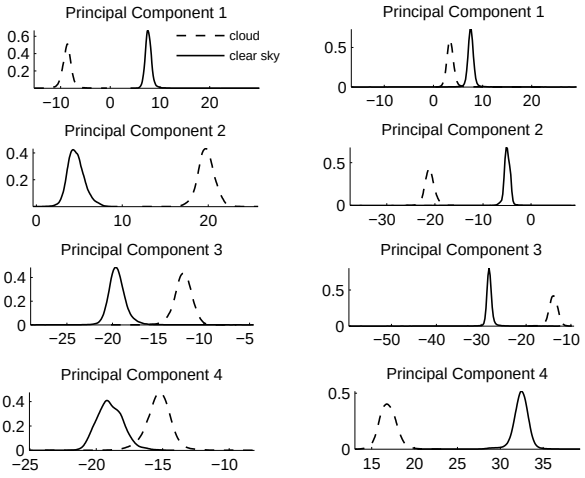

Figure 3.4: Probability density function (pdf) of the first 4 principal components of reflectance/radiance.Left: land pixels; right: water pixels. Plots refer to the data-set A (daytime).

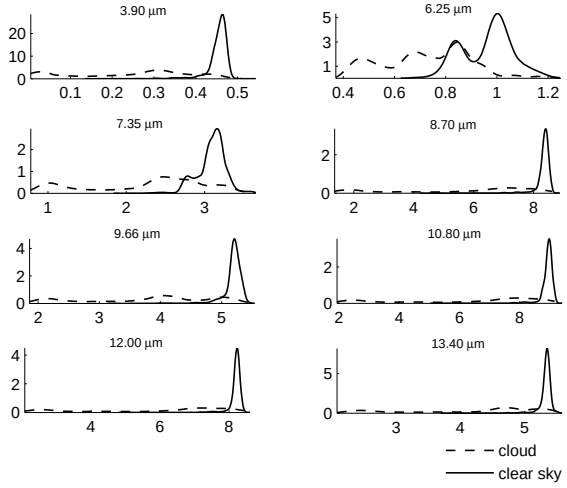

Figure 3.5: Probability density function (pdf) of reflectance/radiance corresponding to the 11 SEVIRI channels. Plots refer to the data-set E (nighttime) and water pixels.

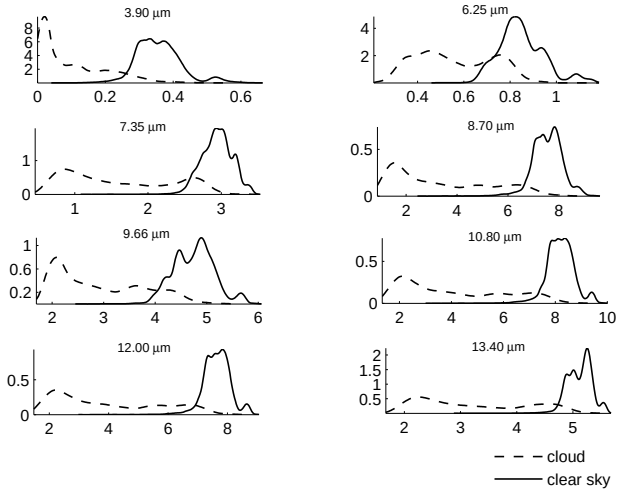

Figure 3.6: Probability density function (pdf) of reflectance/radiance corresponding to the 11 SEVIRI channels. Plots refer to the data-set E (nighttime) and land pixels.

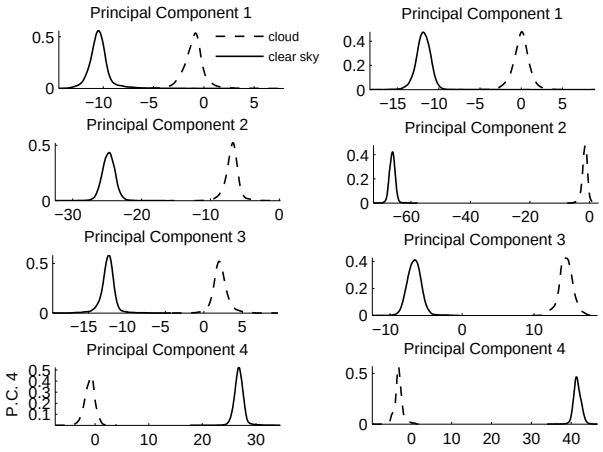

Figure 3.7: Probability density function (pdf) of the first 4 principal components of reflectance/radiance. Left: land pixels; right: water pixels. Plots refer to the data-set E (nighttime).

3.3.2 Experiment 1

Table 3.5 shows the success percentage, S, obtained by QDA and NPDA on the data-sets A (daytime) and E (nighttime) when only one spectral band is considered for classification. Of course in this case results of PCDA and ICDA coincide with NPDA because in this unispectral case original and transformed components coincide. The table aims at giving a first indication about the role of spectral bands in detecting clouds. A more accurate analysis on the use of more bands will be shown later on. Analysis of Table 3.5 indicates that IR bands are better suited to classify clouds over land whereas success percentage is higher for visible and near infrared bands over water. Actually there is a group of infrared spectral bands (8.70–13.40 μm for land) and a group of visible and near infrared spectral bands (0.635–1.64 μm for water) where top performance is quite homogeneous within. In both cases performance obtained by the other (less relevant) group of spectral bands is generally not too bad,

Band	Wavelength	Data-set A				Data-set E			
	(μm)	Land		Water		Land		Water	
		QDA	NPDA	QDA	NPDA	QDA	NPDA	QDA	NPDA
1	0.635	86.0	85.8	96.0	96.7	-	-	-	-
2	0.81	90.2	90.5	95.2	96.1	-	-	-	-
3	1.64	67.6	67.3	95.4	96.0	-	-	-	-
4	3.90	81.5	84.1	68.4	70.2	94.2	94.2	90.0	92.6
5	6.25	75.7	75.3	73.4	76.2	85.6	85.0	73.3	81.6
6	7.35	86.5	86.3	87.6	89.0	91.8	91.7	77.2	76.7
7	8.70	95.9	95.8	93.7	93.8	94.1	94.4	87.1	93.7
8	9.66	95.3	95.2	91.4	93.0	92.5	92.4	83.4	89.9
9	10.80	96.0	95.9	94.0	94.1	93.9	94.2	85.4	93.1
10	12.0	95.9	95.8	93.6	93.8	94.2	94.5	84.6	92.6
11	13.40	94.7	94.5	91.5	92.2	94.5	95.1	83.4	86.5

Table 3.5: Success percentage, S, obtained by QDA and NPDA separately over land and water for data-sets A (daytime) and E (nighttime) when only one single spectral band is used for classification

especially when compared with the physical cloud detection methods, where performances quickly drop with wavelength (see, e.g., [Lutz, 1999] for the SEVIRI cloud mask from EUMETSAT). Comparing Table 3.5 with Figures 3.2 - 3.7 we observe that such digits find an easy explanation in the density functions of radiance/reflectance, since best performances are obtained for those spectral bands whose density functions have least overlap between the cloudy and clear sky classes. Finally we observe that performance during the nighttime is worse than during daytime.

We have first considered the case of land pixels. Figure 3.8 shows global percentage of success, S, for the classification methods of Section 2.2.3 when the spectral bands are progressively chosen by the simple forward procedure. The figure refers to the data-set A of Table 3.2 when training and test data-sets coincide.

Figure 3.8 clearly shows all the features of the classification methods considered in the present work. In particular, QDA and ICDA–PCDA have higher performance; moreover NPDA has a performance curve that soon degrades after

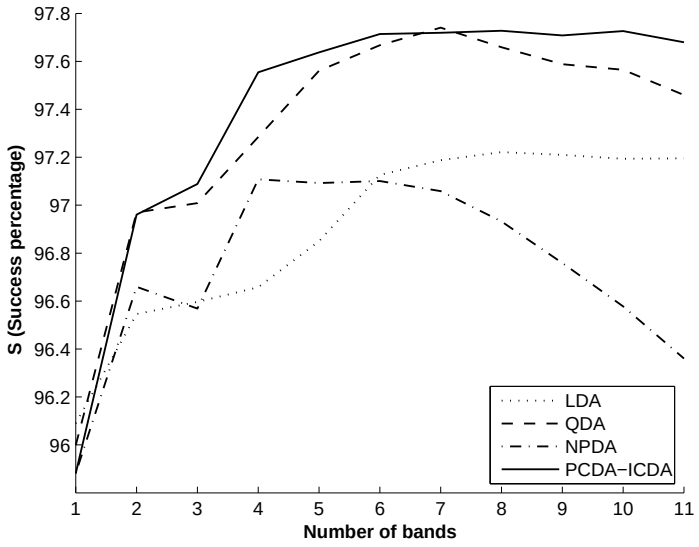

Figure 3.8: Success percentage, S, of the considered classification methods, when the data-set A is considered both as a training and test data-set. Plot refers to pixels over land.

only 4 spectral bands. In practice these curves are the result of transformation and nonparametric density estimation effects. NPDA curve says that a raw use of multispectrality is ineffective in improving classification rates with more spectral bands. Compared with LDA we see that the NPDA performs better with a few bands, but it degrades from the 5th on; these means that the potential increase of the information content of several bands is hindered by the poor nonparametric density estimate for some spectral bands. Therefore even a very simple LDA outperforms NPDA from 5 spectral bands on, because of the robust density estimation obtained by parametric methods (actually, estimate of mean and variance of Gaussian). Anyway, performances of both methods are constantly below QDA and PCDA–ICDA. As far as the latter are concerned, we observe that PCDA and ICDA are able to exploit multispectrality at best, because we do not observe significant decrease with the number of bands. QDA

suffers from a slight decrease of performance from 8 bands on due to the worse estimate of the covariance matrices for the least informative spectral bands. Notice that PCDA, which is nonparametric, does not show significant degradation of the performance also because, as observed in Figures 3.4 and 3.7 of Section 3.3.1, the density functions of the principal components have a much simpler (often Gaussian-like) shape, therefore nonparametric methods increase robustness in this case. This also explains why PCDA and ICDA have similar performances (recall that PCA only decorrelates general probability density functions but makes multivariate Gaussian independent). We also mention that ICDA (which relies on the fastica package) shows difficult or slow convergence in several circumstances. Therefore from now on, unless specified differently, when we show results concerning PCDA we mean that they apply to ICDA as well. In addition no practical difference was detected in the results by using the contrast functions defined in the fastica package and described in Section 2.2.2.

Finally results will be shown when the simple forward procedure has been used for selecting spectral bands progressively (discussed at the end of Section 3.2); actually they substantially coincide with the ones obtained when the forward– backward procedure is considered. A similar analysis conducted on the other data-sets (B and C), not shown here for brevity's sake, has given similar results.

Figure 3.10 shows these results for the nighttime data-set E. Of course lack of visible bands changes the framework deeply, mainly because the contribution of the visible and near infrared bands to the information content of data is missing. As a consequence success curves are now different from the daytime case and values are lower. QDA and PCDA–ICDA keep on being the best performing methods, even though dependence on the number of spectral bands is less clear. This is due to the fact that information content of infrared bands

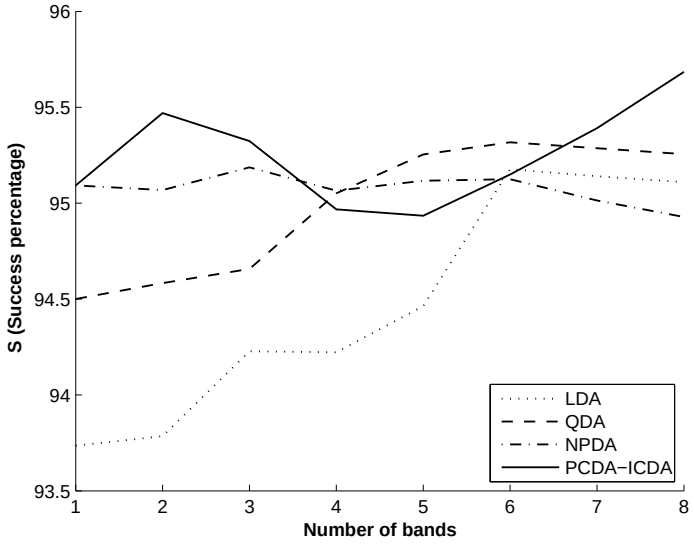

Figure 3.9: Success percentage, S, of the considered classification methods, when the data-set E is considered both as a training and test data-set. Plot refers to pixels over land.

is more homogeneous among bands as can be argued from the one-band success percentages shown in Table 3.5.

Figures 3.10 and 3.11 show the results of the same analysis performed on water pixels for the data-set A (daytime) and E (nighttime), respectively. The conclusion already drawn for land pixels also hold for this case.

It is now useful to discuss the selection of spectral bands accomplished by the methods. Tables 3.6 and 3.7 show the bands progressively chosen and the corresponding global success rate, S, for all the discriminant analysis methods when the training and test data-set A is considered (for land and water, respectively). Tables 3.8 and 3.9 refer to the same results as Tables 3.6 and 3.7 for the case of the nighttime data-set E. Results are consistent with indications from Table 3.5 in that the best performing groups of spectral bands are chosen first. In different data-sets (not shown here) the trend is similar, even though

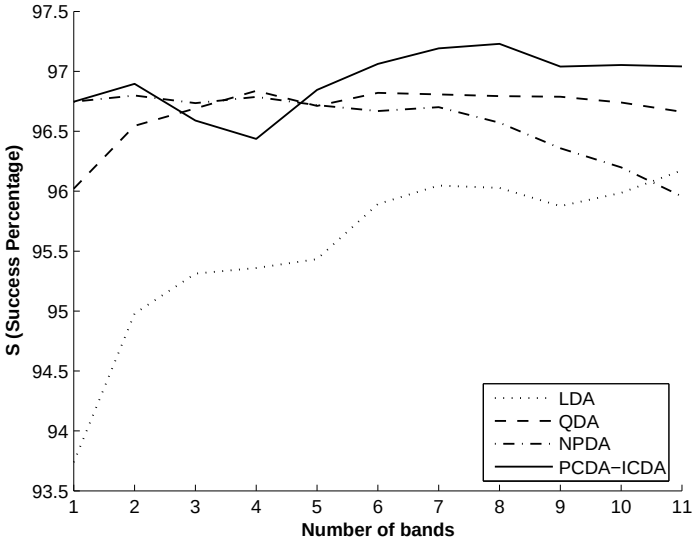

Figure 3.10: Success percentage, S, of the considered classification methods, when the data-set A is considered both as a training and test data-set. Plot refers to pixels over water

the order of choice of the spectral bands can be different among the groups due to their equivalent role in classification.

Finally we observe once more the important role of multispectrality, since adding less performing spectral bands improves the overall performance of the method even with respect to the best performing bands and is even able to make a simple LDA quite competitive.

Results of Figures 3.8, 3.9, 3.10 and 3.11 are global, in the sense that they refer to all clear and cloudy pixels; as a consequence the global indicator S could be biased by the different number of cloudy or clear pixels. For this reason Table 3.10 shows the full set of statistical indicators introduced in this section for all the SEVIRI data-sets of Table 3.2 disaggregated by class when classification is performed on land pixels. Results refer to the use of all the bands for all the classification methods. Each column corresponds to the experiment where the

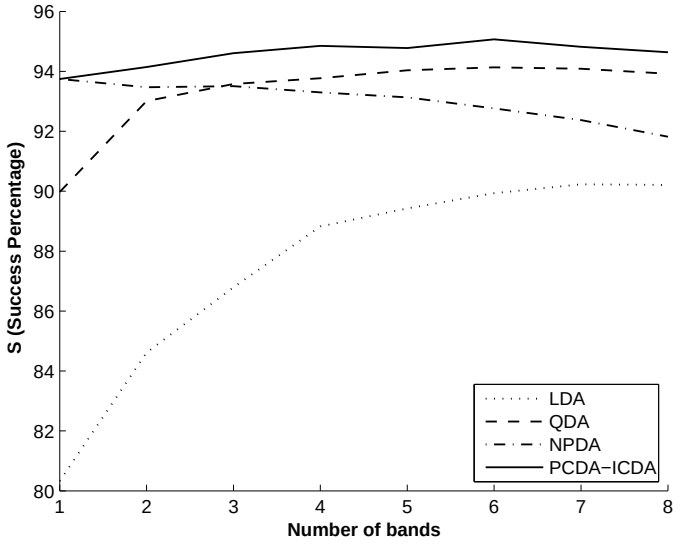

Figure 3.11: Success percentage, S, of the considered classification methods, when the data-set E is considered both as a training and test data-set. Plot refers to pixels over water.

same data-set is used for the training and the testing step (they are shown in the rows 'train' and 'test'); we again recall that only SEVIRI pixels composed of corresponding MOD35 pixels classified as confidently clear or cloudy take part to the analysis. The global success percentage, S, over those pixels (both clear and cloudy) already seen previously is shown in the column 'All', whereas in the columns 'Cloudy' and 'Clear' partial indicators for both conditions are separately given. The high values of the κ index for the global data-set (cloudy and clear sky pixels) suggest that the good values of global success percentage are shared also by clear and cloudy pixels (recall that for 2-class problem global κ index coincides with the class related ones). This is confirmed by inspection of the partial (cloudy or clear) success rates. We also notice that the number of false positive and negative rates is quite limited for both clear and cloudy conditions. In addition for the best performing methods differences between

LDA		QDA		NPDA		PCDA-ICDA	
λ (μm)	S	λ (μm)	S	λ (μm)	S	λ (μm)	S
10.80	96.1	10.80	96.0	10.80	95.9	10.80	95.9
3.90	96.5	3.90	97.0	1.64	96.7	3.90	97.0
8.70	96.6	1.64	97.0	0.635	96.6	13.40	97.1
13.40	96.7	13.40	97.3	13.40	97.1	7.35	97.5
7.35	96.8	7.35	97.6	6.25	97.1	0.81	97.6
6.25	97.1	8.70	97.7	0.81	97.1	6.25	97.7
1.64	97.2	6.25	97.7	12.00	97.1	9.66	97.7
0.81	97.2	0.81	97.7	7.35	96.9	1.64	97.7
0.635	97.2	9.66	97.6	8.70	96.8	12.0	97.7
9.66	97.2	12.00	97.6	9.66	96.6	0.635	97.7
12.00	97.2	0.635	97.5	3.90	96.4	8.70	97.7

Table 3.6: Bands chosen for the classification of the data-set A (daytime) over land pixels

the clear and cloudy classes became small, about 1% at most for PCDA for all cases.

Table 3.11 shows the same results as Table 3.10 when the classification is performed on water pixels. Examining Tables 3.10 and 3.11 we notice that the global success percentage S is, on the average, greater when the classification is performed on daytime data-sets (A, B and C) with respect to nighttime data-sets (D and E), both for land and water pixels. This result depends partly on the greater number of spectral bands used for daytime classification (SEVIRI visible and near infrared channels at 0.635 μm, 0.81 μm and 1.64 μm are missing for nighttime experiments), and partly on the intrinsic better performance of VNIR spectral bands with respect to IR bands over water pixels.

Finally, Figure 3.12 shows the cloud mask estimated by PCDA over all pixels of the considered area (that is, also pixels classified as probably cloudy or clear by MOD35); it has to be compared with Figure 3.1, which is the corresponding RGB image. Summarizing we can say that PCDA and QDA are excellent discriminant analysis tools to detect clouds, relying on an efficient treatment of multispectrality and on a robust estimate of density functions, respectively.

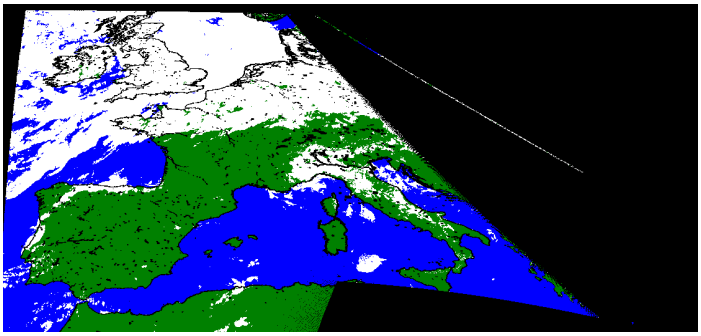

Figure 3.12: Cloud mask obtained by PCDA for the data-set A when the same data-set is used to train discriminant analysis. Black: unprocessed pixels; blue: clear pixels over water; green: clear pixels over land; white: cloudy pixels over land or water.

LDA		QDA		NPDA		PCDA-ICDA	
λ (μm)	S	λ (μm)	S	λ (μm)	S	λ (μm)	S
8.70	93.7	0.635	96.0	0.635	96.7	0.635	96.7
3.90	95.0	13.40	96.5	6.25	96.8	6.25	96.9
10.80	95.3	7.35	96.7	0.81	96.7	3.90	96.6
6.25	95.4	6.25	96.8	10.80	96.8	0.81	96.4
7.35	95.4	0.81	96.7	1.64	96.7	13.40	96.8
13.40	95.9	8.70	96.8	3.90	96.7	7.35	97.1
9.66	96.0	1.64	96.8	8.70	96.7	1.64	97.2
12.00	96.0	12.00	96.8	12.00	96.6	9.66	97.2
1.64	95.9	10.80	96.8	13.40	96.4	8.70	97.0
0.81	96.0	9.66	96.7	9.66	96.2	12.00	97.0
0.635	96.2	3.90	96.7	7.35	96.0	10.80	97.0

Table 3.7: Bands chosen for the classification of the data-set A (daytime) over water pixels.

LDA		QDA		NPDA		PCDA-ICDA	
λ (μm)	S	λ (μm)	S	λ (μm)	S	λ (μm)	S
3.90	93.7	13.40	94.5	13.40	95.1	13.40	95.1
6.25	93.8	6.25	94.6	12.00	95.1	3.90	95.5
9.66	94.2	3.90	94.7	6.25	95.2	7.35	95.3
7.35	94.2	9.66	95.0	8.70	95.1	6.25	95.0
10.80	94.5	10.80	95.2	7.35	95.1	9.66	94.9
12.00	95.2	8.70	95.3	3.90	95.1	10.80	95.1
8.70	95.1	12.00	95.3	9.66	95.0	8.70	95.4
13.40	95.1	7.35	95.3	10.80	94.9	12.00	95.7

Table 3.8: Bands chosen for the classification of the data-set E (nighttime) over land pixels.

LDA		QDA		NPDA		PCDA-ICDA	
λ (μm)	S	λ (μm)	S	λ (μm)	S	λ (μm)	S
3.90	80.3	3.90	90.0	8.70	93.7	8.70	93.7
8.70	84.6	10.80	93.0	3.90	93.5	6.25	94.1
6.25	86.8	13.40	93.6	6.25	93.5	3.90	94.6
10.80	88.8	6.25	93.8	10.80	93.3	13.40	94.8
12.00	89.4	8.70	94.0	12.00	93.1	10.80	94.8
13.40	89.9	9.66	94.1	9.66	92.8	12.00	95.1
7.35	90.2	12.00	94.1	7.35	92.4	9.66	94.8
9.66	90.2	7.35	93.9	13.40	91.8	7.35	94.6

Table 3.9: Bands chosen for the classification of the data-set E (nighttime) over water pixels.

train		A			B			C			D			E		
test		A			B			C			D			E		
		Cloudy	Clear	All	Cloudy	Clear	All	Cloudy	Clear	All	Cloudy	Clear	All	Cloudy	Clear	All
LDA	s	94.1	99.0	97.20	98.5	99.6	98.93	96.7	97.9	97.38	94.0	95.8	94.69	90.3	98.0	95.1
	κ	0.94	0.94	0.94	0.98	0.98	0.98	0.95	0.95	0.95	0.89	0.89	0.89	0.89	0.89	0.90
	F^+	1.6	3.5		0.3	2.2		2.5	2.7		2.5	10.1		3.2	5.9	
	F^-	5.9	1.0		1.5	0.4		3.3	2.1		6.0	4.2		9.7	2.0	
QDA	s	96.7	97.9	97.45	99.1	99.3	99.18	98.0	96.9	97.38	94.4	95.1	94.65	92.5	96.9	95.3
	κ	0.95	0.95	0.95	0.98	0.98	0.98	0.95	0.95	0.95	0.89	0.89	0.89	0.90	0.90	0.91
	F^+	3.5	2.0		0.5	1.3		3.8	1.7		2.9	9.5		5.0	4.6	
	F^-	3.3	2.1		0.9	0.7		2.0	3.1		5.6	4.9		7.5	3.1	
NPDA	s	95.6	96.8	96.36	98.7	99.2	98.89	98.1	96.3	97.13	89.8	95.5	91.91	91.7	96.9	94.9
	κ	0.92	0.92	0.93	0.98	0.98	0.98	0.94	0.94	0.94	0.83	0.83	0.84	0.89	0.89	0.90
	F^+	5.4	2.6		0.6	1.9		4.5	1.5		2.7	17.3		5.1	5.1	
	F^-	4.4	3.2		1.3	0.8		1.9	3.7		10.2	4.5		8.3	3.1	
PCDA	s	97.1	98.2	97.68	99.0	99.3	99.14	97.4	97.7	97.60	94.7	94.9	94.81	95.1	96.0	95.7
	κ	0.95	0.95	0.96	0.98	0.98	0.98	0.95	0.95	0.95	0.89	0.89	0.90	0.91	0.91	0.91
	F^+	3.1	1.7		0.5	1.4		2.7	2.1		3.0	8.9		6.5	3.0	
	F^-	2.9	1.8		1.0	0.7		2.6	2.3		5.3	5.1		4.9	4.0	

Table 3.10: Error indicators of LDA, QDA, NPDA and PCDA methods for classification on land pixels (Experiment 1).

train		A			B			C			D			E		
test		A			B			C			D			E		
		Cloudy	Clear	All	Cloudy	Clear	All	Cloudy	Clear	All	Cloudy	Clear	All	Cloudy	Clear	All
LDA	s	93.3	98.6	96.1	94.3	99.4	97.5	96.0	96.8	96.3	93.6	93.4	93.6	89.6	93.9	90.2
	κ	0.92	0.92	0.92	0.95	0.95	0.95	0.92	0.92	0.93	0.52	0.52	0.87	0.67	0.67	0.80
	F^+	1.6	5.8		1.0	3.3		1.5	8.2		0.3	147.7		1.0	63.5	
	F^-	6.7	1.4		5.7	0.6		4.0	3.2		6.4	6.6		10.4	6.1	
QDA	s	96.2	97.0	96.58	97.4	98.2	97.90	97.2	96.3	96.88	95.4	95.8	95.45	93.6	95.8	93.9
	κ	0.93	0.93	0.93	0.95	0.95	0.96	0.93	0.93	0.94	0.62	0.62	0.91	0.78	0.78	0.88
	F^+	3.5	3.4		3.1	1.5		1.8	5.9		0.2	105.2		0.7	39.0	
	F^-	3.8	3.0		2.6	1.8		2.8	3.7		4.6	4.2		6.4	4.2	
NPDA	s	95.5	96.2	95.87	96.8	97.9	97.50	97.3	95.4	96.68	94.3	93.3	94.30	91.3	94.8	91.8
	κ	0.92	0.92	0.92	0.95	0.95	0.95	0.92	0.92	0.93	0.55	0.55	0.89	0.72	0.72	0.84
	F^+	4.4	3.9		3.7	1.8		2.2	5.6		0.3	130.3		0.9	52.9	
	F^-	4.5	3.8		3.2	2.1		2.7	4.6		5.7	6.7		8.7	5.2	
PCDA	s	97.4	96.8	97.04	97.2	98.6	98.06	97.0	96.1	96.73	94.7	97.1	94.79	94.5	95.3	94.6
	κ	0.94	0.94	0.94	0.96	0.96	0.96	0.93	0.93	0.93	0.58	0.58	0.90	0.80	0.80	0.89
	F^+	3.7	2.3		2.5	1.6		1.9	6.1		0.1	122.4		0.8	33.4	
	F^-	2.6	3.2		2.8	1.4		3.0	3.9		5.3	2.9		5.5	4.7	

Table 3.11: Error indicators of LDA, QDA, NPDA and PCDA methods for classification on water pixels (Experiment 1).

train		A		A		B		B		C		C	
test		B		C		A		C		A		B	
		Cloudy	Clear	Cloudy	Clear	Cloudy	Clear	Cloudy	Clear	Cloudy	Clear	Cloudy	Clear
LDA	s	98.2	99.6	98.0	97.1	87.9	98.4	90.3	97.9	91.9	99.1	98.3	99.6
	κ	0.98	0.98	0.95	0.95	0.88	0.88	0.89	0.89	0.92	0.92	0.98	0.98
	F^+	0.3	2.5	3.5	1.6	2.7	7.2	2.6	8.0	1.5	4.8	0.3	2.4
	F^-	1.8	0.4	2.0	2.9	12.1	1.6	9.7	2.1	8.1	0.9	1.7	0.4
	S	98.8		97.5		94.5		94.4		96.4		98.8	
	$\mathcal{K}$	0.98		0.95		0.89		0.89		0.93		0.98	
QDA	s	99.1	99.3	98.2	95.8	95.5	92.9	96.5	97.8	96.4	93.9	98.4	99.3
	κ	0.98	0.98	0.94	0.94	0.87	0.87	0.94	0.94	0.89	0.89	0.98	0.98
	F^+	0.5	1.3	5.1	1.5	11.9	2.7	2.7	2.9	10.3	2.1	0.5	2.2
	F^-	0.9	0.7	1.8	4.2	4.5	7.1	3.5	2.2	3.6	6.1	1.6	0.7
	S	99.2		96.9		93.9		97.2		94.8		98.8	
	$\mathcal{K}$	0.98		0.94		0.88		0.94		0.90		0.98	
NPDA	s	98.8	99.0	98.3	95.7	95.2	96.2	97.4	96.8	95.7	96.2	98.9	98.9
	κ	0.98	0.98	0.94	0.94	0.91	0.91	0.94	0.94	0.92	0.92	0.98	0.98
	F^+	0.7	1.8	5.2	1.4	4.7	3.9	3.8	2.2	5.5	2.9	0.7	1.8
	F^-	1.2	1.0	1.7	4.3	4.8	3.8	2.6	3.2	4.3	3.8	1.1	1.1
	S	98.8		96.9		95.7		97.1		96.0		98.9	
	$\mathcal{K}$	0.98		0.94		0.91		0.94		0.92		0.98	
PCDA	s	99.3	99.0	98.4	95.1	96.0	95.4	95.3	94.5	96.0	94.2	98.3	98.9
	κ	0.98	0.98	0.93	0.93	0.91	0.91	0.90	0.90	0.90	0.90	0.97	0.97
	F^+	0.6	1.2	5.9	1.3	5.6	3.3	6.4	4.0	8.6	2.7	0.7	2.7
	F^-	0.7	1.0	1.6	4.9	4.0	4.6	4.7	5.5	4.0	5.8	1.7	1.1
	S	99.2		96.6		95.7		94.9		94.9		98.9	
	$\mathcal{K}$	0.98		0.93		0.91		0.90		0.92		0.98	

Table 3.12: Error indicators of LDA, QDA, NPDA and PCDA methods for classification of daytime data-sets on land pixels (Experiment 2).

train		A		A		B		B		C		C	
test		B		C		A		C		A		B	
		Cloudy	Clear	Cloudy	Clear	Cloudy	Clear	Cloudy	Clear	Cloudy	Clear	Cloudy	Clear
LDA	s	93.3	99.1	97.8	94.3	92.7	94.8	94.7	96.2	89.9	98.6	92.0	99.2
	κ	0.93	0.93	0.92	0.92	0.88	0.88	0.89	0.89	0.89	0.89	0.92	0.92
	F^+	1.5	3.8	2.7	4.6	6.0	6.4	1.8	11.1	1.6	8.9	1.8	4.6
	F^-	6.7	0.9	2.2	5.7	7.3	5.2	5.3	3.8	10.1	1.4	8.0	0.8
	S	97.0		96.7		93.8		95.1		94.5		96.5	
	$\mathcal{K}$	0.94		0.93		0.88		0.90		0.89		0.93	
QDA	s	96.7	98.3	99.1	90.1	97.5	93.7	99.2	89.6	93.9	96.3	95.4	98.9
	κ	0.95	0.95	0.91	0.91	0.91	0.91	0.91	0.91	0.90	0.90	0.95	0.95
	F^+	2.9	1.9	4.8	1.8	7.2	2.2	5.0	1.7	4.2	5.4	1.8	2.9
	F^-	3.3	1.7	0.9	9.9	2.5	6.3	0.8	10.4	6.1	3.7	5.0	1.1
	S	97.7		96.2		95.5		96.1		95.2		97.5	
	$\mathcal{K}$	0.95		0.92		0.91		0.92		0.90		0.95	
NPDA	s	95.3	98.2	98.7	89.4	96.8	89.8	99.1	82.7	92.5	97.5	82.8	98.5
	κ	0.94	0.94	0.90	0.90	0.86	0.86	0.85	0.85	0.90	0.90	0.84	0.84
	F^+	3.1	2.7	5.4	2.5	11.5	2.9	9.3	1.6	2.7	6.8	2.5	10.1
	F^-	4.7	1.8	1.3	10.6	3.2	10.2	0.9	17.3	7.5	2.5	17.2	1.5
	S	97.1		95.6		93.1		93.4		95.1		92.7	
	$\mathcal{K}$	0.94		0.91		0.86		0.87		0.90		0.85	
PCDA	s	96.6	98.3	98.7	86.1	93.9	93.3	98.7	87.0	91.6	97.0	94.7	99.1
	κ	0.95	0.95	0.87	0.87	0.87	0.87	0.91	0.91	0.89	0.89	0.95	0.95
	F^+	3.1	1.9	7.1	2.5	7.6	5.3	6.4	2.6	3.4	7.4	1.6	3.0
	F^-	3.4	1.7	1.3	13.9	6.1	6.7	1.3	13.0	8.4	3.0	5.3	0.9
	S	97.7		94.5		93.6		95.4		94.5		97.5	
	$\mathcal{K}$	0.95		0.87		0.87		0.91		0.89		0.95	

Table 3.13: Error indicators of LDA, QDA, NPDA and PCDA methods for classification of daytime data-sets on water pixels (Experiment 2).

		land pixels				water pixels			
train		D		E		D		E	
test		E		D		E		D	
		Cloudy	Clear	Cloudy	Clear	Cloudy	Clear	Cloudy	Clear
LDA	s	90.3	97.9	94.9	94.3	64.4	98.9	97.9	72.5
	κ	0.89	0.89	0.89	0.89	0.33	0.33	0.64	0.64
	F^+	3.4	5.9	3.4	8.6	0.2	218.6	1.2	48.5
	F^-	9.7	2.1	5.1	5.7	35.6	1.1	2.1	27.5
	S	95.0		94.7		69.2		96.8	
	$\mathcal{K}$	0.90		0.89		0.38		0.94	
QDA	s	91.4	97.7	95.4	91.9	83.9	96.7	97.5	86.1
	κ	0.90	0.90	0.87	0.87	0.58	0.58	0.70	0.70
	F^+	3.7	5.2	4.7	7.9	0.5	99.0	0.6	56.3
	F^-	8.6	2.3	4.6	8.1	16.1	3.3	2.5	13.9
	S	95.3		94.1		85.7		97.0	
	$\mathcal{K}$	0.91		0.88		0.71		0.94	
NPDA	s	80.2	99.4	97.1	66.4	72.3	98.4	98.8	64.2
	κ	0.82	0.82	0.67	0.67	0.41	0.41	0.65	0.65
	F^+	0.9	12.8	19.8	4.9	0.2	175.7	1.6	27.8
	F^-	19.8	0.6	2.9	33.6	27.7	1.6	1.2	35.8
	S	91.9		85.7		75.8		97.3	
	$\mathcal{K}$	0.84		0.71		0.52		0.95	
PCDA	s	91.2	97.4	95.9	88.8	84.7	95.5	97.9	81.2
	κ	0.89	0.89	0.85	0.85	0.59	0.59	0.69	0.69
	F^+	4.2	5.5	6.3	7.3	0.7	91.8	0.8	47.8
	F^-	8.8	2.6	4.1	11.2	15.3	4.5	2.1	18.8
	S	95.0		93.4		86.3		97.2	
	$\mathcal{K}$	0.90		0.87		0.73		0.94	

Table 3.14: Error indicators of LDA, QDA, NPDA and PCDA methods for classification of nighttime data-sets on land and water pixels (Experiment 2).

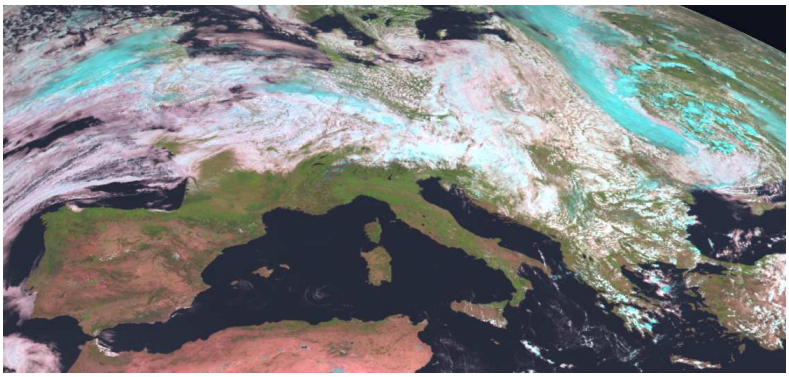

Figure 3.13: RGB image of European area, taken by SEVIRI sensor on July 15, 2004 10:27 (UTC) (data-set B).

3.3.3 Experiment 2

Figure 3.13 shows the RGB image corresponding to the data-set B. In Figure 3.14 the results of QDA classification when B is used as test data-set and A is used as training one are shown. As for the Experiment 1 we notice a good cloud detection both on land and water pixels. This fact is confirmed by numerical results.

Tables 3.12 and 3.14 show the statistical indicators introduced in this section for LDA, QDA, NPDA and PCDA-ICDA methods for daytime SEVIRI data-sets when the classification is performed on land and water pixels, respectively. Comparison with Tables 3.10 and 3.11 shows that the loss of performance in

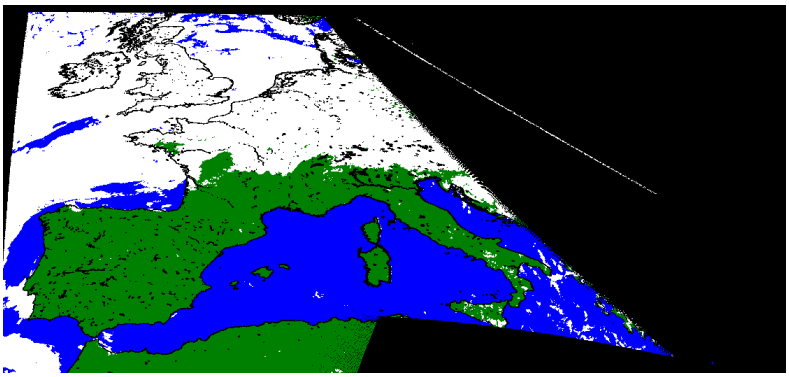

Figure 3.14: Cloud mask obtained by QDA for the data-set B when the data-set A is used to train discriminant analysis. Black: unprocessed pixels; blue: clear pixels over water; green: clear pixels over land; white: cloudy pixels over land or water.

classification is very limited, especially considering that using a training set at a different period than the test one can produce a change of the spectral signature, both for the possibly different cloud typologies and for the land characteristics. Finally Table 3.14 shows a not very good performance for an instance over water (training set D and test set E); it is due to a high number of false positive clear pixels, which occurs also for the reverse case (training set E and test data-set D). In this respect we observe from Table 3.4 that the number of clear pixels used for the training step is very small (about 3000 - 5000), therefore they are not fully representative of the global physical characteristics of water pixels.

Chapter 4

Clustering and Consensus Clustering: Background

4.1 Clustering: an open problem

A large number of clustering definitions can be found in the literature. The simplest definition is shared among all and includes one fundamental concept: the goal of data clustering, also known as cluster analysis, is to discover the natural grouping(s) of a set of patterns, points, or objects [Jain, 2009].

An operational definition of clustering can be stated as follows: given a representation of N objects, find k groups based on a measure of similarity such that the similarities between objects in the same group (cluster) are high while the similarities between objects in different groups are low. An ideal cluster can be defined as a set of points that is compact and isolated. Actually, a cluster is a subjective entity that is in the eye of the beholder and whose significance and interpretation requires domain knowledge. But, while humans are excellent cluster seekers in two and possibly three dimensions, we need automatic algorithms for high dimensional data. It is this challenge along with the unknown number of clusters for the given data that has resulted in thousands of clustering algorithms that have been published and that continue to appear

[Jain, 2009].

Although in the literature there are as many different classifications of clustering algorithms as the number of algorithms itself, there is one simple classification that allows essentially splitting them into the following two main classes:

- Hierarchical Clustering
- Partitional Clustering

Hierarchical clustering creates a hierarchy of clusters which may be represented in a tree structure called dendrogram [Duda et al., 2001]. The root of the tree consists of a single cluster containing all observations, and the leaves correspond to individual observations. Algorithms for hierarchical clustering are generally either agglomerative, in which one starts at the leaves and successively merges clusters together; or divisive, in which one starts at the root and recursively splits the clusters. Any valid metric may be used as a measure of similarity between pairs of observations. The choice of which clusters to merge or split is determined by a linkage criterion, which is a function of the pairwise distances between observations. The major drawback of this kind of approach is that the entire dendrogram is sensitive to previous (and possible erroneous) cluster merging (or splitting) i.e data are not permitted to change cluster membership once assignment has taken place.

Partitional methods attempt to minimize a cost function or an optimality criterion which associates a cost to each instance-cluster assignment. The goal is to solve an optimization problem to satisfy the optimality criterion imposed by the model, which often means minimizing the cost function.

One of the most popular partitional clustering is the classic K-means algorithm, developed 50 years ago in different scientific fields [Ball and Hall, 1965, MacQueen, 1967]. The K-means algorithm assigns each point to the cluster whose center (also called centroid) is nearest. The center is the average of all

the points in the cluster — that is, its coordinates are the arithmetic mean for each dimension separately over all the points in the cluster. The main advantages of this algorithm are its simplicity and speed which allows it to run on large data-sets. Its disadvantage is that it does not yield the same result with each run, since the resulting clusters depend on the initial random assignments. It minimizes intra-cluster variance, but does not ensure that the result has a global minimum of variance. Another disadvantage is the requirement for the concept of a mean to be definable which is not always the case. For such data-sets the K-medoids variant (see the most common realization Partitioning Around Medoids (PAM) algorithm [Theodoridis and Koutroumbas, 2006]) is appropriate. Other popular variants of K-means include the Fast Genetic K-means Algorithm (FGKA) [Lu et al., 2004a] and the Incremental Genetic K-means Algorithm (IGKA) [Lu et al., 2004b].

Other examples of partitional clustering algorithms are Fuzzy C-means clustering [Bezdek, 1981], Gaussian mixture models [McLachlan and Basford, 1988], QT (quality threshold) clustering [Heyer et al., 1999], Simulating Annealing (SA) method [Gelatt et al., 1983] and spectral clustering [Yu and Shi, 2003].

If the goal of traditional clustering is to assign each data point to one and only one cluster, in contrast, Fuzzy C-means clustering assigns different degrees of membership to each point. The membership of a point is thus shared among various clusters. This creates the concept of fuzzy boundaries which differs from the traditional concept of well-defined boundaries. In the Gaussian mixture models approach the data are viewed as coming from a mixture of Gaussian densities, each representing a different cluster. The EM algorithm [Dempster et al.,1977] is often used to infer the parameters of the models. Several Bayesian approaches have been developed to improve the mixture models for data clustering, including Latent Dirichlet Allocation (LDA) [Blei et al., 2003]

and Pachinko Allocation model [Li and McCallum, 2006]. QT (quality threshold) clustering is an alternative method of partitioning data, invented for gene clustering. It requires more computing power than K-means, but does not require specifying the number of clusters a priori, and always returns the same result when run several times. The Simulating Annealing (SA) method was developed in analogy to an experimental annealing procedure, where the stability of metal or glass is improved by heating or cooling. Solutions for an optimization problem are heated and heated in simulations to find a "good" quality solution, i.e. one admissible solution with very low cost. In the case of clustering, a solution which achieved a low value of the associated cost function can be accepted. While convergence in probability to the global minimum has been established, SA techniques are often slow because of its randomized stochastic search in the whole parameter space. Deterministic Annealing (DA) methods intend to overcome this deficiency, while preserving the main advantages of SA. Spectral clustering techniques make use of the spectrum of the similarity matrix of the data to perform dimensionality reduction for clustering in fewer dimensions.

Finally, among all clustering methods, it is important to remember also the self-organizing map (SOM) or self-organizing feature map (SOFM). A self-organizing map is a type of artificial neural network that is trained using unsupervised learning to produce a low-dimensional (typically two-dimensional), discretized representation of the input space of the training samples, called a map. Self-organizing maps are different from other artificial neural networks in the sense that they use a neighborhood function to preserve the topological properties of the input space. This makes SOMs useful for visualizing low-dimensional views of high-dimensional data. The model was first described as an artificial neural network by the Finnish professor Teuvo Kohonen, and is sometimes called a Kohonen map [Kohonen , 2001]. Like most artificial neural

networks, SOMs operate in two modes: training and mapping. Training builds the map using input examples. It is a competitive process, also called vector quantization. Mapping automatically classifies a new input vector.

Despite the development of so many clustering algorithms, and their successful application in a lot of different fields, clustering remains an open problem [Jain, 2009] for various reasons: the ambiguous definition of a cluster, the choice of features used to represent the data, the determination of the number of clusters in the data, the difficulty in defining an appropriate similarity measure and an objective function are only few examples. But one of more discussed problems concerns the cluster validity: clustering algorithms tend to find clusters in the data irrespective of whether or not any clusters are present.

Moreover different clustering algorithms applied to the same data-set produce different solutions because each algorithm imposes a structure on the data. It is clear that the "best" clustering algorithm does not exist and it is better to try different approaches to determine the solution, particularly when there is no a priori knowledge on the data structure. An interesting question is to identify algorithms that generate similar partitions irrespective of the data. In other words a clustering of clustering algorithms has to be performed. In [Jain et al., 2004], for example, the authors clustered 35 different clustering algorithms into 5 groups based on their partitions on 12 different data-sets. The similarity between the clustering algorithms is measured as the averaged similarity between the partitions obtained on the 12 data sets. The similarity between a pair of partitions is measured using the Adjusted Rand Index (ARI) [Hubert and Arabie, 1985] .

Additionally, many popular clustering algorithms, as partitional clustering and model-based clustering, are based on initial random assignments or follow random procedures. Thereby it is common to obtain different clusterings, when

the same algorithm runs several times on the same data, and to explain the data distribution properly with more solutions. Obviously choosing a single solution becomes in some how arbitrary.

All these considerations lead to important questions: as different clustering algorithms (or also more runs of the same algorithm) find different data partitions, are the discovered clusters valid? Does the true solution really exist or is it an utopia? And even more interestingly, how is important to find this hypothetical single solution? These questions have introduced new trends in data clustering research leading to the development of the "consensus clustering" concept.

4.2 Introduction to Consensus Clustering

Consensus clustering, also known in literature as clustering ensembles or clustering aggregation has emerged as an important elaboration of the classical clustering problem. Consensus clustering can be defined as the process of combining multiple individual clustering results obtained for a particular data-set into a single consensus solution which is a better fit in some sense than the existing clusterings. When cast as an optimization problem, consensus clustering is known as median partition, and has been shown to be NP-complete [Barthlemy and Leclerc, 1995].

As mentioned in Section 4.1, we may use a variety of clustering algorithms to partition a data-set into several clusters. Each of these clustering algorithms has its own clustering criteria and imposes partitions on the data based on certain assumptions. Due to the lack of prior information about the underlying cluster structure, which is inherent to cluster analysis, we usually do not know which algorithm to choose in order to correctly identify this structure. Researchers have thus attempted to avoid selecting one particular criterion/algorithm by using

instead a set of clustering solutions produced by different algorithms, called a cluster ensemble, and then incorporate them into a single partition referred to as the consensus solution. There are many different ways of generating a clustering ensemble and then combining the partitions. For example, multiple data partitions can be generated by: multiple clustering algorithms, multiple runs with random initializations of the same clustering algorithm, subsets re-sampled from a data-set, combining of different data representations (feature spaces), etc. A cluster ensemble improves clustering performance, as it can compensate for possible errors made by some clustering solutions by introducing the correct output of others; hence it can be more accurate and robust than each of the individual components.

As for the clustering, also for the consensus clustering problem, a lot of algorithms have been developed to solve different questions in many fields of applications. In the following we show a brief overview of the most important methodologies used for this kind of problems leaving out the algorithms details.

Strehl and Ghosh [Strehl and Ghosh, 2002] consider various formulations for the consensus clustering, most of which reduce the problem to a hyper-graph partitioning problem. This approach introduces the problem of combining multiple partitionings of a set of objects into a single consolidated clustering without accessing the features or algorithms that determined these partitionings. They discuss three approaches towards solving this problem to obtain high quality consensus functions. Their techniques have low computational costs and this makes it feasible to evaluate each of the techniques discussed below and arrive at the best solution by comparing the results against the objective function. The first step of the consensus functions is to transform the data partitions into a hyper-graph representation. The Cluster-based Similarity Partitioning Algorithm (CSPA) uses a pairwise similarity: the similarity between two data-points

is defined to be directly proportional to number of constituent clusterings of the ensemble in which they are clustered together. The intuition is that the more similar two data-points are the higher is the chance that constituent clusterings will place them in the same cluster. CSPA is the simplest heuristic, but its computational and storage complexity are quite expensive. The HyperGraph Partitioning Algorithm (HGPA) obtains the combined partition by partitioning the hyper-graph into k unconnected components of approximately the same size, by cutting a minimum number of hyper-edges. Finally, the Meta-CLustering Algorithm (MCLA) is based on clustering clusters: each cluster is represented by a hyper-edge. The idea in MCLA is to group and collapse related hyper-edges and assign each object to the collapsed hyper-edge in which it participates most strongly.

Also based on the hyper-graph theory is the work of Fern and Brodley [Fern and Brodley, 2004] who proposed the Hybrid Bipartite Graph Formulation (HBGF) algorithm that forms a bipartite graph between clusters and data-points, and then partitions the graph to obtain the final consensus clustering. This paper proposes a new graph formulation that simultaneously models both instances and clusters as vertices in a bipartite graph. Such a graph retains all of the information of an ensemble, allowing both the similarity among instances and the similarity among clusters to be considered collectively to construct the final clusters.

Punera and Ghosh [Punera and Ghosh, 2008] extended the idea of hard clustering ensembles to the soft clustering scenario: while the other techniques are very varied in the algorithms they employ, the common thread is that they only work with hard constituent clusterings. The authors investigated Soft Cluster Ensembles and developed a "soft version" of CSPA, MCLA [Strehl and Ghosh, 2002] and HBGF [Fern and Brodley, 2004] algorithms named respectively sCSPA (soft

CSPA), sMCLA (soft MCLA) and sHBGF (soft HBGF).

In [Gionis et al., 2007] the authors consider the following problem: given a set of clusterings, find a single clustering that agrees as much as possible with the input clusterings. This problem, known as clustering aggregation, appears naturally in various contexts. For example, clustering categorical data is an instance of the clustering aggregation problem; each categorical attribute can be viewed as a clustering of the input rows where rows are grouped together if they take the same value on that attribute. Clustering aggregation can also be used as a meta clustering method to improve the robustness of clustering by combining the output of multiple algorithms. Furthermore, the problem formulation does not require a priori information about the number of clusters; it is naturally determined by the optimization function. In this work, Gionis et al. give a formal statement of the clustering aggregation problem, and propose a number of algorithms which make use of the connection between clustering aggregation and the problem of correlation clustering.

In [Fred and Jain, 2002] the idea of evidence accumulation (EAC) for combining the results of multiple clusterings is addressed. Given a data set (N objects or patterns in D dimensions), a clustering ensemble (a set of object partitions) is produced. According to the EAC concept, each partition is viewed as an independent evidence of data organization, individual data partitions being combined, based on a voting mechanism, to generate a new $N \times N$ similarity matrix between the N patterns. The final data partition of the N patterns is obtained by applying a hierarchical agglomerative clustering algorithm on this matrix. Also the authors have developed a theoretical framework for the analysis of the proposed clustering combination strategy and its evaluation, based on the concept of mutual information between data partitions.

The study performed in [Topchy et al., 2005] extends previous research on

clustering ensembles in several respects. The authors propose a probabilistic model of consensus using a finite mixture of multinomial distributions in a space of clusterings. A combined partition is found as a solution to the corresponding maximum-likelihood problem using the EM algorithm. Also they define a new consensus function that is related to the classical intraclass variance criterion using the generalized mutual information definition and demonstrate the efficacy of combining partitions generated by weak clustering algorithms that use data projections and random data splits.

The problem of consensus clustering is of particular significance in the emerging field of gene expression data analysis and functional genomics, where the need for the molecular-based refinement of broadly defined biological classes is an active field of study, in particular in cancer diagnosis, prognosis, and treatment, among others. In gene expression data analysis the relatively small sample size is compounded by the very high dimensionality of the data available and this fact makes the clustering results especially sensitive to noise and susceptible to over-fitting. A lot of proposals exist for the use of resampling and cross validation techniques to simulate perturbations of the original data set, so as to assess the stability of the clustering results with respect to sampling variability [Ben-Hur et al., 2002, Bertoni and Valentini, 2007, Bhattacharjee et al., 2001, Dudoit and Fridlyand, 2002]. Upon some of those ideas Monti et al. [Monti et al., 2003] develop a general, model-independent resampling-based methodology of class discovery and clustering validation and visualization tailored to the task of analyzing gene expression data. They call the new methodology consensus clustering, as it provides for a method to represent the consensus across multiple runs of a clustering algorithm, to determine the number of clusters in the data, and to assess the stability of the discovered clusters. The method can also be used to represent the consensus over multiple

runs of a clustering algorithm with random restart (such as K-means, model-based Bayesian clustering, SOM, etc.), so as to account for its sensitivity to the initial conditions.

4.3 Meta Clustering: does exist a unique solution?

The brief review of Section 4.2 shows that most ensemble methods combine the clusterings they identify into a one final clustering because their goal is to find a better, single, very compact clustering. But as different clustering algorithms (or also more runs of the same algorithm with different parameters and/or initializations) applied to the same data-set find different data partitions, it is right to think that the "true" solution does not really exist and, even more interestingly, the final goal is not to find this hypothetical single solution. In fact in many applications different clusterings can put in evidence distinct groupings of the data which find a meaningful explanation in the nature of the problem. A typical example comes from the biological data analysis: different partitions of the same data-set can reveal different subtypes of tumors or diseases which could not emerge from a unique solution. In these cases the real problem is the analysis of a small group of equivalently good solutions rather than the search for the best partition of the data-set.

This idea is at the basis of the so called meta clustering which does not attempt to combine different clusterings into one clustering. Instead, it groups different clusterings into meta clusters to allow users to select the clustering that is most useful for them.

A useful work on this topic is presented in [Caruana et al., 2006] where the authors introduce the meta clustering as a new approach to the problem of clustering: rather than finding one optimal clustering of the data, meta clustering

finds many alternate good clusterings of the data and allows the user to select which of these clusterings is most useful, exploring the space of reasonable clusterings. To prevent the user from having to evaluate too many clusterings, the many base-level clusterings are organized into a meta clustering, a clustering of clusterings that groups similar base-level clusterings together. This meta clustering makes it easier for users to evaluate the clusterings and efficiently navigate to the clustering(s) useful for their purposes. The whole process is composed of three steps. First, a large number of potentially useful high-quality clusterings is generated. Then a distance metric over clusterings measures the similarity between pairs of clusterings. Finally, the clusterings are themselves clustered at the meta level using the computed pairwise similarities. The clustering at the meta level allows the user to select a few representative yet qualitatively different clusterings for examination. If one of these clusterings is appropriate for the task at hand, the user may then examine other nearby clusterings in the meta level space.

Founded on this main idea, the goal of the second proposed application is to develop an automatic procedure which, starting from the generation of an initial ensemble of clustering solutions for a certain selected data-set and passing through well defined steps, allows to provide a limited number of different equivalently "good" clustering solutions. To this purpose we propose a consensus clustering algorithm called Least-Squares Consensus Clustering as explained in Section 5.1. This method extends the idea of the Least-Squares Clustering [Dahl, 2006] and allows to extrapolate in an automatic way a small number of different clustering solutions from an initial (large) set of clusterings obtained by applying any clustering algorithm to a selected data-set. As for a consensus clustering algorithm it is fundamental to evaluate the obtained results, we also define a measure of quality in terms of Least-Squares Error (see Section 5.2).

In addition to evaluate the level of representativeness of the consensus clustering solutions, this measure of quality represents the discrimination threshold to select a small group of meaningful solutions. In order to have an immediate feedback on the analysis results, we also suggest a graphical visualization (see Section 5.6).

As we will explain in more details in Chapter 5, unlike related works (also [Caruana et al., 2006]) the developed methodology is completely automatic and totally independent from the methods used for the generation of the initial clusterings ensemble. Also it is user-independent because, once selected the data-set and generated the clusterings ensemble, the user is only called to analyze and understand a limited group of solutions provided by the procedure itself.

Chapter 5

Least Squares Consensus Clustering Algorithm

5.1 Least-Squares Consensus Clustering

The proposed consensus clustering algorithm is based on the idea of Least-Squares Clustering (LS) used in [Dahl, 2006] which describes a model-based clustering procedure for microarray expression data based on a well-defined statistical model, specifically, a conjugate Dirichlet process mixture (DPM) model. In the assumed model, two genes come from the same mixture component if and only if their relevant latent variables governing expression are equal. The model is fit using Markov Chain Monte Carlo (MCMC). Each iteration of the Markov chain yields a clustering of the data. Providing a single point estimate for clustering based on the thousands of clusterings in the Markov chain has been proved to be challenging [Medvedovic and Sivaganesan, 2002]. One approach is to select the observed clustering with the highest posterior probability; this is called the maximum a posteriori (MAP) clustering. Unfortunately, the MAP clustering may only be slightly more probable than the next best alternative, yet represents a very different allocation of observations. Alternatively, Medvedovic and Sivaganesan suggest using hierarchical agglomerative clustering based on a

distance matrix formed using the observed clusterings in the Markov chain. As alternative, Dahl proposes a method to form a clustering from the many clusterings observed in the Markov chain. The method is called Least-Squares Model Based Clustering (or, simply, Least-Squares Clustering). It selects the observed clustering from the Markov chain that minimizes the sum of squared deviations from the averaged pairwise probability matrix that elements are clustered together.

Starting from this basic idea, we have defined a Least-Squares Consensus Clustering of a set of clusterings solutions and its associated quality measure.

Let Y be a given data-set of dimensions $N \times D$, where N is the number of elements to cluster and D is the number of features. Let γ_k be a vector of dimension N, encoding a clustering solution for the data-set Y. We define $\Gamma = \{\gamma_1, \ldots, \gamma_M\}$ a collection of $M >> 1$ distinct clustering solutions for Y obtained from any clustering algorithm.

For each clustering $\gamma_k \in \Gamma$, $k = 1, \ldots, M$, we have built an association matrix $\delta(\gamma_k)$ of dimension $N \times N$, whose (i, j) element is $\delta_{i,j}(\gamma_k)$, an indicator of whether element i is clustered with element j. Element-wise averaging of the association matrices of all the clusterings $\gamma_k \in \Gamma$ yields the pairwise probability matrix of clusterings:

$$\pi = \frac{1}{|\Gamma|} \sum_{\gamma_k \in \Gamma} \delta_{i,j}(\gamma_k). \tag{5.1}$$

where $|\Gamma|$ is defined as the number of elements in the set Γ.

Specifically, the Least-Squares Consensus Clustering $\hat{\gamma}_{LS}$ is defined as the observed clustering which minimizes the sum of squared deviations of its association matrix from the pairwise probability matrix π:

$$\hat{\gamma}_{LS} = \arg\min_{\gamma_k \in \Gamma} \sum_{i=1}^{N} \sum_{j=1}^{N} (\delta_{i,j}(\gamma_k) - \pi_{i,j})^2. \tag{5.2}$$

The Least-Squares Consensus Clustering presents many advantages:

1. it uses information from all the starting clusterings via the pairwise probability matrix;

2. it selects as consensus one of the original clusterings, instead of forming a new clustering via an external, ad hoc algorithm;

3. it is independent from the number of clusters k present in each single clustering;

4. the consensus clustering presents a label for all the N data-set elements: none element is eliminated by the procedure.

5.2 Least-Squares Error

The aim of a consensus clustering algorithm is to combine different clustering solutions to obtain a new final clustering which is representative of the initial clustering ensemble. The level of representativeness of the obtained consensus clustering should be evaluated using an objective criterion. For this reason we have defined a quality measure called Least-Squares Error which account for the goodness of the Least-Squares Consensus Clustering.

More specifically, given a Least-Squares Consensus Clustering $\hat{\gamma}_{LS}$ for a certain group of clustering solutions $\Gamma = \{\gamma_1, \ldots, \gamma_M\}$ we have defined the Least-Squares Error E_{LS} associated to a Least-Squares Consensus Clustering $\hat{\gamma}_{LS}$ as

$$E_{LS}(\Gamma) = \frac{1}{|\Gamma|} \sum_{\gamma_k \in \Gamma} \sum_{i=1}^{N} \sum_{j=1}^{N} (\delta_{i,j}(\gamma_k) - \delta_{i,j}(\hat{\gamma}_{LS}))^2. \tag{5.3}$$

Intuitively, the Eq. (5.3) shows that the greater is the Least-Squares Error E_{LS} the more distant is the Least-Squares Consensus Clustering $\hat{\gamma}_{LS}$ from the other clusterings $\gamma_k \in \Gamma, k = 1, \ldots, M$. On the other hand, when the Least-

Squares Error E_{LS} is small, the clustering $\hat{\gamma}_{LS}$ is closer to the other clusterings and therefore it is more representative of the whole group.

As well as to define the level of representativeness of the Least-Squares Consensus Clustering, this measure of quality represents the discrimination threshold to select a small group of meaningful solutions as explained in Section 5.4.

5.3 Similarity measure and hierarchical clustering

Whereas Least-Squares Error can provide a measure of the closeness of a group of clusterings, the pairwise comparison of two partitions can be done using similarity measures such as Minkowski Index, Jaccard Coefficient, correlation and matching coefficients (see [Ben-Hur et al., 2002] for a review). In our studies we used a measure S based on the entropy of the confusion matrix between clustering solutions [Bishehsari et al., 2007].

Given two clustering solutions γ_l and γ_r, where γ_l is made of n clusters and γ_r is made of m clusters, we define the confusion matrix Z^{lr} between γ_l and γ_r as a matrix which entries are the number of elements belonging to the cluster i of γ_l, denoted as γ_l^i , and to the cluster j of γ_r, denoted as γ_r^j :

$$Z_{i,j}^{lr} =\mid \{a_{kil} \in \gamma_l^i, k = 1, \ldots, \mid \gamma_l^i \mid : a_{kil} \in \gamma_r^j\} \mid \tag{5.4}$$

for $i = 1, \ldots, n$ and $j = 1, \ldots, m$. The obvious tool to measure the disorder of a cluster is the entropy H. If R_i is the i-th row of Z and C_j is the j-th column of Z, then $H(R_i)$ measures the disorder of the i-th cluster of γ_l with respect to γ_r, and $H(C_j)$ measures the disorder of the j-th cluster of γ_r with respect to γ_l. The similarity of γ_r versus γ_l is defined as the mean entropy of the clusters of γ_r versus γ_l:

$$S(Z^{lr}) = \sum_i (P(\gamma_l^i) \cdot H(R_i)) \tag{5.5}$$

where the a-priori probability of a cluster γ_l^i, $P(\gamma_l^i)$, can be approximated as $(\mid \gamma_l^i \mid)/(\text{total number of objects})$. The similarity of γ_l versus γ_r can be obtained with the analogue formula on C_j , which turns to be $S((Z^{lr})')$. As in general is $S(Z^{lr}) \neq S((Z^{lr})')$ the final measure of similarity between the two clusterings lies in the trade-off between $S(Z^{lr})$ and $S((Z^{lr})')$ and is defined as follows:

$$S_a(Z^{lr}) = S(Z^{lr}) + a \cdot S((Z^{lr})') \tag{5.6}$$

where $a \in [0, 1]$ can be used to set the acceptable level of "sub-clusteringness" of γ_r versus γ_l (see [Bishehsari et al., 2007] for details).

When a collection of M clustering solutions $\Gamma = \{\gamma_1 \dots \gamma_M\}$ is considered, the similarity measure (Eq. (5.6)) can be computed for any pair of clusterings $(\gamma_r, \gamma_l) \in \Gamma$ and assembled in a similarity matrix S_M of dimension $M \times M$. In order to visualize the relationships between the different clusterings, we can apply a hierarchical clustering algorithm on the similarity matrix S_M, giving origin to a meta-clustering dendrogram in which leaves represent the clustering solutions.

In general for a selected data-set of dimensions $N \times D$, where N is the number of items to be clustered and D the number of dimensions, given an $N \times N$ distance (or similarity) matrix, the basic process of hierarchical clustering (defined by S.C. Johnson in [Johnson, 1967]) is the following:

1. Start by assigning each item to a cluster, so that if there are N items, there are also N clusters, each containing just one item. Let the distances (similarities) between the clusters the same as the distances (similarities) between the items they contain.

2. Find the closest (most similar) pair of clusters and merge them into a single cluster, so that now there is one cluster less.

3. Compute distances (similarities) between the new cluster and each of the old clusters.

4. Repeat steps 2 and 3 until all items are clustered into a single cluster of size N.

Step 3 can be done in different ways, which depends on the selected linkage criterion.

If cluster r is formed from clusters p and q, n_r is the number of objects in cluster r and x_{ri} is the i-th object in cluster r, the main linkage functions can be summarized as follows:

- Single linkage, also called nearest neighbor, uses the smallest distance between objects in the two clusters:

$$d(r,s) = min(dist(x_{ri}, x_{sj})), i \in (1,\ldots,n_r), j \in (1,\ldots,n_s) \tag{5.7}$$

- Complete linkage, also called furthest neighbor, uses the largest distance between objects in the two clusters:

$$d(r,s) = max(dist(x_{ri}, x_{sj})), i \in (1,\ldots,n_r), j \in (1,\ldots,n_s) \tag{5.8}$$

- Average linkage uses the average distance between all pairs of objects in any two clusters:

$$d(r,s) = \frac{1}{n_r n_s} \sum_{i=1}^{n_r} \sum_{j=1}^{n_s} dist(x_{ri}, x_{sj}) \tag{5.9}$$

- Centroid linkage uses the Euclidean distance between the centroids of the two clusters:

$$d(r,s) = |\overline{x}_r - \overline{x}_s| \tag{5.10}$$

where

$$\overline{x}_r = \frac{1}{n_r} \sum_{i=1}^{n_r} x_{ri}$$

- Median linkage uses the Euclidean distance between weighted centroids of the two clusters:

$$d(r,s) = |\tilde{x}_r - \tilde{x}_s| \tag{5.11}$$

where $\tilde{x}_r$ and $\tilde{x}_s$ are weighted centroids for the clusters r and s. If cluster r was created by combining clusters p and q, $\tilde{x}_r$ is defined recursively as

$$\tilde{x}_r = \frac{1}{2}(\tilde{x}_p + \tilde{x}_q)$$

- Ward's linkage uses the incremental sum of squares; that is, the increase in the total within-cluster sum of squares as a result of joining two clusters. The within-cluster sum of squares is defined as the sum of the squares of the distances between all objects in the cluster and the centroid of the cluster. The equivalent distance is:

$$d^2(r,s) = n_r n_s \frac{|\overline{x}_r - \overline{x}_s|^2}{(n_r + n_s)} \tag{5.12}$$

where $\overline{x}_r$ and $\overline{x}_s$ are the centroids of clusters r and s, as defined in the centroid linkage.

The results of hierarchical clustering are usually presented in a dendrogram. This is a tree-like plot where each step of hierarchical clustering is represented as a fusion of two branches of the tree into a single one. The branches represent clusters obtained on each step of hierarchical clustering. Figure 5.1 shows an example of dendrogram obtained when a hierarchical agglomerative clustering is applied to a synthetic data-set (2-dimensional Gaussian of 50 items) using the complete linkage criterion (see Eq. (5.8)). In this example each leaf of the tree represents a single item.

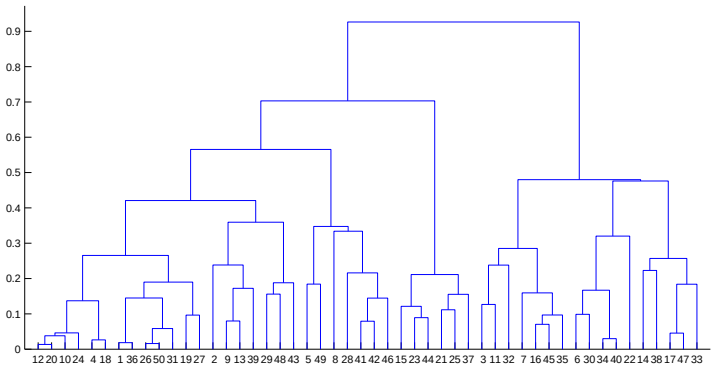

Figure 5.1: Dendrogram example.

The final step of a hierarchical clustering algorithm is the horizontal cut of the dendrogram in order to obtain a suitable partition of the tree in groups of sub-trees which represent the desired data partition. Unfortunately no hierarchical clustering algorithms performs an automatic cut of the dendrogram but to this aim an external criterion or a partially knowledge of the data-set properties must be used in contrast with the nature of the clustering problem.

In an analogous way it is possible to apply a hierarchical clustering to a collection of M clustering solutions $\Gamma = \{\gamma_1 \dots \gamma_M\}$ using the similarity matrix S_M for the distances between clusterings and selecting a linkage criterion. The result of this procedure is the same described for a single clustering but in this case the dendrogram leaves represent the clustering solutions $\Gamma = \{\gamma_1 \dots \gamma_M\}$ instead of the items.

5.4 Algorithm

The proposed methodology, based on the Least-Squares Consensus Clustering, allows to extrapolate in an automatic way a small number of different clustering solutions from an initial (large) set of clusterings obtained by applying any

clustering algorithm to a selected data-set. In the final analysis the aim of the methodology is to find an automatic procedure to cut the dendrogram of clustering solutions in order to obtain a suitable partition of the tree in a group of sub-trees. Each sub-tree will be characterized by a consensus clustering which is one leaf of the same sub-tree and by its quality measure.

Let once again Y be a given data-set of dimensions $N \times D$ and $\Gamma = \{\gamma_1, \ldots, \gamma_M\}$ a collection of $M >> 1$ distinct clustering solutions for Y obtained from any clustering algorithm. The goal is to find a set of L solutions $\gamma_1^*, \ldots, \gamma_L^*$, with $L << M$, which are representative of the solutions in Γ, and to define a measure of quality $E_1, \ldots, E_L$ associated to them.

The algorithm can be summarized in the following steps:

1. Consider a set of M clustering solutions $\Gamma = \{\gamma_1 \ldots \gamma_M\}$ for the selected data-set Y.

2. Calculate the similarity matrix S_M (Eq. (5.6)).

3. Construct a dendrogram using a hierarchical clustering algorithm applied to S_M.

4. For $i = 1, \ldots, M-1$ (number of dendrogram nodes)

   (a) Denote $\Gamma_1 \ldots \Gamma_l$ the groups of solutions (sub-trees) obtained when cutting the tree at the i-th node. Note that only a group can be changed at each step as the dendrogram cut can be realized at each level of leaves aggregation.

   (b) Compute $\gamma_1^* \ldots \gamma_l^*$ as the Least-Squares Consensus clusterings of the sub-sets $\Gamma_1 \ldots \Gamma_l$ using Eq. (5.2).

   (c) Compute the errors $E_{LS}(\Gamma_k)$ (Eq. (5.3)) for each of the sets $\Gamma_k, k = 1 \ldots l$.

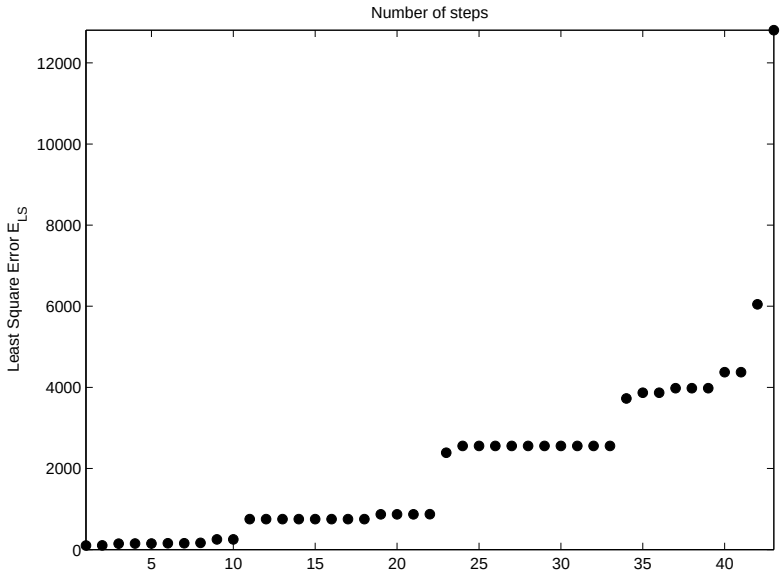

Figure 5.2: Global Least-Squares Error E_{LS} versus the number of steps.

(d) Compute a global error term E^i_{LS} at the i-th node as

$$E^i_{LS} = max(E_{LS}(\Gamma_k), E^{i-1}_{LS}) \tag{5.13}$$

Each Γ_k, $k = 1 \dots l$ can be composed of a leaf that is aggregated to the set of previous solutions or by two sub-groups joined in a node.

5. Construct the plot of the global Least-Squares Error E^i_{LS} versus the number of steps i (see Figure 5.2 for example).

6. Cut the dendrogram on the basis of the behavior obtained for the global Least-Squares Error function (see Section 5.5 for details).

7. Retrieve as Least-Squares Consensus Clusterings $\gamma^*_1 \dots \gamma^*_L$ corresponding to such cut off.

This procedure evaluates at each step a sub-group of clusterings solutions,

their consensus solution and associated errors. The algorithm can be easily implemented either like a top-down or a bottom-up procedure on the tree. The dendrogram cut allows to split the original tree in a certain number of subtrees $\Gamma_1 \dots \Gamma_L$ with $L \ll M$, each representing a group of clusterings with its representative consensus γ_i and its quality measure $E_{LS}(\gamma_i), i = 1 \dots L$.

5.5 Automatic cutoff selection

The plot of the Least-Squares Error versus the number of steps presents some "jumps" as we can see from Figure 5.2 obtained applying the whole procedure to a synthetic data-set described in more details in Chapter 6. These jumps emphasize the aggregation of a single clustering or a group of solutions which are far (in Least-Square sense) from the group of solutions obtained by the previous aggregation. The determination of a threshold value on the plot, in correspondence to one of these jumps, permits to individuate a corresponding cut-off on the dendrogram, putting into evidence several groups of clusterings that are similar. Obviously the threshold selection represents a crucial step of the whole procedure and an automatic selection, rather than a manual and subjective one, is desirable. Many different techniques can be applied to this purpose.

In [Zhu and Ghodsi, 2006] , the authors present an automatic cut-off selection from the scree plot via the use of profile likelihood. Their work places into the context of dimensionality reduction methods and the problem of automatically select the number of coordinates to use for projection in a lower dimension space.

Let $d_1 \geq d_2 \geq \dots \geq d_p \geq 0$ be the ordered coordinates. In the case of Principal Component Analysis (PCA), for example, these are the ordered eigenvalues. If a gap exists at position q, $1 \leq q \leq p$, then $\Sigma_1 = \{d_1, d_2, \dots, d_q\}$

and $\Sigma_2 = \{d_{q+1}, d_{q+2}, \ldots, d_p\}$ can represent samples from two different distributions $f(d, \theta_1)$ and $f(d, \theta_2)$. The log-likelihood function, under the independence assumption, can be written as:

$$l(q, \theta_1, \theta_2) = \sum_{i=1}^{q} \log f(d_i; \theta_1) + \sum_{j=q+1}^{p} \log f(d_j; \theta_2) \tag{5.14}$$

By plugging into the Eq. (5.14) the maximum likelihood estimates (MLE) of θ_1 and θ_2, a profile log-likelihood for q can be written as:

$$l_q(q) = \sum_{i=1}^{q} \log f(d_i; \hat{\theta}_1(q)) + \sum_{j=q+1}^{p} \log f(d_j; \hat{\theta}_2(q)) \tag{5.15}$$

An estimate of q can be obtained by maximizing the profile log-likelihood defined in the Eq. (5.15). To this aim a simple exhaustive search can be used, that is computing $l_q(1), l_q(2), \ldots, l_q(p)$ and estimating q with:

$$\hat{q} = \arg\max \; l_q(k) \; k = 1, 2, \ldots, p \tag{5.16}$$

For simplicity assume f to be the Gaussian distribution:

$$f(d, \mu_j, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} exp\left\{ -\frac{(d - \mu_j)^2}{2\sigma^2} \right\} \; j = 1, 2. \tag{5.17}$$

It is important to use a common scale parameter σ for both $\Sigma_1$and Σ_2. If a different σ is used for each model, it becomes too flexible and it is possible for the profile log-likelihood (Eq. (5.15)) to become infinite, e.g., when $q = 1$ and $q = p - 1$. For completeness, for any given q, the MLEs for $\mu_1$and $\mu_2$are simply the sample averages:

$$\hat{\mu}_1 = \frac{\sum_{d_i \in \Sigma_1} d_i}{q} \quad and \quad \hat{\mu}_2 = \frac{\sum_{d_j \in \Sigma_2} d_j}{p - q} \tag{5.18}$$

and the MLE for the common scale parameter σ^2 is:

$$\hat{\sigma}^2 = \frac{(q-1)s_1^2 + (p-q-1)s_2^2}{p-2}, \tag{5.19}$$

where s_j^2 is the sample variance of Σ_j.

We have applied the procedure to our problem replacing the scree plot with the least-squares error curve and assuming the Gaussian distribution for the profile log-likelihood. This allows to determine a threshold for the dendrogram. The result is a list of groups of solutions (sub-trees) which are all distinct each other and, by construction, do not admit overlaps, but the result is strongly dependent on the hierarchical clustering algorithm used to construct the dendrogram from the similarity matrix.

5.6 Pairwise matrix visualization

As noticed in Section 5.4 the proposed approach applied to a given data-set provides a limited number of solutions $\Gamma_1, \ldots, \Gamma_L$ with $L \ll M$. Generally the end user is called to analyze this restrict group of solutions and the development of a visualization tool to this aim becomes very important to simplify and speed up his work.

Different ways to visualize the consensus clustering results are available in literature. For example in [Monti et al., 2003] the authors proposed a consensus matrix reordering and visualization to help assess the clusters composition and number. In particular, in the range of their work, associating a color gradient to the 0-1 range of real numbers, so that white corresponds to 0, and dark red corresponds to 1, and assuming the matrix is arranged so that items belonging to the same cluster are adjacent to each other , a matrix corresponding to perfect consensus will be displayed as a color-coded heat map characterized by red blocks along the diagonal, on a white background.

For our purposes, in order to have an immediate feedback on the analysis results, we suggest the following graphical visualization. Let Γ_l be one of the solutions and $\hat{\gamma}_{LS}^l$ its corresponding Least-Squares consensus clustering. Without

loss of generality, we can re-arrange the samples in order to place together elements with the same label, so that $\hat{\gamma}^l_{LS}$ contains first all the samples that belong to cluster 1, then the ones in cluster 2, finally the ones in cluster k. The average pairwise probability matrix $\hat{\pi}^l_{ij}$ can be rearranged accordingly. Note that each element (i, j) of this matrix takes the value 1 if the corresponding couple of elements are allocated in the same cluster in all the clusterings of the group Γ_l and takes the value 0 if the corresponding couple of elements are allocated in different clusters in each clustering. All the values included in the range $[0, 1]$ model the intermediate conditions. In this way, associating a color gradient to this range of real numbers, a heat-map of the pairwise probability matrix of each group Γ_i can be displayed. In this heat-map the pixels will represent the data-set elements.

It is easy to see that when the clusterings of the same group are similar (in Least-Square sense), homogeneous blocks appear in the matrix showing that these sets of elements have been clustered always in the same manner. Also this visualization enables to display "how many" and "which" clusters are "mixed", that is to highlight the elements classified in a different way in several clusterings. To improve this type of visualization a further reorganization of the blocks is possible: for each block, the portion of cells with value 1 are arranged in the upper left corner of the block and, using an iterative procedure, all the other portions are arranged in a descending order depending on their values in the range [0,1]. This makes easier to evaluate the homogeneity of each clusters, to detect outliers or possible different assignments but, on the other hand, all the information on the elements position is missed.

Figure 5.3 shows an example of heat-map visualization. In this case it is evident the presence of six different clusters. Two of them (respectively the first and the fifth starting from the upper left corner) are totally homogeneous. Other

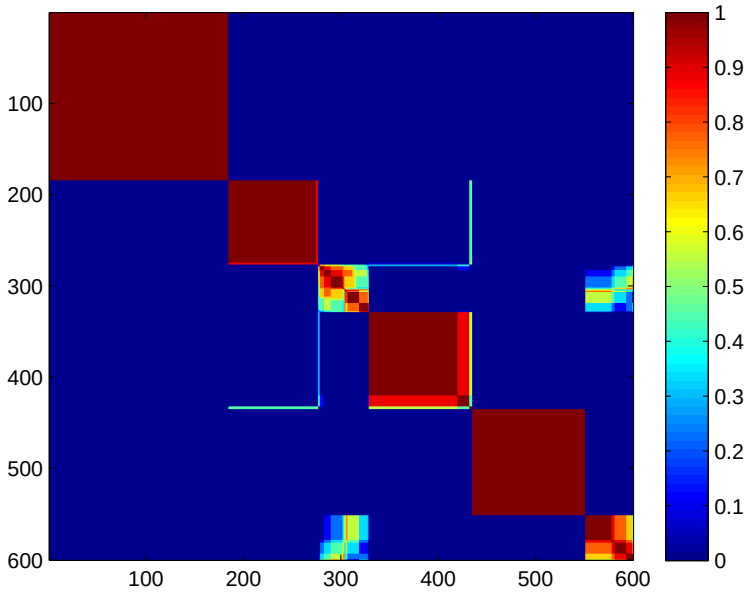

Figure 5.3: Heat-map visualization example.

two (the second and the fourth) are only partially mixed. Finally there are two clusters (the third and the sixth) which are very mixed, that is, their elements are clustered in different ways in the clusterings of the selected sub-tree.

Chapter 6

Results and Analysis

6.1 Experimental setup

To evaluate the performance of our method we have considered several different synthetic and real data-sets and generated the initial set of clustering solutions applying different clustering algorithms to them. All the software we developed runs in the MATLAB environment.

6.1.1 Simulated data

Several synthetic data-sets have been generated to test the procedure described in the Chapter 5. We present here four of the whole sets of experiments performed. Three of these data-sets are composed by mixtures of Gaussians in 2 dimensions with different covariance matrices. The first data-set is composed by a mixture of 5 Gaussians from each of them we have sampled 150 points with a total of N=750 points; the second data-set is composed by a mixture of 6 Gaussians from each of them we have sampled 100 points with a total of N = 600 points; finally the third data-set is composed by a mixture of 7 Gaussians from each of them we have sampled 100 points with a total of N = 700 points The data-sets are shown in Figure 6.1.

The last data-set we consider has an a priori known multi-level hierarchi-

cal structure inspired by the one used in [Bertoni and Valentini, 2008] where the authors proposed a new method based on Bernstein's inequality to assess the statistical significance and to discover multi-level structures in biomolecular data. It is a two-dimensional synthetic data-set with a three level hierarchical structure: at a first level three large clusters are present in the data; at a second level we have six clusters and finally at a third-level twelve clusters may be detected. The data-set is composed by a total of N = 600 points and it is shown in Figure 6.2.

This data-set allows to show the effectiveness and practical utility of our methodology in discovering hidden sub-structures of the data.

In the first set of simulations we have used K-means as clustering algorithm to generate the initial group of clustering solutions for each of the three synthetic data-sets. After 500 runs of K-means we have retained only the 35 distinct clusterings as initial set for the first data-set, only the 44 distinct clusterings for the second data-set and only the 49 distinct clusterings for the third one. For each group of these solutions we have computed the similarity matrix S_M according to Equation 5.6, we have constructed the dendrogram of clustering solutions using the "complete linkage" algorithm, implemented in MATLAB toolbox, and then we have applied the proposed algorithm to the hierarchical tree of the solutions in a bottom-up approach.

Figure 5.2 shows the plot of Least Squares Error of the clusterings set at each step for the data-set composed by 6 Gaussians (second data-set); the figures for the other two data-sets (not shown for brevity reasons) are very similar. Applying the automatic selection procedure to the Least Squares Error curve for this data-set, we have obtained a threshold value in correspondence of the step number i = 33. This step corresponds to a particular node and consequently to a particular cut on the dendrogram. The result of this cut is shown in the central

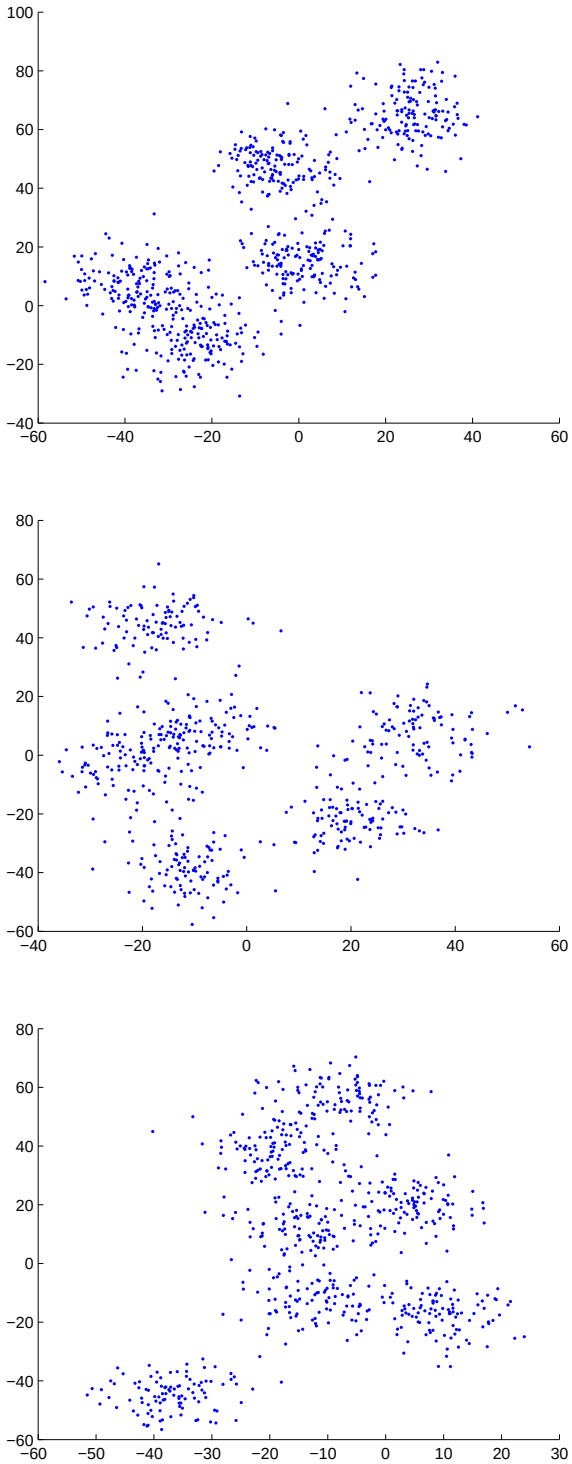

Figure 6.1: Synthetic data-sets composed, respectively, by a mixture of 5, 6 and 7 Gaussians in 2 dimensions.

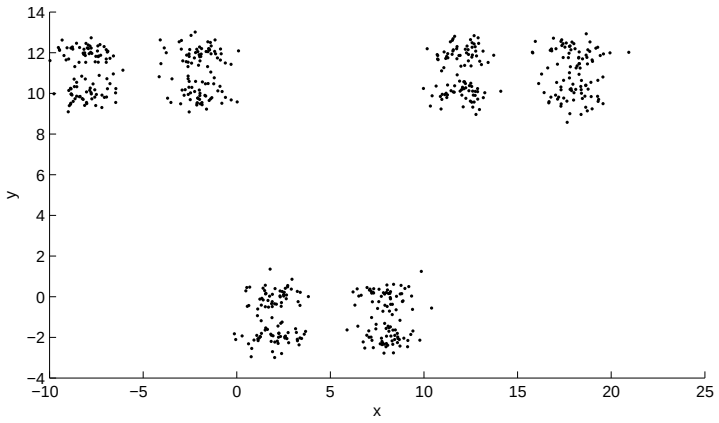

Figure 6.2: Synthetic data set: a three-level hierarchical structure with 3, 6 and 12 clusters in 2 dimensions.

panel of the Figure 6.3: we have highlighted different groups of clusterings using different colors for each group of clusterings solutions Γ_i. The upper and lower panels of the figure show the dendrograms for the data-set one and the data-set three respectively.

Now consider again the second data-set for example (similar considerations can be made for the other data-sets). We have extrapolated a set of 12 solutions (8 groups and 4 singletons) from the initial 44 clusterings. See Table 6.1 for details.

Finally in Figure 6.4 we show the pairwise matrix visualization applied on the group $\Gamma_7 = \{\gamma_{28}, \gamma_{29}, \gamma_{31}, \gamma_{32}, \gamma_{33}, \gamma_{34}, \gamma_{35}, \gamma_{36}, \gamma_{38}\}$ formed by 9 clusterings, which is highlighted in red color in the central pane of Figure 6.3.

Two homogeneous blocks are clearly visible along the diagonal: they represent two clusters whose elements have been clustered together in all the 9 solutions of the group. Other two clusters present only minor mixed regions identifiable in different colors on the border of the blocks. Finally two clusters present less conserved areas and illustrate the situation when some elements can

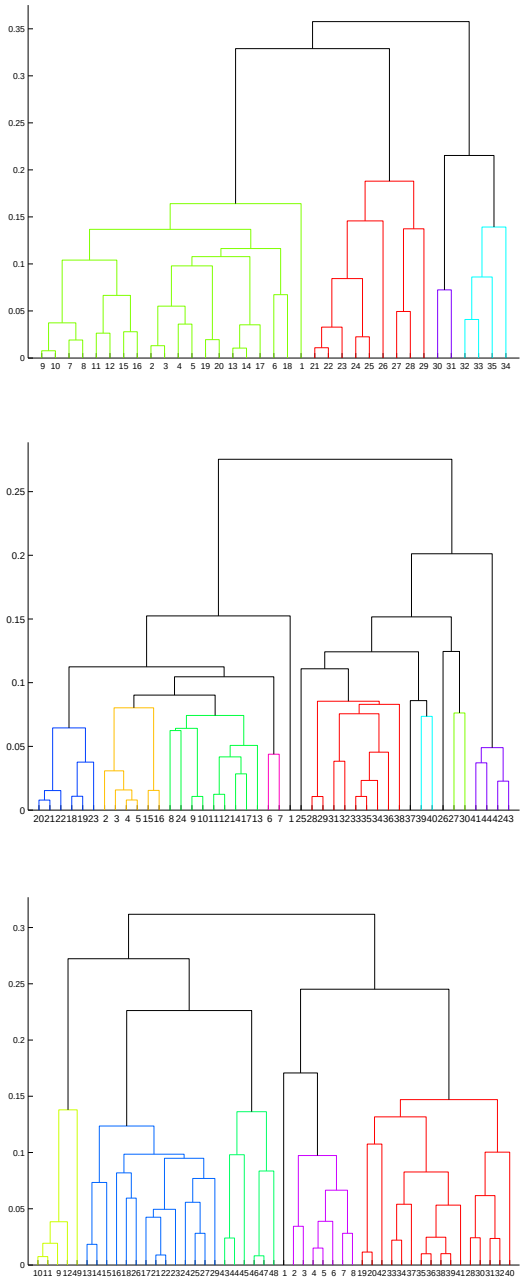

Figure 6.3: Dendrogram of the clustering solutions for the synthetic data-sets. Different colors indicate different groups of aggregated clusterings.

Group	Clusterings
Γ_1	$\gamma_{18}, \gamma_{19}, \gamma_{20}, \gamma_{21}, \gamma_{22}, \gamma_{23}$
Γ_2	$\gamma_2, \gamma_3, \gamma_4, \gamma_5, \gamma_{15}, \gamma_{16}$
Γ_3	$\gamma_8, \gamma_9, \gamma_{10}, \gamma_{11}, \gamma_{12}, \gamma_{13}, \gamma_{14}, \gamma_{17}, \gamma_{24}$
Γ_4	γ_6, γ_7
Γ_5	γ_1
Γ_6	γ_{25}
Γ_7	$\gamma_{28}, \gamma_{29}, \gamma_{31}, \gamma_{32}, \gamma_{33}, \gamma_{34}, \gamma_{35}, \gamma_{36}, \gamma_{38}$
Γ_8	γ_{37}
Γ_9	γ_{39}, γ_{40}
Γ_{10}	γ_{26}
Γ_{11}	γ_{27}, γ_{30}
Γ_{12}	$\gamma_{41}, \gamma_{42}, \gamma_{43}, \gamma_{44}$

Table 6.1: Groups of clustering solutions obtained by the dendrogram cut for the second synthetic data-set.

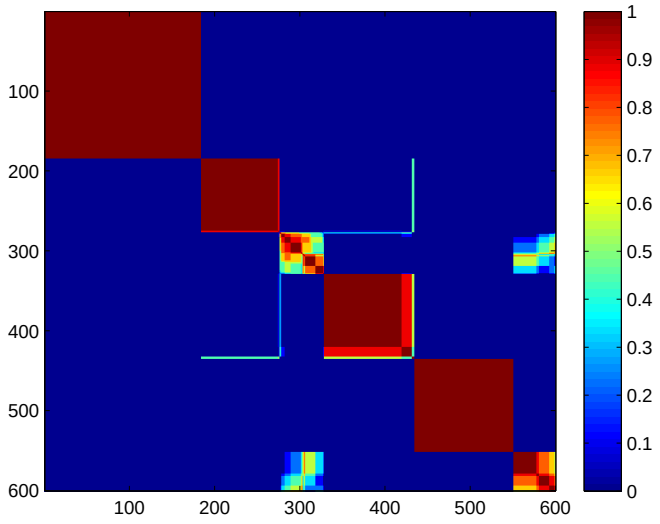

Figure 6.4: Pairwise matrix visualization for the group of clusterings $\Gamma_7 = \{\gamma_{28}, \gamma_{29}, \gamma_{31}, \gamma_{32}, \gamma_{33}, \gamma_{34}, \gamma_{35}, \gamma_{36}, \gamma_{38}\}$ obtained applying the algorithm to the second synthetic data-set.

be clustered either with one or the other cluster.

We have repeated the study with

1. Γ generated by K-means with different values of k in a certain range of values.

2. Γgenerated by EM algorithm with assigned value of k.

3. Γgenerated by EM algorithm with different values of k in a certain range of values.

4. Γgenerated by both K-means and EM algorithms.

In all these cases the experimental results are very similar to those shown in the presented example.

In order to show the capability of the proposed approach to detect a multi level structure present in a data-set we also present the results obtained when the whole procedure is applied to the synthetic data-set described in Figure 6.2. In the first set of simulations we have used K-means as clustering algorithm to generate the initial group of clustering solutions. Clearly when K-means runs with k = 3 clusters our procedure results unnecessary as the 3 clusters of the first level structure are well separated. Instead we obtained very interesting results after 500 runs of K-means with k = 6 clusters. In this case we have retained only the 26 distinct clusterings as initial set Γ. For these solutions we have computed the similarity matrix S_M according to Equation 5.6, we have constructed the dendrogram of clustering solutions using the "complete linkage" and then we have applied the proposed algorithm to the hierarchical tree of the 26 solutions in a bottom-up approach.

Figure 6.5 shows the plot of Least Squares Error of the clusterings set at each step. Applying the automatic selection procedure to the Least Squares Error curve, we have obtained a threshold value in correspondence of the step

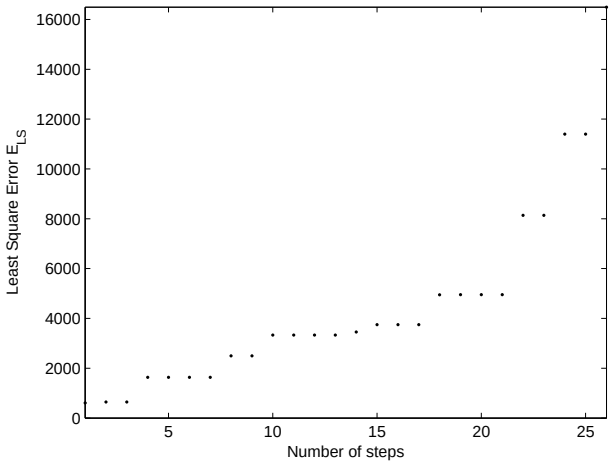

Figure 6.5: Least Square Error of the clusterings set at each step of the Algorithm for the synthetic data-set with a multi-level hierarchical structure.

Group	Clusterings	Consensus
Γ_1	γ_{24}, γ_{26}	γ_{24}
Γ_2	$\gamma_{19}, \gamma_{22}, \gamma_{27}$	γ_{27}
Γ_3	$\gamma_6, \gamma_7, \gamma_{10}, \gamma_{12}$	γ_6
Γ_4	$\gamma_{17}, \gamma_{20}, \gamma_{23}$	γ_{23}
Γ_5	$\gamma_2, \gamma_3, \gamma_5, \gamma_9, \gamma_{11}, \gamma_{15}$	γ_{15}
Γ_6	$\gamma_1, \gamma_4, \gamma_8, \gamma_{13}, \gamma_{14}, \gamma_{16}$	γ_{16}
Γ_7	$\gamma_{18}, \gamma_{21}, \gamma_{25}$	γ_{18}

Table 6.2: Groups of clustering solutions obtained by the dendrogram cut for the synthetic data-set with the multi-level hierarchical structure.

number i = 21. This step corresponds to a particular node and consequently to a particular cut on the dendrogram. The result of this cut is shown in Figure 6.6. We have extrapolated a set of 7 solutions from the initial 26 clusterings. See Table 6.2 for details.

Finally in Figure 6.7 and Figure 6.8 we show the pairwise matrix visualization applied on all the groups of clusterings obtained from the procedure application. For each group is also visualized the scatter plot of the corresponding Least Square consensus clustering of the group. A rapid analysis of these

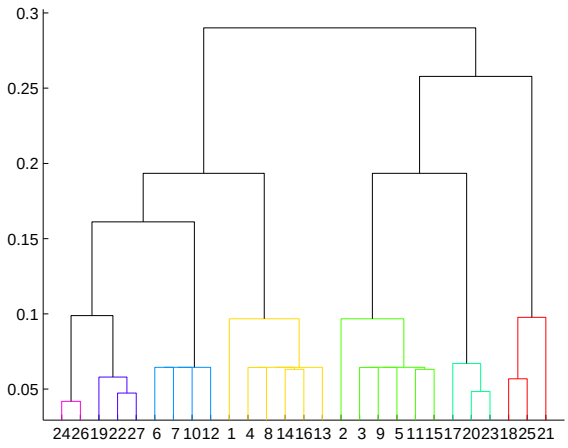

Figure 6.6: Dendrogram of the clustering solutions for the synthetic data-set with the multi-level hierarchical structure.

plots allow the user to have a complete vision of the data structure. In fact each group of clusterings emphasizes a sub-structure: one or two clusters of the first level structure are detected and the remaining one or two are partitioned in two or more clusters detecting the second and third level structure (six and twelve clusterings respectively) embedded in the selected data-set. Moreover a further look to the pairwise visualization shows that the blocks on the diagonal are quite homogeneous for the majority of the groups, that is the clusterings belonging to the same group are very similar each other.

We stress that a single run of K-means applied to this data-set gives only one (random) of this clustering solutions hiding the natural structure of the data-set. On the other hands multiple running of K-means gives a lot of solutions which result very hard to analyze. Our result represents a good trade-off between the two situations providing the minimum number of solutions to analyze to have a general understanding of the data structure.

Finally we have applied the procedure to the same data-set with k = 12

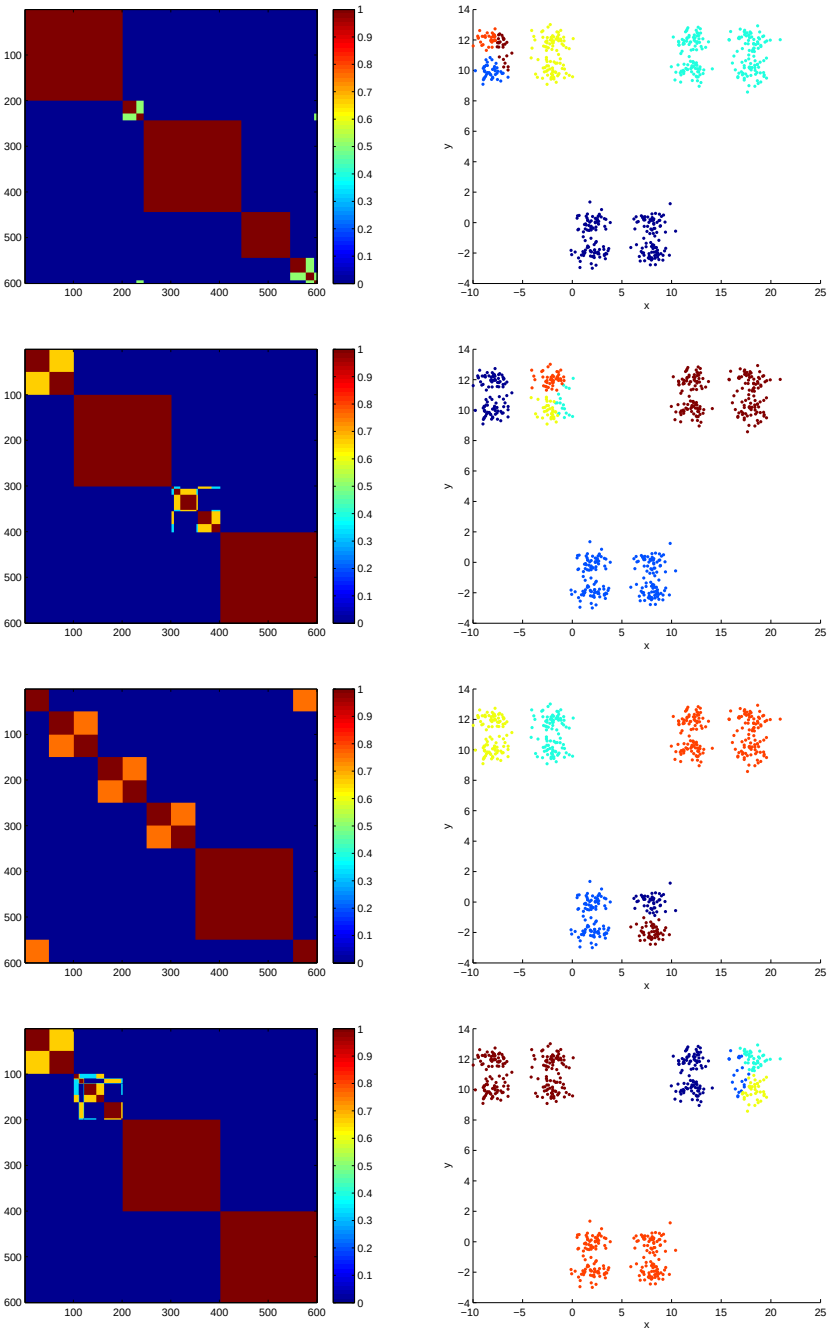

Figure 6.7: Pairwise matrix visualization for the groups of clusterings Γ_1,Γ_2,Γ_3 and Γ_4 (starting from the upper panel) obtained applying the algorithm to the synthetic data-set with the multi-level hierarchical structure.

(that is the real classification). Considering 500 runs of K-means we have obtained 380 different clustering solutions which the proposed algorithm reduced to 80 groups with their Least Squares consensus clusterings. It is important to notice that only few of the K-means solutions are similar to the real classification and it seems very unlikely to obtain one of them with a single run of K-means. Nevertheless our approach is able to show not only these solutions near to the real classification but also other groups of clusterings which highlight the hierarchical structure of the considered data-set.

6.1.2 Real data-set

As real data set we have chosen the well known Leukemia data-set [Golub, 1999]. It is composed by a group of 25 acute myeloid leukemia (AML) samples and another group of 47 acute lymphoblastic leukemia (ALL) samples, that can be subdivided into 38 B-Cell and 9 T-Cell subgroups, resulting in a two-level hierarchical structure.

We have applied the procedure described in Section 6.1.1 with all its variants drawn in Points 1.-4.: no significant differences have been obtained. For this reason we have focalized our attention on the K-means algorithm. As for the synthetic data-sets, we have considered 500 runs to obtain the starting set of clustering solutions Γ. For the sake of brevity we report only the experiment with k = 3. In this case Γ consists of M = 12 different solutions. We have computed the similarity matrix S_M and we have built the hierarchical tree using the complete linkage, then we have applied the algorithm. Applying the automatic selection procedure to the Least Squares Error curve we have obtained a threshold value in correspondence of the step number i = 8.

The result of the subsequent dendrogram cut is shown in Figure 6.9: we have highlighted different groups of clusterings using different colors for each group and extrapolated a set of 5 solutions from the starting 12 clusterings. Figure

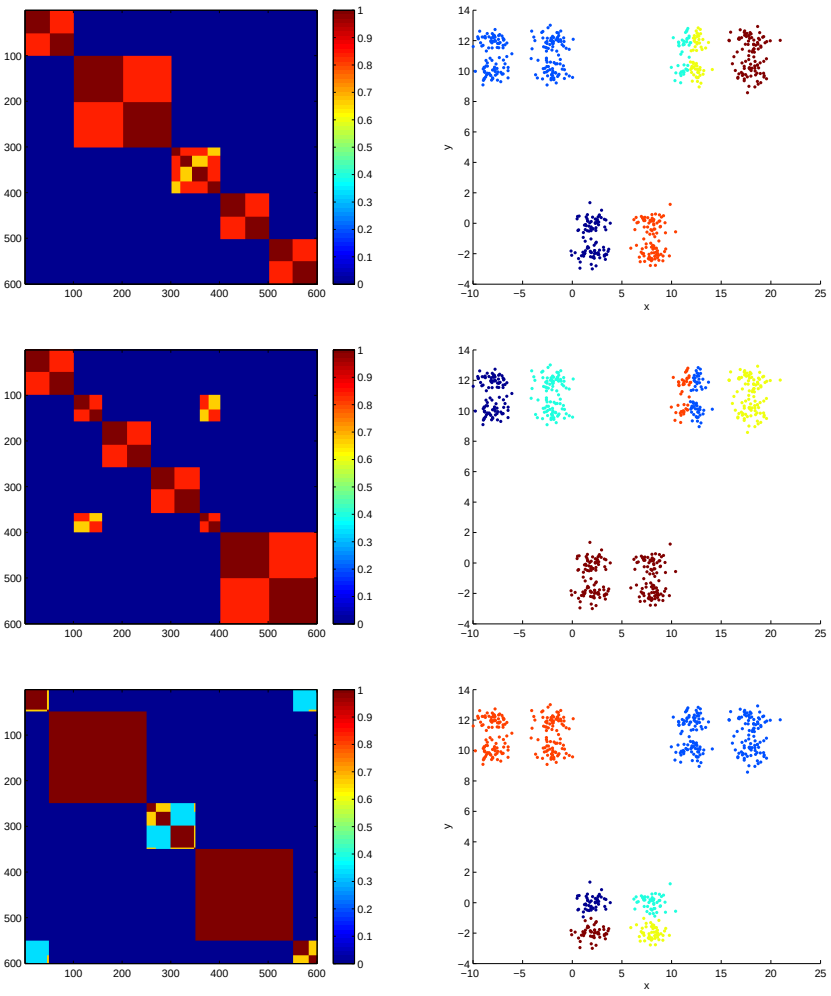

Figure 6.8: Pairwise matrix visualization for the groups of clusterings Γ_5, Γ_6 and Γ_7 (upper, central and lower panel respectively) obtained applying the algorithm to the synthetic data-set with the multi-level hierarchical structure.

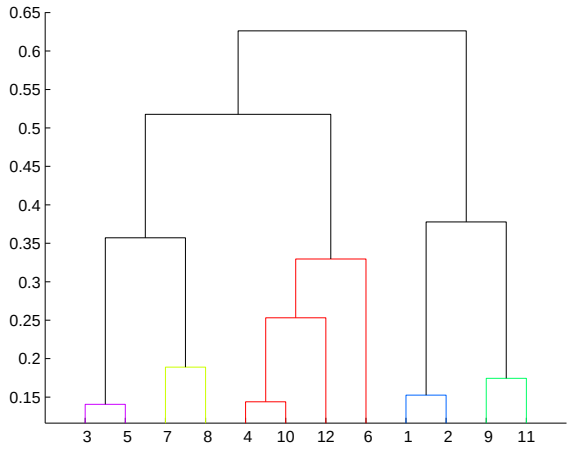

Figure 6.9: Dendrogram of the clustering solutions for the real data-set. Different colors indicate different groups of aggregated clusterings.

6.10 shows the pairwise matrix visualization applied on all the groups of the obtained clusterings. Comparing the Least Square consensus clusterings of each group (γ_3, γ_7, γ_4, γ_1 and γ_9, respectively) with the known data classification it is clear that only γ_1 separates the data in the right manner.

On the other hand the clusterings γ_3, γ_7, γ_4 and γ_9 separate the AML elements from the ALL ones which are divided in different ways displaying a sub level structure which is actually present in the data.

6.2 Method robustness

In order to have a preliminary idea of the robustness of the proposed procedure we have carried out a series of tests on the same synthetic data-sets.

For brevity reasons we report here only the tests performed on the data-set composed of 6 Gaussians. The experiments have been organized in the following way:

1. We have carried out a set of 100 experiments applying K-means algorithm

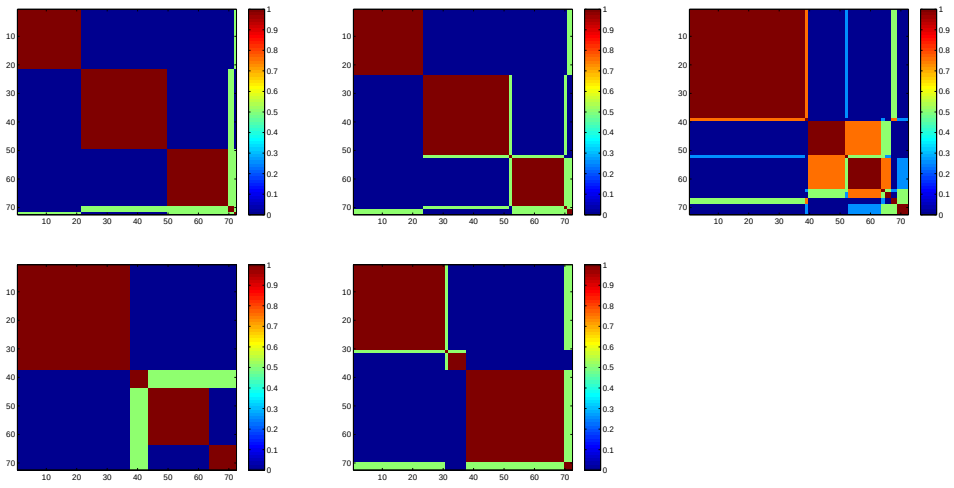

Figure 6.10: Pairwise matrix visualization for the 5 groups of clusterings obtained applying the algorithm to the real data-set. Starting from the upper panel on the left, the plots are referred to the clusterings groups with γ_3, γ_7, γ_4, γ_1 and γ_9 as consensus clustering respectively.

with fixed value of k to generate the starting ensemble of clustering solutions Γ. In each experiment we have run K-means 500 times on the data-set with k = 6 and we have considered only the different solutions.

2. We have carried out a set of 100 experiments applying K-means algorithm with k variable in a certain range to generate the starting ensemble of clustering solutions Γ. In each experiment we have run K-means 100 times on the data-set with k = 3, 100 times with k = 4, 100 times with k = 5 and 100 times with k = 6.

3. We have carried out a set of 100 experiments applying EM algorithm with fixed value of k to generate the starting ensemble of clustering solutions Γ. In each experiment we have run EM 500 times on the data-set with k = 6 and we have considered only the different solutions.

For each experiment our method, as a consequence of the data-dependent den-

drogram cut, can provide a different number L of representative clustering solutions. It is clear that the procedure results robust if the number L is the same for each experiment or, at least, it is constrained in a narrow range and the clustering solutions are not too much different each other.

Figure 6.11 summarizes the results for the 3 experiments. The upper panel shows the histogram of the number L of solutions for the first set of experiments. We observe that for 70 experiments the dendrogram cut gives 4 representative solutions. Another less pronounced peak in the histogram is clear for $L = 12$. In conclusion the procedure appears robust at a first approximation. The central panel shows the histogram of the number L of solutions for the second set of experiments. Again the procedure appears robust even though, as expected, the number of solutions is going to be larger than the previous case. The lower panel shows the histogram of the number L of solutions for the third set of experiments. The histogram appears very similar to the first case (K-means algorithm).

Obviously this robustness analysis is only a preliminary one as it is limited to the control of the number of different solutions obtained with the repeated experiments and not to the control of the single solution obtained but it gives a fast feed-back to the proposed problem opening new perspectives for further investigations.

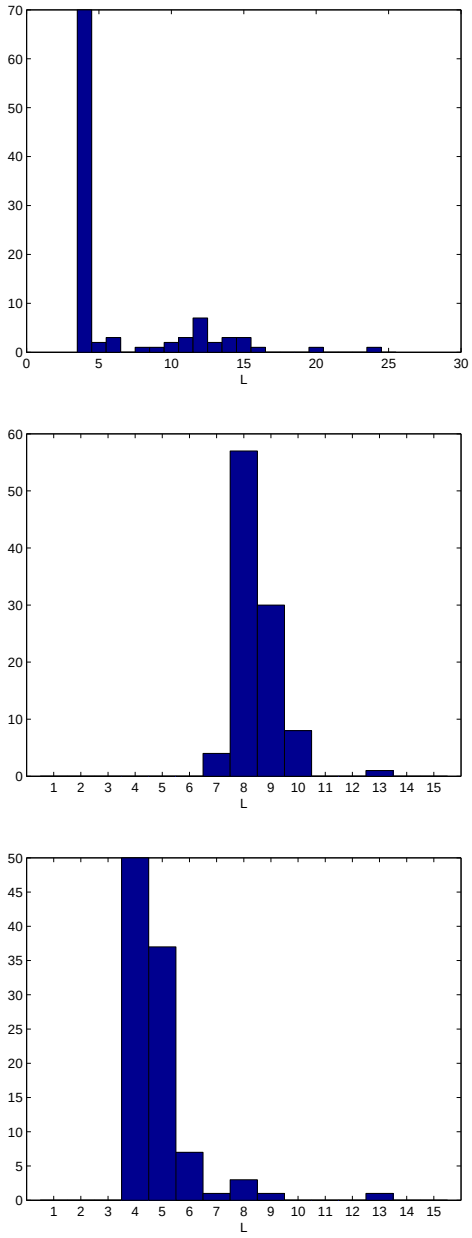

Figure 6.11: Upper panel: histogram of the number L of solutions obtained from the first set of experiments (K-means algorithm with fixed value of k); Central panel: Histogram of the number L of solutions obtained from the second set of experiments (K-means algorithm with variable values of k); Lower panel: Histogram of the number L of solutions obtained from the third set of experiments (EM algorithm with fixed value of k).

Chapter 7

Conclusions and Future Work

In this thesis two data mining problems related to the management of high dimensional data have been addressed. The first one deals with cloud detection, a problem of multispectral satellite image classification, demonstrating the high reliability of the statistical techniques of discriminant analysis in classifying this type of images. The second application addresses the need to handle high dimensional data (as biological data for example) for which it is necessary to find significant structures within them.

In details, the first application demonstrated very good feasibility of statistical (supervised) discriminant analysis in detecting cloud mask over a Western European area from multispectral remotely sensed images taken from radiometers on board geostationary satellites, precisely SEVIRI on board MSG. Reliability of cloud detection ranged from good to excellent in all analyzed conditions (over land, water, on daytime and nighttime). This result was achieved resorting to some mathematical tools (namely, Principal and Independent Component Analysis and nonparametric density estimation) able to exploit multispectral character of the sensor at best and to fix some theoretical issues intrinsic with multivariate data analysis. The method can be considered fully integrated physical/statistical, where the link with physics is guaranteed by the use of a very

consolidated cloud mask to train the (statistical) discriminant analysis. By its very nature it can be considered as a valid alternative for sensors having cloud masks not consolidated yet and as a way to develop cloud masks of new sensors in a quite fast time.

Several points have to be addressed in order to improve accuracy of the cloud mask further and, especially, to extend it to the full disk (i.e., all latitudes and longitudes of the hemisphere looked at by the geostationary satellite) and to the whole day. First of all robustness of the cloud mask has to be evaluated with respect to the "true" cloud mask used for the training of the discriminant analysis (in the present work product MOD35 based on the MODIS sensor): even though same robustness is guaranteed by the statistical character of the method and by choosing only pixels estimated confidently clear or cloudy for the training phase, however particular conditions, as light clouds, could deserve more attention. Better spatial classification can be obtained by using also the High Resolution Visible (HRV) channel. Further improvement of the methodology can be obtained by resorting on classification tools region-based rather than pixels-based: actually clouds naturally have an intrinsic spatial correlation that is transferred into the image.

With regard to the second application, we have investigated the multiple clustering solution problem. Our work differs from the classical consensus clustering approach as it relies on the belief that a single optimal solution for a clustering problem does not exist and it is often more desirable to provide a limited number of different "good" solutions.

To this purpose we have proposed a consensus clustering algorithm called Least-Squares Consensus Clustering which extends the idea of the Least-Squares Clustering and allows to extrapolate in an automatic way a small number of different clustering solutions from an initial (large) set of clusterings obtained by

applying any clustering algorithm to a selected data-set. We have also defined a measure of quality in terms of Least-Squares Error and, in order to have an immediate feedback on the analysis results, we have suggested a graphical visualization of the obtained solutions. The developed methodology is completely automatic and totally independent from the methods used for the generation of the initial clusterings ensemble.

We have illustrated the motivation, the practical utility and the performance of the proposed method using both simulated and real data. In all the experiments the algorithm allows to discover the multi level patterns hidden in the data providing the minimum number of clustering solutions to analyze to have a global understanding of the data structure.

Even if the proposed approach is user-independent, a drawback of the procedure is, of course, its dependence on the hierarchical clustering algorithm used to construct the dendrogram. To this end our future work will be dedicated to overcome this disadvantage using, for example, other clustering algorithms. Moreover we will also investigate the stability of the clustering solutions.

Bibliography

[Ackerman et al., 1998] Ackerman, S.A., Strabala, K.I., Menzel, W.P., Frey, R.A., Moeller, C.C., Gumley, L.E. Discriminating clear-sky from clouds with MODIS. Journal of Geophysics Research Atmospheres, 103, 32141-32157, 1998.

[Aly, 2005] Aly, M. Survey on Multi-Class Classification Methods. Technical Report, Caltech, USA, 2005.

[Amato et al., 2003] Amato, U., Antoniadis, A., Gregoire, G. . Independent Component Discriminant Analysis. International Journal of Mathematics, 3, 735-753, 2003.

[Amato et al., 2008] Amato, U., Antoniadis, A., Cuomo, V., Cutillo, L., Franzese, M., Murino, L., Serio, C. Statistical cloud detection from SEVIRI multispectral images. Remote Sensing Of Environment, Elsevier editor. Vol. 112, 750-766, 2008.

[Ameur et al., 2004] Ameur, Z., Ameur, S., Adane, A., Sauvageot, H., & Bara, K. Cloud classification using the textural features of Meteosat images. International Journal of Remote Sensing, 25, 4491—4503, 2004.

[Anderson, 1984] Anderson, T.W. An introduction to multivariate statistical analysis, 2nd Ed. New York: John Wiley & Sons, 1984.

[Ball and Hall, 1965] Ball, G., & Hall, D. ISODATA, a novel method of data analysis and pattern classification. Tech. rept. AD 699616. Stanford Research Institute, Stanford, CA, 1965.

[Barthlemy and Leclerc, 1995] Barthlemy, J.P., Leclerc, B. The median procedure for partitions. I.J. Cox, P. Hansen, B. Julesz (Eds.), Partitioning Data Sets, American Mathematical Society, Providence, RI, 3-34, 1995.

[Baum et al., 1997] Baum, B. A., Tovinkere, V., Titlow, J., &Welch, R. M. Automated cloud classification of global AVHRR data using a fuzzy logic approach. Journal of Applied Meteorology, 36, 1519—1540, 1997.

[Bay, 1998] Bay Stephen D. Combining nearest neighbor classifiers through multiple feature subsets. In Proceedings of the 17th International Conference on Machine Learning, 37–45, Madison, WI, 1998.

[Ben-Hur et al., 2002] Ben-Hur, A., Elisseeff, A. and Guyon, I. A Stability Based Method for Discovering Structure in Clustered Data. Pacific Symposium on Biocomputing 2002, Vol. 7. Lihue, Hawaii, 6–17, 2002.

[Bertoni and Valentini, 2007] Bertoni, A., Valentini, G. Model order selection for biomolecular data clustering, BMC Bioinformatics, vol.8, Suppl.3, 2007.

[Bertoni and Valentini, 2008] Bertoni, A., Valentini, G. Discovering multi-level structures in biomolecular data through the Bernstein inequality. BMC Bioinformatics 9(Suppl 2):S4 , 2008.

[Bezdek, 1981] Bezdek, J. C. Pattern Recognition with Fuzzy Objective Function Algorithms. New York, 1981.

[Bhattacharjee et al., 2001] Bhattacharjee, A., Richards, W. G., Staunton, J., Li, C., Monti, S., Vasa, P., Ladd, C., Beheshti, J., Bueno, R., Gillette, M., Loda, M., Weber, G., Mark, E. J., Lander, E. S., Wong, W., Johnson, B. E., Golub, T. R., Sugarbaker, D. J. and Meyerson, M. Classification of Human Lung Carcinomas by mRNA Expression Profiling Reveals Distinct Adenocarcinomas Subclasses. Proceedings of the National Academy of Sciences 98(24), 13790–13795, 2001.

[Bishehsari et al., 2007] Bishehsari, F., Mahdavinia, M., Malekzadeh, R., Mariani-Costantini, R., Miele, G., Napolitano, F., Raiconi, G., Tagliaferri, R., Verginelli, F.. PCA based feature selection applied to the analysis of the international variation in diet. Lecture Notes in Artificial Intelligence 4578, 551-556 , 2007.

[Blei et al., 2003] Blei, D. M., Ng, A. Y., & Jordan, M. I. Latent dirichlet allocation. Journal of machine learning research, 3, 993–1022, 2003.

[Breiman et al., 1984] Breiman, L., Friedman, J., Olshen, R. A. and Stone, C. J. Classification and Regression Trees. Chapman and Hall, 1984.

[Caruana et al., 2006] Caruana, R., Elhawary, M., Nguyen, N. and Smith, C. Meta Clustering. The Proceedings of the Sixth International Conference on Data Mining (ICDM), December 2006.

[Comon, 1984] Comon, P. Independent Component Analysis, a new concept. Signal Processing, 36, 287-314, 1984.

[Cortes and Vapnik, 1995] Cortes C. and Vapnik V. Support-vector networks. Machine Learning, 273–297, 1995.

[Cutillo and Amato, 2006] Cutillo, L., Amato, U. Localized empirical discriminant analysis. Computational Statistics & Data Analysis, vol. 52, 4966- 4978, 2008.

[Dahl, 2006] Dahl, D. B. Model-Based Clustering for Expression Data via a Dirichlet Process Mixture Model, in Bayesian Inference for Gene Expression and Proteomics, in : Kim-Anh Do, Peter Mller, Marina Vannucci (Eds.), Cambridge University Press, 201-218, , 2006.

[Damman and Mueller, 2006] Damman, K., Mueller, J. MSG Level 1.5 Image Data Format Description. Technical Report EUM/MSG/ICD/105, (2006).

[Dempster et al.,1977] Dempster, A. P., Laird, N.M., Rubin, D.B. Maximum likelihood from incomplete data via the EM algorithm. J. Roy. Statist. Spc. 39, 1-38, 1977.

[Derrien and Le Gleau, 2005] Derrien, M., & Le Gleau, H. MSG/SEVIRI cloud mask and type from SAFNWC. International Journal of Remote Sensing, 26, 4707—4732, 2005.

[Donoho, 2000] Donoho, D. High-Dimensional Data Analysis: The Curses and Blessings of Dimensionality. AMS Math Challenges Lecture, 2000.

[Duda et al., 2001] Duda, R., Hart, P., & Stork, D. Pattern classification. 2 edn. New York: John Wiley & Sons, 2001.

[Dudoit and Fridlyand, 2002] Dudoit, S. and Fridlyand, J.A Prediction-based Resampling Method for Estimating the Number of Clusters in a Dataset. Genome Biology 3(7), 1–21, 2002.

[Falcone and Azimi-Sadjadi, 2005] Falcone, A. K., & Azimi-Sadjadi, M. R. Multi-satellite cloud product generation over land and ocean using canonical coordinate features. MTS/IEEE Oceans Conference, September 18-23, Washington, DC., 2005.

[Fern and Brodley, 2004] Fern, X. Z. and Brodley, C. E. Solving cluster ensemble problems by bipartite graph partitioning. Proceedings of the Twenty-first International Conference on Machine Learning. ACM Press, 2004.

[Fred and Jain, 2002] Ana L.N. Fred, Anil K. Jain, Combining Multiple Clusterings Using Evidence Accumulation. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 27, no. 6, 835-850, June 2005.

[Gelatt et al., 1983] Gelatt, C., Kirkpatrick, S. and Vecchi, M. Optimization by simulated annealing. 220:671-680, 1983.

[Ghosh et al., 2003] Ghosh, A., Pal, N. R., & Das, J. Fuzzy rule based approaches for cloud cover estimation using Meteosat 5 images. Geoscience and Remote Sensing Symposium, 2003 IGARSS '03. Proceedings, Vol. 6, 3438−3440, 2003.

[Gionis et al., 2007] Gionis, A., Mannila, H., and Tsaparas, P. Clustering aggregation. ACM Trans. Knowl. Discov. Data. article 4, 2007.

[Golub, 1999] Golub, T. et al. Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring. Science 286, 531-537, 1999.

[Groves and Bajcsy, 2003] Groves, P., Bajcsy, P. Methodology for hyperspectral band and classification model selection. In 2003 IEEE Workshop on Advances in Techniques for Analysis of Remotely Sensed Data, 120-128, 2003.

[Gu et al., 1989] Gu, Z. Q., Duncan, C. N., Renshaw, E., Mugglestone, M. A., Cowan, C. F. N., & Grant, P. M. Comparison of techniques for measuring cloud texture in remotely sensed satellite meteorological image data. Radar and Signal Processing, IEE Proceedings F, 136, 236−248, 1989.

[Han et al., 2006] Han, B., Kang, L., & Song, H. A fast cloud detection approach by integration of image segmentation

and Support Vector Machine. In J.Wang, Z. Yi, J. M. Zurada, B. L. Lu, & Y. Hujun (Eds.), Advances in Neural Networks, Third International Symposium on Neural Networks, ISNN 2006, Part IIILecture Notes in Computer Science, Vol. 3973, 1210—1215, Berlin Heidelberg: Springer-Verlag, 2006 .

[Heyer et al., 1999] Heyer, L.J., Kruglyak, S. and Yooseph, S., Exploring Expression Data: Identification and Analysis of Coexpressed Genes, Genome Research 9:1106-1115, 1999.

[Hubert and Arabie, 1985] Hubert L. and Arabie P., Comparing partitions. Journal of Classification, 2:193–218, 1985.

[Hyvärinen, 1997] Hyvärinen, A. Independent component analysis by minimization of mutual information, In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP'97, 3917-3920, 1997.

[Hyvärinen, 1999] Hyvärinen, A. Fast and robust fixed-point algorithms for Independent Component Analysis, IEEE Transactions on Neural Networks, 10, 626-634, 1999.

[Jain et al., 2004] Jain, A. K., Topchy, A., Law, M. H. C., & Buhmann, J. M. Landscape of clustering algorithms. Proceedings of the International Conference on Pattern Recognition, vol. 1, 260–263, 2004.

[Jain, 2009] Jain, A.K. Data clustering: 50 years beyond K-means. Pattern Recognition Letters Volume 31, Issue 8, 651-666, 1 June 2010.

[Johnson, 1967] Johnson, S. C. Hierarchical Clustering Schemes Psychometrika, 2:241-254, 1967.

[Kohonen , 2001] Kohonen, T. Self-Organizing Maps, Springer Series in Information Sciences, Third, extended edition, 2001.

[Kotsiantis, 2007] Kotsiantis, S. B. Supervised Machine Learning: A Review of Classification Techniques ,Informatica, Vol. 31, No. 3, 2007.

[Lee et al., 2004] Lee, Y., Wahba, G., & Ackerman, S. Classification of satellite radiance data by Multicategory Support Vector Machines. Journal of Atmospheric and Oceanic Technology, 21, 159—169, 2004.

[Li et al., 2003] Li, J., Menzel, W.P., Yang, Z., Frey, R.A., Ackerman S.A. High-spatial-resolution surface and cloud-type classification from MODIS multispectral band measurements. Journal of Applied Meteorology, 42, 204-226, 2003.

[Li and McCallum, 2006] Li, W., & McCallum, A. Pachinko allocation: Dagstructured mixture models of topic correlations. Proceedings of the 23rd International Conference on Machine Learning, 577—584, 2006.

[Liu et al., 2004] Liu, Y., Key, J.R., Frey, R.A., Ackerman, S.A., Menzel W.P. Nighttime polar cloud detection with MODIS, Remote Sensing of Environment, 92, 181-194, 2004.

[Lu et al., 2004a] Yi Lu, Shiyong Lu, Farshad Fotouhi, Youping Deng, and Susan Brown, FGKA: A Fast Genetic K-means

Algorithm, in Proc. of the 19th ACM Symposium on Applied Computing, 162-163, Nicosia, Cyprus, March, 2004.

[Lu et al., 2004b] Yi Lu, Shiyong Lu, Farshad Fotouhi, Youping Deng, and Susan Brown, Incremental Genetic K-means Algorithm and its Application in Gene Expression Data Analysis, BMC Bioinformatics, 5(172), 2004.

[Lutz, 1999] Lutz, H. J. Cloud processing for METEOSAT Second Generation. Technical Memorandum EUMETSAT No. 4, 1999.

[Macias-Macias et al., 2004] Macías-Macías, M., García-Orellana, C. J., González-Velasco, H. M., Gallardo- Caballero, R., & Serrano-Pérez, A. A comparison of PCA and GA selected features for cloud field classification. In F. J.Garijo, J. C.Riquelme, C. Jos, & M. Toro (Eds.), Advances in Artificial Intelligence — IBERAMIA 2002 8th Ibero-American Conference on AILecture Notes in Computer Science, Vol. 2527, 42—49, Berlin Heidelberg: Springer-Verlag, 2002.

[MacQueen, 1967] MacQueen, J. Some methods for classification and analysis of multivariate observations. Fifth Berkeley Symposium on Mathematics, Statistics and Probability. University of California Press, 281–297, 1967.

[Markovitch and Rosenstein, 2002] Markovitch S. & Rosenstein D. Feature Generation Using General Construction Functions, Machine Learning 49: 59-98, 2002.

[McLachlan and Basford, 1988] McLachlan GL, Basford KE. Mixture Models: Inference and Applications to Clustering New York: Marcel Dekker, 1988.

[Medvedovic and Sivaganesan, 2002] Medvedovic, M. and Sivaganesan, S. Bayesian infinite mixture model based clustering of gene expression profiles. Bioinformatrics, 18, 1194–1206, 2002.

[Monti et al., 2003] Monti, S., Tamayo, P., Mesirov, J., Golub, T. Consensus Clustering: A Resampling-Based Method for Class Discovery and Visualization of Gene Expression Microarray Data, Machine Learning, v.52 n.1-2, 91-118, July-August 2003.

[Platnick et al., 2003] Platnick, S., King, M.D., Ackerman, S.A., Menzel, W.P., Baum, B.A., Ridi, J.C., Frey, R.A. The MODIS cloud products: Algorithms and examples from Terra. IEEE Transactions on Geoscience and Remote Sensing, 41, 459-473, 2003.

[Punera and Ghosh, 2008] Punera, K. and Ghosh, J. Consensus-based ensembles of soft clusterings. Applied Artificial Intelligence, 22:7, 780 - 810, 2008.

[Rish, 2001] Rish I. An empirical study of the naive bayes classifier. In IJCAI Workshop on Empirical Methods in Artificial Intelligence, 2001.

[Salomonson et al., 1998] Salomonson, V. V., Barnes, W. L., Maymon, P. W., Montgomery, H. E., & Ostrow, H. MODIS advanced

facility instrument for studies of the Earth as a system. IEEE Transactions on Geoscience and Remote Sensing, 27, 145−153, 1998.

[Shi et al., 2007] Shi, T., Clothiaux, E.E., Yu, B., Braverman, A.J., Groff, G.N. Detection of daytime arctic clouds using MISR and MODIS Data. Remote Sensing of Environment, 107, 172-184, 2007.

[Stephanidis et al., 1995] Stephanidis, C. N., Cracknell, A. P., & Hayes, L. W. B. The implementation of self organised neural networks for cloud classification in digital satellite images. Quantitetive Remote Sensing for Science and Applications, Geoscience and Remote Sensing Symposium — IGARSS '95, Vol. 1, 455−457,1995.

[Strehl and Ghosh, 2002] Strehl, A. and Ghosh, J. Cluster ensembles – a knowledge reuse framework for combining multiple partitions. Journal of Machine Learning Research 3 ,583-617, 2002.

[Theodoridis and Koutroumbas, 2006] Theodoridis, S. & Koutroumbas, J. Pattern Recognition 3rd ed., 635, 2006.

[Tian et al., 1999] Tian, B., Shaikh, A., Azimi-Sadjadi, M. R., & Vonder Haar, T. H. A study of cloud classification with neural network using spectral and textural features. IEEE Transactions on Neural Networks, 11, 138−151, 1999.

[Tian et al., 2000] Tian, B., Azimi-Sadjadi, M. R., Vonder Haar, T. H., & Reinke, D. Temporal updating scheme for probabilistic

neural network with application to satellite cloud classification. IEEE Transactions on Neural Networks, 11, 903−920, 2000.

[Topchy et al., 2005] Topchy, A., Jain, A. K., Punch, W. Clustering ensembles: models of consensus and weak partitions. Pattern Analysis and Machine Intelligence, IEEE Transactions on 27 (12), 1866-1881, October 2005.

[Tukey, 1977] Tukey, J.W. Exploratory data analysis. Addison-Wesley, 1977.

[Yang et al., 2007] Yang, Y., Lin, H., Guo, Z., & Jiang, J. A data mining approach for heavy rainfall forecasting based on satellite image sequence analysis. International Journal of Computers & Geosciences, 33, 20−30, 2007.

[Yang et al., 2006] Yang, Y., Lin, H., & Jiang, J. Cloud analysis by modeling the integration of heterogeneous satellite data and imaging. IEEE Transactions on Systems, Man and Cybernetics — Part A: Systems and Humans, 36, 162−172, 2006.

[Yu and Shi, 2003] Yu, Stella X., Shi, Jianbo. Multiclass spectral clustering. In: 374 Proc.Internat.Conf. on Computer Vision, 313-319, 2003.

[Wand and Jones, 1995] Wand, M.P, & Jones, M.C. Kernel smoothing. London: Chapman and Hall, 1995.

[Zhu and Ghodsi, 2006] Zhu, M., Ghodsi, A. Automatic dimensionality selection from the scree plot via the use of profile likelihodd.

Computational statistics and data analysis 51, 918-930, 2006.