Year : 2011

BIO-INSPIRED COMPUTATIONAL TECHNIQUES APPLIED TO THE CLUSTERING AND VISUALIZATION OF SPATIO-TEMPORAL GEOSPATIAL DATA

BARRETO SANZ Miguel

UNIL | Université de Lausanne

FACULTÉ DES HAUTES ÉTUDES COMMERCIALES

DÉPARTEMENT DES SYSTÈMES D'INFORMATION

BIO-INSPIRED COMPUTATIONAL TECHNIQUES APPLIED TO THE CLUSTERING AND VISUALIZATION OF SPATIO-TEMPORAL GEOSPATIAL DATA

THÈSE DE DOCTORAT

présentée à la

Faculté des Hautes Etudes Commerciales
de l'Université de Lausanne

pour l'obtention du grade de
Docteur en Systèmes d'Information

par

Miguel BARRETO SANZ

Codirecteurs de thèse
Prof. Marco Tomassini
Prof. Andres Perez-Uribe

Jury

Prof. Michael Rockinger, Président
Prof. Ann Van Ackere, experte interne
Prof. David Elizondo, expert externe
Prof. Carlos Peña-Reyes, expert externe

LAUSANNE
2011

UNIL | Université de Lausanne
HEC Lausanne
Le Doyen
Bâtiment Internef
CH-1015 Lausanne

IMPRIMATUR

Sans se prononcer sur les opinions de l'auteur, la Faculté des hautes études commerciales de l'Université de Lausanne autorise l'impression de la thèse de Monsieur Miguel Arturo BARRETO SANZ, titulaire d'un diplôme d'Ingénieur en Electronique de l'Université del Valle, Cali, Colombie, en vue de l'obtention du grade de docteur en Systèmes d'information.

La thèse est intitulée :

**BIO-INSPIRED COMPUTATIONAL TECHNIQUES APPLIED
TO THE CLUSTERING AND VISUALIZATION
OF SPATIO-TEMPORAL GEOSPATIAL DATA**

Lausanne, le 5 décembre 2011

Le doyen

Daniel Oyon

Members of the jury

Monsieur Marco TOMASSINI, Professeur, Faculté des Hautes Etudes Commerciales, Université de Lausanne. Codirecteur de thèse.

Monsieur Andres PEREZ-URIBE, Professeur, Institut des Technologies de l'Information et de la Communication, Haute Ecole d'Ingénierie et de Gestion du Canton de Vaud (HEIG-VD), Yverdon. Codirecteur de thèse.

Madame Ann VAN ACKERE, Professeure, Faculté des Hautes Etudes Commerciales, Université de Lausanne. Experte interne.

Monsieur David ELIZONDO, Professeur, Centre for Computational Intelligence, De Montfort University, Leicester, UK. Expert externe.

Monsieur Carlos PEÑA-REYES, Professeur, Institut des Technologies de l'Information et de la Communication, Haute Ecole d'Ingénierie et de Gestion du Canton de Vaud (HEIG-VD), Yverdon. Expert externe.

Monsieur Michael ROCKINGER, Professeur et Vice-doyen de la Faculté des Hautes Etudes Commerciales de l'Université de Lausanne. Président du jury.

Université de Lausanne
Faculté des Hautes Etudes Commerciales

Doctorat en Systèmes d'Information

Par la présente, je certifie avoir examiné la thèse de doctorat de

Miguel Arturo BARRETO SANZ

Sa thèse remplit les exigences liées à un travail de doctorat.
Toutes les révisions que les membres du jury et le-la soussigné-e ont demandées durant le colloque de thèse ont été prises en considération et reçoivent ici mon approbation.

Signature : MTomassini Date : 5/12/2011

Prof. Marco TOMASSINI
Codirecteur de thèse

Université de Lausanne
Faculté des Hautes Etudes Commerciales

Doctorat en Systèmes d'Information

Par la présente, je certifie avoir examiné la thèse de doctorat de

Miguel Arturo BARRETO SANZ

Sa thèse remplit les exigences liées à un travail de doctorat.
Toutes les révisions que les membres du jury et le-la soussigné-e ont demandées durant le colloque de thèse ont été prises en considération et reçoivent ici mon approbation.

Signature : Andres Perez U. Date : 2.12.2011

Prof. Andres PEREZ-URIBE
Codirecteur de thèse

Université de Lausanne
Faculté des Hautes Etudes Commerciales

Doctorat en Systèmes d'Information

Par la présente, je certifie avoir examiné la thèse de doctorat de

Miguel Arturo BARRETO SANZ

Sa thèse remplit les exigences liées à un travail de doctorat.
Toutes les révisions que les membres du jury et le-la soussigné-e ont demandées durant le colloque de thèse ont été prises en considération et reçoivent ici mon approbation

Signature : ______________________ Date : 1/12/2011

Prof. Ann VAN ACKERE
Membre interne du jury

Université de Lausanne
Faculté des Hautes Etudes Commerciales

Doctorat en Systèmes d'Information

Par la présente, je certifie avoir examiné la thèse de doctorat de

Miguel Arturo BARRETO SANZ

Sa thèse remplit les exigences liées à un travail de doctorat.
Toutes les révisions que les membres du jury et le-la soussigné-e ont demandées durant le colloque de thèse ont été prises en considération et reçoivent ici mon approbation

Signature : ______________________ Date : 1/12/12

Prof. David ELIZONDO
Membre externe du jury

Université de Lausanne
Faculté des Hautes Etudes Commerciales

Doctorat en Systèmes d'Information

Par la présente, je certifie avoir examiné la thèse de doctorat de

Miguel Arturo BARRETO SANZ

Sa thèse remplit les exigences liées à un travail de doctorat.
Toutes les révisions que les membres du jury et le-la soussigné-e ont demandées durant le colloque de thèse ont été prises en considération et reçoivent ici mon approbation

Signature : __________________ Date : 1 décembre 2011

Prof. Carlos PENA-REYES
Membre externe du jury

Abstract

The coverage and volume of geo-referenced datasets are extensive and incessantly growing. The systematic capture of geo-referenced information generates large volumes of spatio-temporal data to be analyzed. Clustering and visualization play a key role in the exploratory data analysis and the extraction of knowledge embedded in these data. However, new challenges in visualization and clustering are posed when dealing with the special characteristics of this data. For instance, its complex structures, large quantity of samples, variables involved in a temporal context, high dimensionality and large variability in cluster shapes.

The central aim of my thesis is to propose new algorithms and methodologies for clustering and visualization, in order to assist the knowledge extraction from spatio-temporal geo-referenced data, thus improving making decision processes.

I present two original algorithms, one for clustering: the *Fuzzy Growing Hierarchical Self-Organizing Networks* (FGHSON), and the second for exploratory visual data analysis: the *Tree-structured Self-organizing Maps Component Planes*. In addition, I present methodologies that combined with FGHSON and the Tree-structured SOM Component Planes allow the integration of space and time seamlessly and simultaneously in order to extract knowledge embedded in a temporal context.

The originality of the FGHSON lies in its capability to reflect the underlying structure of a dataset in a hierarchical fuzzy way. A hierarchical fuzzy representation of clusters is crucial when data include complex structures with large variability of cluster shapes, variances, densities and number of clusters. The most important characteristics of the FGHSON include: (1) It does not require an a-priori setup of the number of clusters. (2) The algorithm executes several self-organizing processes in parallel. Hence, when dealing with large datasets the processes can be distributed reducing the computational cost. (3) Only three parameters are necessary to set up the algorithm.

In the case of the Tree-structured SOM Component Planes, the novelty of this algorithm lies in its ability to create a structure that allows the visual exploratory data analysis of large high-dimensional datasets. This algorithm creates a hierarchical structure of Self-Organizing Map Component Planes, arranging similar variables' projections in the same branches of the tree. Hence, similarities on variables' behavior can be easily detected (e.g. local correlations, maximal and minimal values and outliers).

Both FGHSON and the Tree-structured SOM Component Planes were applied in several agroecological problems proving to be very efficient in the exploratory analysis and clustering of spatio-temporal datasets.

In this thesis I also tested three soft competitive learning algorithms. Two of them well-known non supervised soft competitive algorithms, namely the *Self-Organizing Maps* (SOMs) and the *Growing Hierarchical Self-Organizing Maps* (GHSOMs); and the third was our original contribution, the FGHSON. Although the algorithms presented here have been used in several areas, to my knowledge there is not any work applying and comparing the performance of those techniques when dealing with spatio-temporal geospatial data, as it is presented in this thesis.

I propose original methodologies to explore spatio-temporal geo-referenced datasets through time. Our approach uses time windows to capture temporal similarities and variations by using the FGHSON clustering algorithm. The developed methodologies are used in two case studies. In the first, the objective was to find similar agroecozones through time and in the second one it was to find similar environmental patterns shifted in time.

Several results presented in this thesis have led to new contributions to agroecological knowledge, for instance, in sugar cane, and blackberry production.

Finally, in the framework of this thesis we developed several software tools: (1) a Matlab toolbox that implements the FGHSON algorithm, and (2) a program called BIS (Bio-inspired Identification of Similar agroecozones) an interactive graphical user interface tool which integrates the FGHSON algorithm with Google Earth in order to show zones with similar agroecological characteristics.

Acknowledgments

> "Our lives are defined by opportunities, even the ones we miss."
>
> Benjamin Button

This thesis has been the pretext to meet many interesting people. Many of them have unknowingly contributed to the culmination of this thesis with his love, teaching or support. I think that that would be necessary to add a new chapter to mentioned all those teachers, friends and family that have shaped my life. Nevertheless, I would like to mentioned some of those people that have strongly marked my life after and before arriving to Switzerland.

Teachers. I thank Humberto Loaiza and Eduardo Caicedo at the "Universidad del Valle", who introduced me to the research world and who had the initiative of proposing me as candidate for the scholarship that led me to Switzerland. I thank Mvriam Sanchez and all the members of the Coporación BIOTEC whom I enjoyed to work during the project "Precision agriculture and the construcion of field-crop model for tropical fruit species". I thank James Cock, Andy Jarvis (both with British nationality but Colombian heart) and Daniel Jimenez from CIAT, people that I admire and from whom I learned to love the agriculture and appreciate the hard work of the farmers of my country. I am very grateful to Eduardo Sanchez who allowed me to make my research at the HEIG-VD. I admire his dedication to maintain the scientific collaboration between Colombia and Switzerland and his clever sense of humor. I thank Etienne Messerli and all the REDS who makes me feel all time welcome and part of their team. I thank to professor Marco Tomassini for receiving me in his group and accepting to supervise my thesis. I appreciate the interest he showed in the topics of my research, and his incoditional whenever I needed to resolve many situations during my thesis. My profound acknowledgments to the members of the jury, Prof. Ann Van Ackere, Prof. David Elizondo, for their interest in my research, careful reading and useful suggestions.

I am very gratefull to the two best teachers that I had have in my academic life Andres Perez-Uribe and Carlos Peña-Reyes. They taught me (and they still do) to think, to love the research, the science and the most important to love the work that I do. They are for me models of integrity in their work and personal life. To them my grateful for their patience, trust and continuous support.

Family. First of all, I am immensely grateful to my parents Maria Teresa and Bromberg Arturo, and my sister Maria Angelica. They supported me all time in my projects and they never put limits to my dreams. I am very grateful to all my family specially my grandmother Amanda and my aunt Carmen who have been my second mothers.

Friends. During life you have the brothers and sisters that the nature gave you and the brothers and sisters that you choose. My sincerely grateful to my brothers and sisters of life: at the "Salesiano San Juan Bosco" school Carlos Guzman, Jose Abdon Ordoñez, and Mauricio Agudelo which remain a source of inspiration for me, at the "Universidad del Valle" Yaned Castillo, Daniel Escobar, German Ospina, Monica Idarraga, Susana Castrillon, Paula Montenegro, Evlep Garcia, and Angelica Jaramillo whom I shared extra academic moments when life was simple and the only mathematical problem to be solved was to calculate the number of falling stars while we were lying on the grass of Guasanó. I thank to my friends at the Faculty of Electronics at the "Universidad del Valle" Laura Sofia, Diana Maria Cadavid and Julian Flórez, who mutual support and friendship makes possible that we conclude our bachelor.

Friends in Switzerland. To live in a foreign country implies to leave family and friends and to adapt to a new culture. I was so lucky to have three families in Switzerland that make my adaptation process easy and pleasant. They allow me to learn about Switzerland and other cultures while at the same time maintaining my heart in Colombia. I called them my Colombian, Swiss, and the rest of world family.

My thanks to the people that make me feel at home: Andres Upegui, Joao Palacios, Olga Lucia Correa, Jonathan Paz, Vladimir, Daniel Avella, Ricardo Avella, Juan David Villegas, Catalina Muñoz, Jorge Castro, Monica Rengifo, Yenni and Jazmin Rosas, Patricia Gomez, Fernando Villabona and Maria Alejandra Ramirez.

I thank the rest of the world friends, they let me enter to their lives since they shared the same situation to be far away of their families: Cyrille Thomas, Jose Luis Carrasco, Pedro Faustino, Pilar Gomis, Luis Panichelli, Manuela Fernandez, Alejandra Peyon, Carmen Cianfrani, Filipe Bandeirinha, Carlotta Catarusa , Felicien Mazille and Phung Anh Khoa.

At he same time that I was discovering the country I discovered the people. My new and old Swiss friends taught me not only French but also make me discover the Swiss culture and make part of their lives: Patricia Jaunin, Nicole Rothenbühler, Sophie Michel, Sandra Viatte, Karin Juillerat, Laurance Rodríguez, Gilles Curchod, Romain Bornet, Igor Rothenbühler, and Sébastien Gerber.

There are two special friends, that although they have a vision of life totally different, they make my days exceptionally happy and have given me their unconditional support in the most difficult moments and their smiles in the happy ones. The biggest of the thanks to Catalina Garcia and Héctor Sátizábal.

I am very grateful to Catalina Garcia who has been my support this last two years and whose friendship has nourished my spirit, my heart and my stomach. She also introduced me a fantastic group of friends with whom I spent the last year of this journey: Laura Parra, Zully Ruiz, Anastasia Boykova, Vanessa Cule, Teresa Perez, Vivian Kümmerlen, Virgie Alonso, Elena Komissarova, Arnulfo, Rodolfo Calderon, Rudi Jordaan and Salvatore Boccheti.

Héctor is the most transparent and sincere person that I had known. Being his friend is a great honor and the most valuable discovering in this six years in Switzerland, my sincere thanks to him.

Contents

Chapter 1

Introduction

> "It is by logic that we prove, but by intuition that we discover."
>
> Henri Poincaré

During the last decade, our ability to collect and store geospatial data has far outpaced our ability to process, analyze and exploit it. Scientists and engineers in many fields increasingly capture data associated to geographical areas, such as data daily received from space-borne instruments, remote sensing systems, and environmental monitoring devices. Therefore, the coverage and volume of geospatial datasets are extensive and incessantly growing. For instance, some researchers estimate that about 80% of the data stored in corporate databases integrate spatial information [23].

This systematic capture generates large volumes of spatio-temporal geospatial data to be analyzed. Clustering and visualization play a key role in the exploratory data analysis and the extraction of knowledge embedded in these datasets. However, new challenges are posed when dealing with its special characteristics. For instance, its complex structures, large quantity of observations, high dimensionality, large variability in cluster shapes, and unknown data distribution.

The central aim of my thesis is to propose new algorithms and methodologies for clustering and visualization which can assist the knowledge extraction from spatio-temporal geospatial datasets, thus improving making decision processes.

1.1 General context

1.1.1 Problem description

Satellites, automatic weather stations, and GPS devices allow to capture huge amounts of geospatial information. Thus, we had moved from a data poor and computation-poor to a data-rich and computation-rich environment. When the quantity and diversity of spatio-temporal geospatial data increases, several challenges appear extracting useful information from these, namely:

- In many cases when dealing with spatio-temporal geospatial datasets there is not a-priori information about the classes or groups to use as a reference to classify new observations. Hence, supervised methods for classification cannot be used.

- Classical techniques used for high-dimensional data visualization such as scatterplot matrices and parallel coordinates present several drawbacks when the number of variables and observations is large. For instance, in the case of the scatterplot matrices the number of displays grows quadratically as the number of variables increase [16, 61]. Therefore, this type of visualization is not practical in applications where it is necessary to explore for relationships between a large number of variables (e.g. to find partial correlations), since the large quantity of scaterplots to analyze make this a tedious process.

- Traditional statistical techniques such as frequentist inference and Bayesian inference are based on the statistical hypothesis testing method (confirmatory data analysis). This method require an a-priori hypothesis to be tested that is usually based on assumptions about the data distribution. However, when we deal with large geospatial data sets being updated constantly several drawbacks are presented: (1) the continuous updating of observations and variables renders difficult to make assumptions about the data distribution (2) statistical inferences techniques often come with a high computational cost, especially in models with a large number of parameters (3) statistical inference require skills to translate subjective prior beliefs into a mathematically formulated prior (4) the fact to assume priors render difficult to discover new and unexpected patterns, trends, and relationships hidden in data sets [87, 76].

- When working with spatio-temporal data we deal with the "spatiality" (the places where the variables are measured) but also the "temporality" (the behavior of the variables through time). An actual issue of research in spatio-temporal data mining concerns the development of efficient techniques for clustering and visualization of spatial-temporal geospatial datasets [12, 87, 101, 102, 100, 132]. These techniques have as objective to help discovering relationships and patterns embedded in a temporal context (e.g. similar patterns through time, relationships among variables and cluster dynamics).

Therefore, it is necessary to develop new algorithms and methodologies that can deal with the challenges posed by the spatio-temporal georeferenced data, in order to help explore, synthesize and extract knowledge from these datasets. The next Section presents our proposed solutions to the exploratory data analysis, clustering and data visualization of spatio-temporal geospatial data.

1.1.2 Proposed solution

I propose two algorithms in this thesis for clustering and visualization of spatio-temporal geospatial datasets, namely the *Fuzzy Growing Hierarchical Self-Organizing Networks* (FGHSON) and the *Tree-Structured SOM Component Planes*. The first focus on clustering, and the second on visualization. Both approaches are bio-inspired algorithms based on the *soft competitive learning* (SCL) [52] paradigm. This paradigm exhibits characteristics that render it adequate to tackle some of the problems addressed in this thesis, namely:

- **Discovery of natural clusters in unlabeled datasets**. The SCL algorithms group similar observations without having a-priori knowledge about groups or classes. They explore the underlying structure of the data allowing to classify patterns into categories taking into account this similarities;

- **The reduction of information redundancy contained in the data**. The SCL algorithms create prototypes of the observations. Hence, large datasets can be reduced with minimal loss of information;

- **Discovery of nonlinear, local or partial correlations between variables**. Usually, the high-dimensionality of the datasets makes difficult to visualize correlations among variables. SCL algorithms allow the projection of a high-dimensional space into a low dimensional space (usually two dimensions). Thus, helping the visual exploratory data analysis, facilitating the discovery of, for instance, non linear, local, or partial correlations between variables;

- **Exploration of data with unknown distribution**. The SCL algorithms do not use information about data distribution to group similar observations. Usually, a competitive learning process allows grouping the input data based on its similar patterns. Hence, clusters emerge based on data similarities.

Although the use of the SCL paradigm allows tackling many of these issues, when dealing with spatio-temporal datasets, several challenges still remain:

- **The representation of spatio-temporal data at various levels**;

  The spatial and temporal resolution used to present the information can have direct impact on the knowledge discovery process. Identical analysis at different resolution scales can lead to different results. Hence, visualization and clustering techniques used to explore spatio-temporal georeferenced data must allow hierarchical representations making possible to navigate clusters and structures at different levels.

- **To handle fuzzy boundaries in spatio-temporal clusters**;

  Clustering algorithms are used to group together elements with similar characteristics. Traditionally crisp clustering is used, which means that each element of the data set is assigned to only one cluster. Nevertheless, in many real word application elements can belong to several clusters with certain degree of membership. For instance, to assigning a geo-referenced place to a climatic group it is not a trivial task, since climatic groups are not only deserts or subarctic climates but all a palette of possibilities. Thus, a geo-referenced place can be part of several climatic groups with a certain degree of membership. In this context, an approach that allow to an element be part of several clusters is the fuzzy clustering. It creates intermediate classifications, rather than binary or "crisp" one, thus an object can belong to several clusters with a certain degree of membership. This is particularly useful when the boundaries between clusters are ambiguous and not well separated.

- **The temporal context in which some variables are involved**;

  Most research efforts extracting knowledge from spatio-temporal geospatial datasets only consider a global vision or average of the data. They are thus susceptible to ignore interesting patterns embedded in temporal context, for instance different phases of a phenomena, its transitions and dynamics.

In order to deal with the aforementioned issues, I develop the *Fuzzy Growing Hierarchical Self-Organizing Networks* (FGHSON) algorithm. FGHSON presents the advantages of the soft competitive learning paradigm and also allows creating fuzzy hierarchical classifications. This is particularly useful when the boundaries between the clusters are ambiguous. Most important characteristics of the FGHSON include: (1) It does not require a-priori setup of the number of clusters. (2) The algorithm executes several self-organizing processes in parallel. Hence, when dealing with large datasets the processes can be distributed reducing the computational cost. (3) Only tree parameters are necessary to set up the algorithm.

I also propose a new visualization technique applied to the exploratory analysis of spatio-temporal high-dimensional datasets called *Tree-structured SOM component planes*. This algorithm creates a tree structure that allows the visual exploratory data analysis of large high-dimensional datasets. The structure arranges similar variable projections in the same branches of the tree. Hence, similarities of variables' behavior can be easily detected (e.g. Local correlations, maximal and minimal values or outliers).

In addition, I present several methodologies to cluster spatio-temporal geospatial datasets through time. These approaches use time windows for capturing temporal similarities and variations by using the FGHSON clustering algorithm. Thus, allowing to integrate space and time seamlessly and simultaneously in order to extract knowledge embedded in a temporal context.

Both FGHSON and the Tree-Structured SOM component planes were applied to several agroecological problems proving to be very efficient in the exploratory data analysis and clustering of spatio-temporal geospatial datasets.

1.2 Outline of the thesis

This thesis is organized in 6 chapters, the chapter 1 being this introduction. Chapter 2.1 presents the theoretical rationales of the unsupervised competitive clustering that will serve as introduction to explain the FGHSON algorithm at the end of the chapter. The chapter starts by (Section 2.2) presenting the basic principle of competitive learning. After presenting the two main principal approaches of competitive learning (hard and soft competitive learning) in Section 2.3 the chapter continues with a discussion of the hierarchical self-organizing structures. Section 2.4 discuss on the fuzzy representation in Hierarchical Self-Organizing structures that serves as an introduction to Section 2.4 that presents the Fuzzy Growing Hierarchical Self-Organizing Networks (FGHSON) a novel approach for clustering of spatio-temporal data.

Chapter 3 presents several techniques for the visualization of spatio-temporal geospatial data. In Section 3.1 we explain the methodologies for interpreting clusters by using Self-Organizing Maps (SOMs). This serves as a theoretical introduction to Section 3.2 where we present the *tree-structured component planes representation*. Finally, in Section 3.3 we present several applications of these algorithms in agriculture.

In the introduction of chapter 4, I explain the challenges when clustering spatio-temporal geospatial data. Later, in Section 4.1 several clustering approaches based on unsupervised soft competitive learning are used in real word problems. In addition, we discuss the advantages and disadvantages of these algorithms when dealing with geospatial datasets. Finally, in Section 4.2 the FGHSON method is applied to the problem of clustering spatio-temporal geospatial data.

In chapter 5, we discuss how bio-inspired clustering approaches can be used for extract information embedded in a temporal context. Section 5.1 presents the problem of finding similar agroecozones through time, the several methodological approaches to tackle this and the related work. In Section 5.2 we illustrate the use of self-organizing maps to delineate sugar cane agroecozones. Finally, Section 5.4 presents a more developed methodology using the FGHSON method for clustering and finding analogous ecoregions through time.

Finally in chapter 6, we present the conclusions, summarizing the goals achieved in the present work, and the original contributions made. The chapter ends with several considerations on possible future work derived from this thesis.

In addition, annex chapters (corresponding to the published papers) are included containing a more detailed discussion of some aspects. They are mentioned in the relevant sections in order to give a deeper description of the methodologies developed applications.

Chapter 2

Unsupervised competitive ANN-Based clustering

THere are two main learning paradigms in Artificial Neural Networks (ANN): supervised and unsupervised [9]. In *supervised learning* a correct answer (output) for every input pattern is provided to the network. Then, weights are calculated to allow the network to produce answers as close as possible to the known correct answers. *Reinforcement learning* is a variant of supervising learning in which the network is provided with only a critique on the correctness of network outputs, not the correct answers themselves. In contrast, *unsupervised learning*, does not require a correct answer associated with each input pattern in the training dataset. It explores the underlying structure of the data, and organizes patterns into categories from their similarities.

Unsupervised learning algorithms are particularly good in achieving the following goals [14]:

1. To help discover natural clusters in unlabeled datasets;
2. The reduction of information redundancy contained in the data;
3. To deal with data with unknown distribution;

One of the most popular learning rules that implement the unsupervised learning paradigm is the *competitive learning*. Several algorithms based on competitive learning have been developed to resolve problems in diverse fields. Usually, new features are added to previous algorithms making the new ones more robust and/or more suitable to tackle specific problems. An interesting family of algorithms is the so-called *hierarchical self-organized structures* [80, 96, 97]. They have the ability of representing hierarchical clusters from data using soft competitive learning methods. The networks created by these algorithms allow capturing dataset topologies in terms of hierarchical relationships and clustering structures. Although these hierarchical structures provide satisfactory results, they generate crisp partitions of the datasets, thus assigning each element of the datasets to only one cluster. This property of creating crisp hierarchical clustering makes these algorithms less suitable for many real applications where data include complex structures with large variability of cluster shapes, densities, number of clusters and overlapping clusters [53].

In order to face the challenges presented when exploring real world data, we assemble the features of the hierarchical self-organized structures with the characteristics of the fuzzy clustering on a new algorithm: the *Fuzzy Growing Self-Organizing Networks* algorithm (FGHSON). The originality of the FGHSON algorithm lies in its capability to reflect the similarities of the elements of a dataset in a hierarchical fuzzy way. This algorithm creates a fuzzy hierarchical cluster structure, thus each element of the dataset is grouped in several clusters with a certain degree of membership, and in addition, in several levels of a hierarchy. It is an effective method for exploring the structure of complex real data with unknown relationships. Its hierarchical structure allow to navigate the clusters at different resolutions, thus allowing to explore the similarities of the elements at several granuralities.

In this chapter in section 2.2 I present the theoretical rationales of the unsupervised competitive learning. After presenting the soft competitive learning approach in section 2.2 the chapter continues with a discussion of the hierarchical self-organizing structures in section 2.3. Section 2.4 discusses the fuzzy representation in hierarchical self-organizing structures that serves as an introduction to the Section 2.4 that presents my original contribution i.e., the Fuzzy Growing Hierarchical Self-Organizing Networks (FGHSON) algorithm.

2.1 Competitive learning

Competitive learning allows to categorize an input dataset grouping similar patterns into the same output units of a network, as shown in Figure 2.1. The basic principle of competitive learning consists in gather together similar input vectors by means of a competitive process where the output units of the network compete for activation based on their similarities with the input data.

Specifically, a two-layer feedforward neural network that implements the idea of competitive learning is depicted in Figure 2.1. The nodes in the input layer admit input patterns and are fully connected to the output nodes in the competitive layer. Each output node corresponds to a cluster and is associated with a prototype or weight vector $\mathbf{w}_j$, j=1,..., K, where K is the number of clusters, stored in terms of synaptic weights w_{ij}, j = 1,...,d, representing the connection between input node i and output node j. Given an input pattern $\mathbf{x} = \{x_1, \ldots, x_d\}$ presented at iteration t, the similarity between the weight vector $\mathbf{w}_j$ of the randomly initialized cluster j , and $\mathbf{x}$ can be obtained by

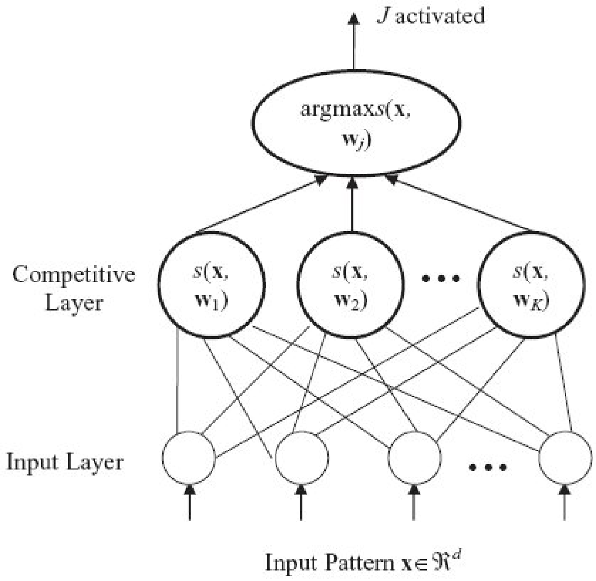


Figure 2.1: *A competitive learning network with excitatory connections from the input nodes to the output neurons. Each node in the competitive layer is associated with a weight vector* $\mathbf{w}_j$*. The neuron that is nearest to the input pattern, based on the pre-specified similarity or distance function, is fired, and its prototype is adapted to the input pattern thereafter. However, updating will not occur for other losing neurons.*

computing the net activation v_i,

$$s\left(\mathbf{x}, \mathbf{w}_j\right) = v_j = \mathbf{w}_j^T \mathbf{x} = \sum_{i=1}^{d} w_{ji} x_i, \tag{2.1}$$

The neurons in the competitive layer then compete with each other, and only the one with the largest net activation value becomes activated or fired, written as,

$$J = \arg\max_j s\left(\mathbf{x}, \mathbf{w}_j\right) \tag{2.2}$$

The weight vector of this winning neuron J is further moved towards the input pattern following the updating equation,

$$\mathbf{w}_J\left(t+1\right) = \mathbf{w}_J(t) + \eta\left(\mathbf{x}\left(t\right) - \mathbf{w}_J\left(t\right)\right), \tag{2.3}$$

where η is the learning rate.

The competitive learning paradigm described above only allows learning for the particular winning neuron that best matches the given input pattern. Thus, it is also known as *winner-take-all* (WTA) or *hard competitive learning*. On the other hand, learning can also occur in a cooperative way, which means that not just the winning neuron adjusts its prototype, but all other cluster prototypes have the opportunity to be adapted based on how similar they are to the input pattern. This learning method, that will be presented in next Section, is called *soft competitive learning* or *winner-take-most* (WTM).

2.2 Soft competitive learning

Two mayor major problems exist in hard competitive learning. First: the dead neuron problem. It is presented when neurons that are positioned far away from the input patterns so they have no opportunity to ever win the competition, and therefore, no opportunity to be trained. Second: the random initialization of the neurons. In hard competitive learning different random initializations may lead to very different results. The purely local adaptations may not be able to get the system out of the poor local minimum where it was started.

One solution to address both problems is to change the *winner-take-all* approach to the *winner-take-most* also called *soft competitive learning*. In this case, both winning and losing neurons are allowed to move towards the presented input pattern, but with different learning rates. The winning neuron learns much faster than the other neurons. A simple soft competitive learning rule can be stated as:

$$\mathbf{w}_j(t) = \begin{cases} \mathbf{w}_j(t-1) + \eta_w \left(\mathbf{x}_j\left(t\right) - \mathbf{w}_j\left(t-1\right)\right), & \text{if } \mathbf{w}_j \text{ wins,} \\ \mathbf{w}_j(t-1) + \eta_l \left(\mathbf{x}_j\left(t\right) - \mathbf{w}_j\left(t-1\right)\right), & \text{if } \mathbf{w}_j \text{ loses} \end{cases} \tag{2.4}$$

where $\mathbf{w}_j$ represents a weight vector, $\mathbf{x}_j$ an input given pattern, η_w is the learning rate for the winner neuron, and η_l is the learning rate for the loser neurons.

According to Fritzke [52] there are two main classes of soft competitive learning: the soft competitive learning *without fixed network dimensionality* and *with fixed network dimensionality*.

On the one hand, in the *soft competitive learning without fixed network dimensionality* no topology of a fixed dimensionality is imposed on the network. The dimensionality of the network depends on the local dimensionality of the data and may vary within the input space.

On the other hand, the algorithms based on fixed network dimensionality allow to map a high-dimensional input space into a low-dimensional network structure (usually in two or three dimensions). This makes it possible to get a low dimensional representation of the input patterns. This representation in a low-dimensional space is frequently used for data visualization purposes. This kind of algorithms includes the Fritzke's Growing Cell Structures [51] and the Self-Organizing Maps (SOMs) [75]. In order to further explain the soft competitive learning with fixed network dimensionality I present the SOM algorithm in the next section.

2.2.1 Self-Organizing Maps (SOMs)

A SOM is made of artificial neurons situated on a regular low-dimensional grid in an ordered fashion (see figure 2.2) that can be effectively utilized to visualize and explore properties of the data. This grid can be one, two or three dimensional. Generally two dimensions are used. The neurons in the grid have a rectangular or a hexagonal form. Each neuron i represents an n-dimensional prototype vector $\mathbf{m}_i = [\mathbf{m}_{i_1}, \ldots, \mathbf{m}_{i_n}]$, where n is equal to the dimension of the input space.

The training process starts with the initialization of the prototype vectors with random values. On each step of the training, a data vector (observation) $\mathbf{x}$ from the input data is randomly selected and presented to the SOM. The unit $\mathbf{m}_c$ closest to $\mathbf{x}$ is located from the map, this winner unit is called: *the best-matching unit* or *BMU*. The *BMU* and its neighboring prototype vectors on the grid are moved in the direction of the observation vector, as follows:

$$\mathbf{m}_i = \mathbf{m}_i + \alpha(t) h_{ci}(t)(\mathbf{x} - \mathbf{m}_i) \tag{2.5}$$

where $\alpha(t)$ is learning rate and $h_{ci}(t)$ is a neighborhood kernel centered on the winner unit c, as shown in Figure 2.2. The learning rate and neighborhood kernel radius decrease monotonically with time.

Through the iterative training, the SOM auto-organizes the neurons. So those neurons that represent similar observations in the input space are located on the map in contiguous zones, trying to preserve the relations of the input space.

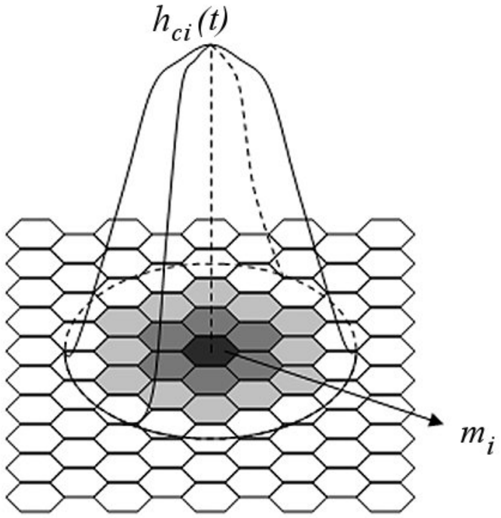


Figure 2.2: *The Self-Organizing Map with a hexagonal neuron lattice. The neighborhood function $h_{ci}(t)$ is centered over the best matched neuron m_i, which is shown as a black cell. The neighboring neurons that have their weights recalculated by this best match are shown in gray. Other neurons are not affected.*

2.3 Hierarchical Self-Organizing Structures

Often, real-world data present some degree of hierarchical structure. Hierarchical representations can help organize and explore the data patterns at different levels [127]. A hierarchical representation allow us to drill down the data in order to discover patterns that might be hidden to other simpler models (See example in Figure 2.3).

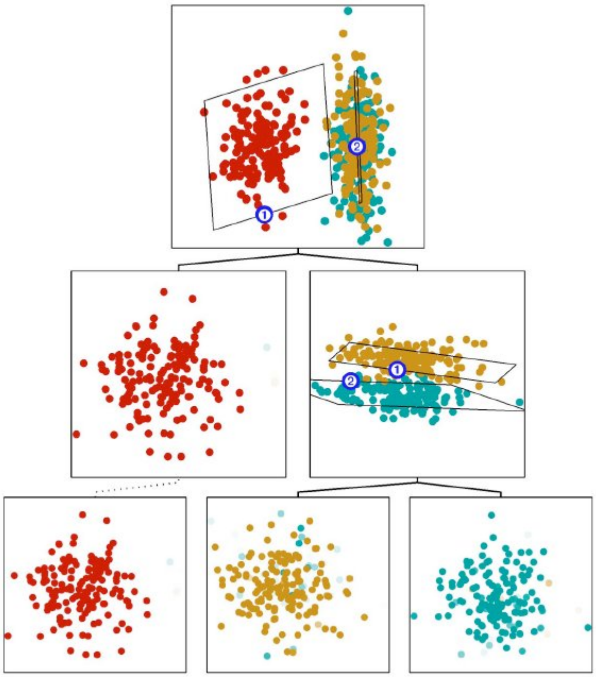


Figure 2.3: *An example of a hierarchical model, where more details on the structure of the data are revealed in each level, from Bishop et al [25]*

Several approaches have been introduced combining the advantages of a hierarchical representation and the capabilities of the soft competitive learning, for instance when dealing with unlabeled datasets, reduction of information redundancy, projection of high-dimensional data and data unknown distribution [98, 27, 84, 43, 114, 63, 86, 80]. These algorithms have been successfully applied in many areas, for instance in document retrieval [33], image retrieval [130], computer vision [77], and environmental sciences [79].

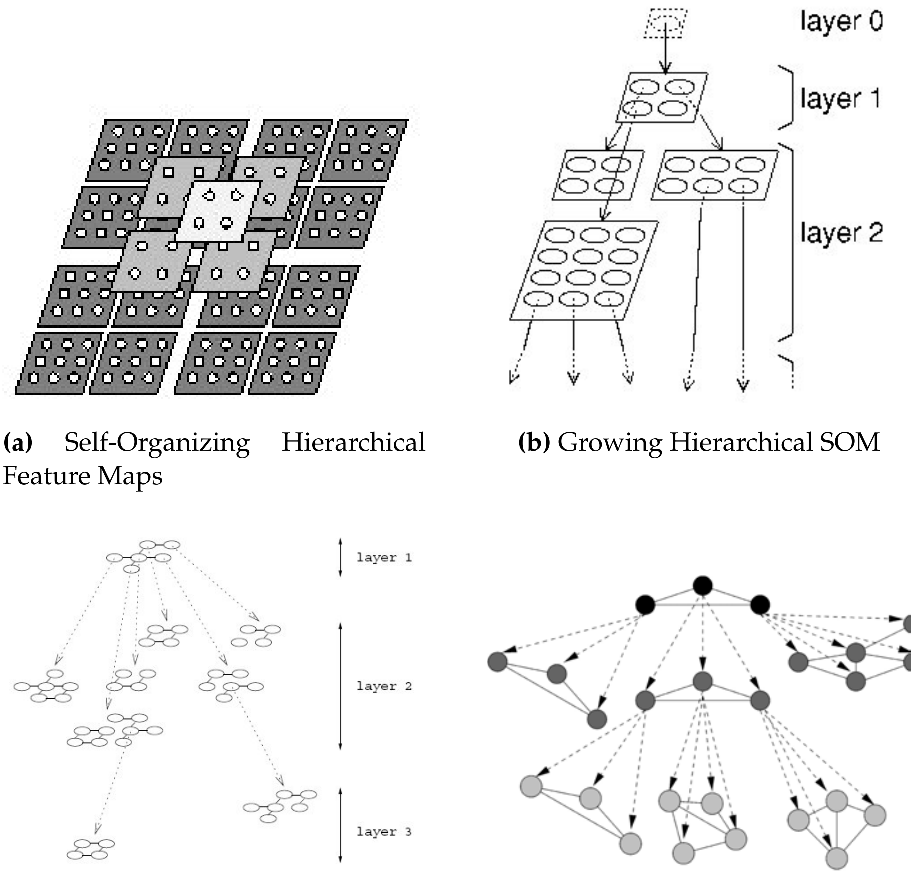
It is possible to classify the hierarchical self-organizing algorithms in two classes taking into account the algorithm used as inspiration. Usually they are based on the SOMs or on Growing Cell Structures (GCS) [19, 127].

The algorithms based on SOMs include the *Growing Hierarchical Self-Organizing Maps* (GHSOM) [98], the *Hierarchical Feature Maps* (HFM) [86], the *Adaptive Hierarchical Incremental Grid Growing* (AHIGG) [84], and the *Self-Organizing Hierarchical Feature Maps* [77].

The algorithms inspired on GCS are the *TreeGCS* [63], the *Hierarchical Growing Cell Structures* (HiGCS) [27] and the *Hierarchical Growing Neural Gas* (TreeGNG) [44].

In Figure 2.4 are shown some of the structures created for the algorithms aforementioned.

Although these hierarchical structures provide satisfactory results, they generate crisp partitions of the datasets where an observation can only be a member of a single



(a) Self-Organizing Hierarchical Feature Maps

(b) Growing Hierarchical SOM

(c) Adaptive Hierarchical Incremental Grid Growing

(d) Hierarchical Growing Cell Structures

Figure 2.4: *Structures created for Hierarchical Self-Organizing algorithms*

cluster. Nevertheless, in many applications crisp partitions are not the optimal representation since some observations could belong to different clusters with a certain degree of membership. An approach to tackle this problem is to group similar elements of a dataset in fuzzy hierarchical structures [18]. Hence, by using a fuzzy representation of memberships, it is possible to produce intermediate classifications, rather than binary or "crisp" ones. The improvements in the hierarchical self-organizing structures in order to add this feature is the topic of the next section.

2.4 Fuzzy representation in Hierarchical Self-Organizing Structures

In many research fields knowledge discovery is performed by using large datasets. Usually, knowledge extraction is conducted by optimally grouping analogous patterns in order to find similar behaviors for further analysis. The clustering algorithms used for this aim may fail when data include complex structures with large variability of cluster shapes, variances, densities and number of observations in each cluster. Examples can be found in complex biomedical signals [53], agro-ecology [17] and text mining [83].

One of the main problems, in these cases, is the estimation of the number of clusters, the so-called *cluster validity problem* [94]. Cluster validity is a difficult problem that is crucial to the practical application of clustering . Most of the common criteria for cluster validity have failed to estimate the correct number of clusters in complex data with a large number of clusters and a large variety of distributions within and between clusters. Hierarchical clustering seems to be a natural approach to solve this problem. In this context, *hard hierarchical clustering* methods are very well-known for recursively partitioning clusters into sub-clusters [131]. The partition can be bottom-up or top-down but, in both cases, a data pattern that is classified in one of the clusters cannot be reclassified to other clusters. This property of the classical hard hierarchical clustering methods make them unpractical for many real applications where data include complex structures with large variability of cluster shapes, number of clusters and overlapping clusters. In order to tackle this problem, hierarchical clustering algorithms should be developed to represent fuzzy dataset structures.

An approach to tackle this problem consists in including fuzzy representations in the hierarchical structures [18] by using fuzzy clustering. Hence, using a fuzzy representation it is possible to create intermediate classifications, rather than "crisp" ones. In fuzzy clustering, an object can belong to several clusters with a certain degree of membership. This is particularly useful when the boundaries between clusters are ambiguous and not well separated. Moreover, the memberships may help the users discover more sophisticated relationships between a given object and the clusters.

In order to amalgamate the advantages of a fuzzy hierarchical representation and the capabilities of soft competitive learning, I develop the *Fuzzy Growing Hierarchical Self Organizing Networks* (FGHSON) algorithm. This original algorithm creates a top-down hierarchical partition of a dataset by means of an unsupervised fuzzy clustering inspired by the *Fuzzy Kohonen Clustering Networks* (FKCN) [117]. In the next section we present *Fuzzy Kohonen Clustering Networks* (FKCN)as a theoretical base to subsequently introduce, in Section 2.5, the FGHSON.

2.4.1 Fuzzy Kohonen Clustering Networks (FKCN)

The FKCN [117], also known as *Fuzzy Learning Vector Quantization* (FLVQ), integrates the idea of fuzzy membership from *Fuzzy C-Means* (FCM) [21] with the updating rules from *Self-Organizing Maps*. This, creates a self-organizing algorithm that automatically adjusts the size of the updated neighborhood during a learning process. The learning process terminates when the FCM objective function is minimized. The updating rule for the FKCN algorithm can be given as:

$$\mathbf{W}_{i,t} = \mathbf{W}_{i,t-1} + \alpha_{ik,t}(\mathbf{Z}_k - \mathbf{W}_{i,t-1});\ for\ k = 1, 2, ..., n;\ for\ i = 1, 2, ..., c \qquad (2.6)$$

where $\mathbf{W}_{i,t}$ represents the centroid of the i^{th} cluster at iteration t , $\mathbf{Z}_k$ is the k_{th} vector example from the dataset and α_{ik} is the only parameter of the algorithm. The key idea of the FKCN algorithm is to obtain the value of the learning rate α_{ik} at iteration t via U_{ik}. According to [65]:

$$\alpha_{ik,t} = (U_{ik,t})^{m(t)} \qquad (2.7)$$

where $m(t)$ is an exponent like the fuzzification index in FCM and $U_{ik,t}$ is the membership value of the compound $\mathbf{Z}_k$ to be part of cluster i. Both of these constants vary at each iteration t according to:

$$U_{ik,t} = \left(\sum_{j=1}^{c} \left(\frac{\|\mathbf{Z}_k - \mathbf{W}_i\|}{\|\mathbf{Z}_k - \mathbf{W}_j\|} \right)^{2/(m-1)} \right)^{-1} \quad ;\ 1 \leq k \leq n\ ;\ 1 \leq i \leq c \qquad (2.8)$$

$$m(t) = m0 - m\Delta \cdot t \quad ; \quad m\Delta = (m0 - m_f)/iterate\ limit \qquad (2.9)$$

Where $m0$ is a constant value greater than the final value (m_f) of the fuzzification parameter m. The final value m_f should not be less than 1.1, in order to avoid the divide by zero error in equation (2.8).

The iterative process will stop if $\left\|\mathbf{W}_{i,(t)} - \mathbf{W}_{(i,t-1)}\right\|^2 < \epsilon$, where ϵ is a termination criterion or after a given number of iterations. At the end of the process, a matrix U is obtained, where U_{ik} is the degree of membership of the $\mathbf{Z}_k$ element of the dataset to the cluster i. In addition, the centroid of each cluster will form the matrix W where $\mathbf{W}_i$ is the centroid of the i^{th} cluster. The FKCN algorithm is given below:

```
1. Fix $c$, and $\epsilon > 0$ to some small positive constant.

2. Initialize $\mathbf{W}_0 = (\mathbf{W}_{1,0}, \mathbf{W}_{2,0}, \cdots, \mathbf{W}_{c,0}) \in \Re^c$.
   Choose $m_0 > 1$ and $t_{max} = max.\ number\ of\ iterations$.

3. For $t = 1, 2, \cdots, t_{max}$
   a. Compute all $cn$ learning rates $\alpha_{ik,t}$ with equations (2.7) and (2.8).
   b. Update all $c$ weight vectors $\mathbf{W}_{i,t}$ with
      $\mathbf{W}_{i,t} = \mathbf{W}_{i,t-1} + [\sum_{k=1}^{n} \alpha_{ik,t}(\mathbf{Z}_k - \mathbf{W}_{i,t-1})] \,/\, \sum_{j=1}^{n} \alpha_{ij,t}$
   c. Compute $E_t = \left\|\mathbf{W}_{(t)} - \mathbf{W}_{(t-1)}\right\|^2 = \sum_{i=1}^{c} \left\|\mathbf{W}_{i,(t)} - \mathbf{W}_{(i,t-1)}\right\|^2$
   d. If $E_t < \epsilon$ stop.
```

2.5 Fuzzy Growing Hierarchical Self-Organizing Networks (FGHSON)

Hierarchical Self-Organizing Networks are used to visualize similarities in the elements of a dataset, since they arrange similar observations in branches (clusters) of a hierarchical structure. The hierarchical structures are useful in exploratory data analysis, because they permit to explore the clusters at various levels, allowing to find groups or relationships at several granularities. One disadvantage of the hierarchical self-organizing networks is that they classify one observation in only one cluster (also called crisp partitions). However, in a variety of important applications, overlapping clustering, wherein some items are allowed to be members of two or more discovered clusters, is more appropriate. To deal with this challenge we propose the Fuzzy Growing Hierarchical Self-Organizing Networks algorithm (FGHSON). The FGHSON [18] is an unsupervised clustering algorithm able to reflect the underlying structure of a dataset in a hierarchical fuzzy way. FGHSON arranges the observations of a dataset in a hierarchical structure, where each level of the hierarchy consists of one or several *Fuzzy Kohonen Clustering Networks* (FKCN) [117]. Hence, observations can be members of several clusters with a certain membership degree. A schema of this structure is shown in Figure 2.8. Based on an unsupervised learning process, the FGHSON creates a structure capable to grow and adapt to the manifold of the data fulfilling the following objectives:

1. To find the most suitable number of prototypes for each FKCN with the aim of representing in the most accurate way the input dataset.

2. If it is necessary, create new FKCNs (in a hierarchical way) in order to better represent particularities of the data.

The growing processes mentioned above are modulated by three parameters τ_1, τ_2, and φ that respectively regulate: 1) the so-called breadth growth (in this process new

prototypes are added to the FKCNs), 2) depth growth process (process in which new FKCNs are created forming a hierarchical structure), and 3) the minimal membership degree of observations to the prototypes.
The algorithm is presented below in more detail.

2.5.1 The algorithm

The FGHSON algorithm is performed in three stages, namely:

- **The initial setup and global network control.** Here the setup of the algorithm is conducted by using the information about the input dataset (e.g. number of observations and variables). Based on this information we obtain a global measure of the data distribution that will serve as a reference to find a better hierarchical representation of the data in the next steps.
- **The breadth growth process.** This process is responsible for calculating if new prototypes must be added to the FKCNs, in order to obtain a better representation of the data similarities. It is called *breadth growth process* because each time a prototype is added the FKCN is expanded.
- **The depth growth process.** In this process the algorithm decides if a new level must be created. It is called *depth growth process* because it controls the vertical growth in depth of the hierarchical structure by adding new levels.

A more detailed description of these processes is presented below.

2.5.1.1 Initial setup and global network control

Let us consider a dataset Z of size $m\ X\ n$, where m is the number of observations and n the number of variables.

The first step of the algorithm is focused on obtaining a global measure of the data distribution. For this aim we create a first prototype $\mathbf{W}_0$.

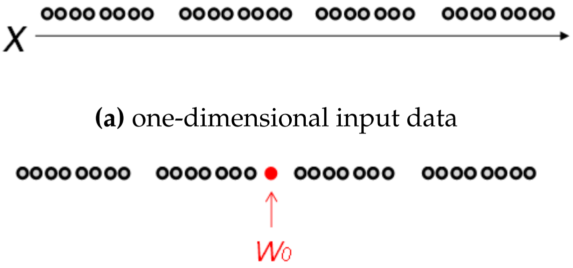
This prototype is defined as: $\mathbf{W}_0 = [x_1, x_2, \ldots, x_n]$, where x_i for $i = 1, 2, \ldots, n$; is computed as an average of the variable i for the whole input dataset. The $\mathbf{W}_0$ vector corresponds to the mean of each one of the input variables. As an example in Figure 2.5 is presented the W_0 of a one-dimensional dataset.

Thus, to obtain a quality measure of the $\mathbf{W}_0$ representation quantization error it is calculated. It will be called the global quantization error or qe_0, and it is calculated as shown in (2.10).

$$qe_0 = \sum_{\mathbf{Z}_m \in Z} \|\mathbf{W}_0 - \mathbf{Z}_m\| \tag{2.10}$$

where, Z_m represents the input vectors from the whole dataset Z.

The qe_0 represent the lowest quality of data representation, since in most cases, only one prototype it is no enough to represent in an accurate manner the several clusters that usually can be find in a dataset. It means that prototypes in the subsequent layers must be able to reduce the global representation error qe_0.



(a) one-dimensional input data

(b) The prototype W_0 created for this data

Figure 2.5: *A one-dimensional dataset illustrating the first prototype created by the FGHSON*

2.5.1.2 Breadth growth process

After computing qe_0, the creation of the first layer starts. This layer is made of a unique $FKCN$ as shown in Figure 2.6 (it will be denoted $FKCN_1$). To begin with, the $FKCN_1$ is composed of two prototypes, which will be called $\mathbf{W}_1$ and $\mathbf{W}_2$. Both prototypes are vectors with the same dimensionality as the input patterns, and they are initialized with random values.

The $FKCN_1$ is trained as shown in Section 2.4.1, taking as input, in the exceptional case of the first layer, the whole dataset. After training, the quantization error for both prototypes is calculated. Thus, qe_i represents the quantization error of the prototype $\mathbf{W}_i$.

In order to measure if the number of prototypes is enough for an acceptable representation of the data, or if it is necessary to add new ones, the Mean Quantization Error or (MQE) is calculated. It is computed according to expression (2.11).

$$MQE = \frac{1}{d} \cdot \sum_i qe_i \tag{2.11}$$

where d refers to the number of prototypes in the FKCN, and qe_i represents the quantization error of the prototype W_i.

If MQE is higher than:

$$\tau_1 \cdot qe0, \tag{2.12}$$

a new prototype is added. Where τ_1 represents a fixed percentage which control the growing process of the network of each layer. For this aim, the prototype with the highest qe is selected, and a new prototype representing the same data is created. After the insertion, all the FKCN parameters are reset to the initial values (except for the values of the prototypes) and the training begins according to the standard training process of the FKCN method. This process is repeated until the aforementioned condition is fulfilled. For instance, in Figure 2.6 it is shown that four prototypes in the $FKCN_1$ were necessary to represent the dataset in level one.

Note that :

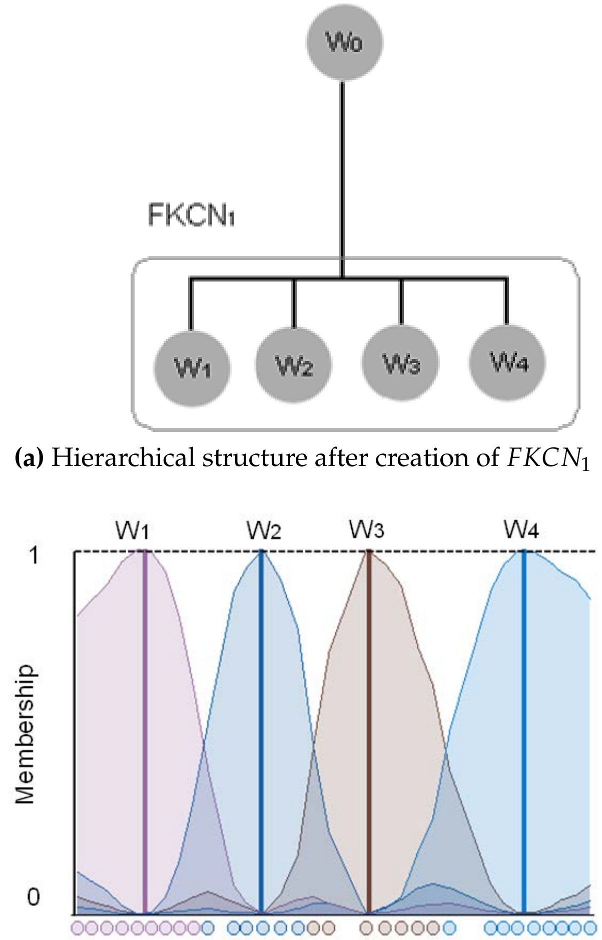
1. The same value of the parameter τ_1 is used in all layers of the FGHSON;

2. A membership matrix U containing the membership degree of the dataset elements to the prototypes is created, as explained in Section 2.4.1.

3. In a general form, equation (2.12) can be rewritten as :

   $$\tau_1 \cdot qe_u \tag{2.13}$$

   where qe_u represents the qe of the corresponding prototype u in the previous layer. In the specific case of the first-layer, qe_u is $qe0$.



(a) Hierarchical structure after creation of $FKCN_1$

(b) One-dimensional dataset, prototypes and memberships of the observations

Figure 2.6: *Structure created after creation of $FKCN_1$, and its prototypes in feature space. In this example, a one dimensional dataset is used. Observations are represented by dots. The degree of membership is shown for each prototype. After training, four prototypes in the $FKCN_1$ were necessary to represent the dataset in level one.*

2.5.1.3 Depth growth process

When the breadth process is finished, the qe's of the prototypes that composing the $FKCNs$ are evaluated. In the previous step, the Mean Quantization Error was used as an error measure; in this case the prototypes will be evaluated individually. In particular those prototypes which do not fulfill equation (2.14), will be selected for a further data representation.

$$qe_i < \tau_2 \cdot qe_0 \tag{2.14}$$

Where qe_0 is the global error calculated at the previous step, and the parameter τ_2 is a percentage that regulates the granularity of the data representation.

The method used for the FGHSON algorithm to improve the representation of the data of those prototypes consists of using the members of each prototype as inputs to a new FKCN, and subsequently, applying the breath growing process described in the previous section. The new FKCNs will be arranged in a deeper level, hence, creating a hierarchical structure. For instance, Figure 2.7 (a) shown that $\mathbf{W}_1$ and $\mathbf{W}_3$ are prototypes that do not fulfill equation (2.14), are expanded to form $FKCN_2$ and $FKCN_3$ respectively.

As it was mentioned previously, the methodology for adding new prototypes to the FKCNs, as well as the termination criterion of the breadth process is essentially the same as the one used in the first layer. However, the data used for the training processes of the FKCNs in the second and also in the subsequent layers, in comparison with the first, is only a fraction of the whole input data. This portion of data will be selected according to a minimal membership degree (φ), as shown in Figure 2.7 (b).

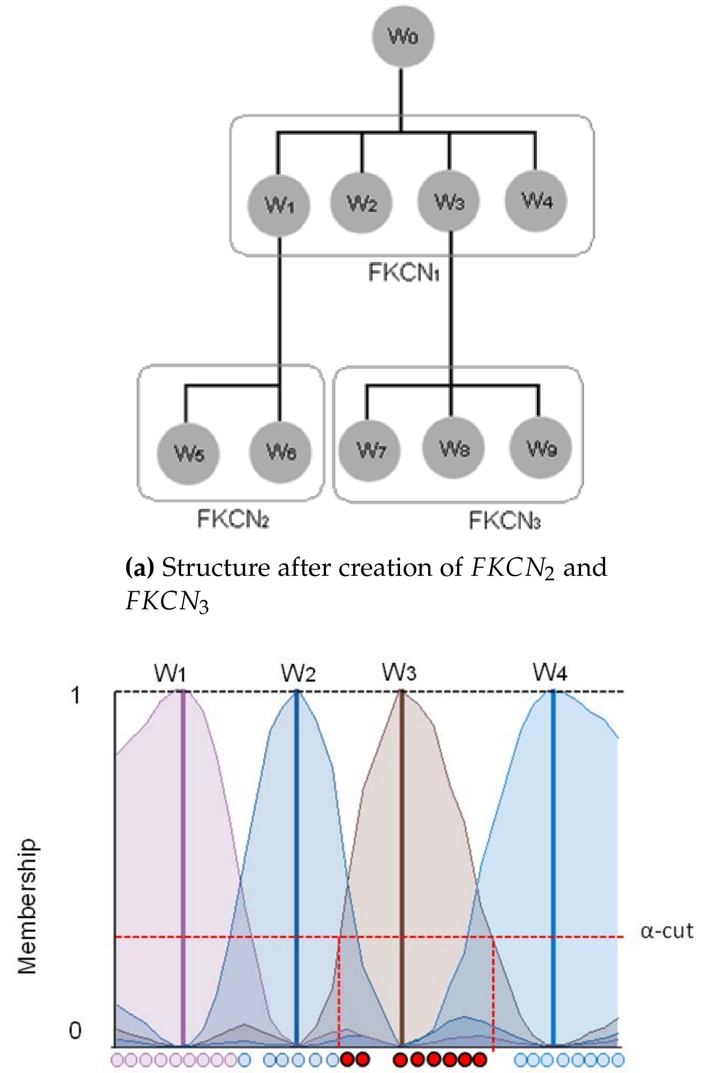
The parameter φ (the well-known $\alpha - cut$) represents the minimal degree membership of an observation to be part of the dataset represented by a prototype vector. Hence, $\alpha - cut$ is used as a selection parameter. Thus, all the observations represented by $\mathbf{W}_i$ have to fulfill expression (2.15)

$$\varphi < U_{ik} \tag{2.15}$$

where U_{ik} is the degree of membership of the Z_m element of the dataset to the cluster i.

At the end of the creation of layer two, the same procedure described in step 2 is applied in order to build the layer 3 and so forth.

The training process of the FGHSON is terminated when no more prototypes require further expansion.



(a) Structure after creation of $FKCN_2$ and $FKCN_3$

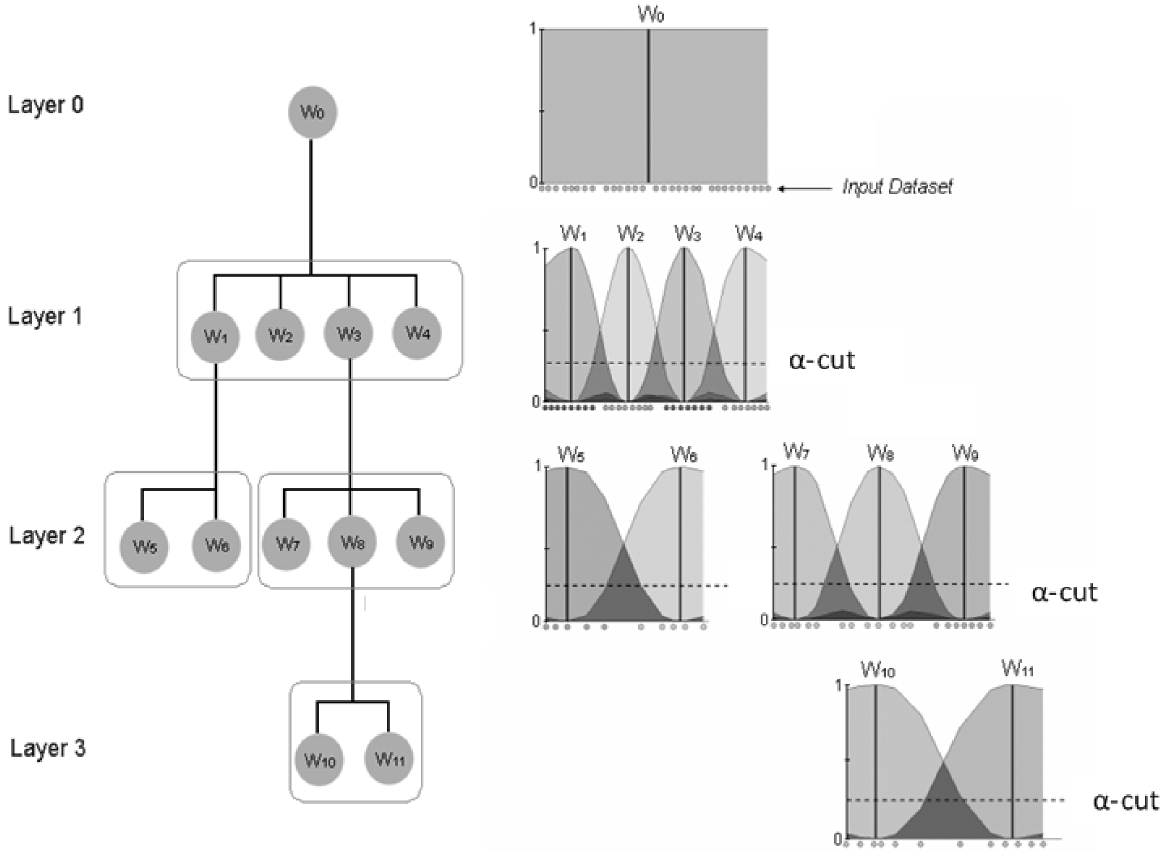
(b) Membership of the observations to the prototypes in $FKCN_1$

Figure 2.7: *Final structure after the creation of $FKCN_2$ and $FKCN_3$, and its prototypes in feature space. Note that $\mathbf{W}_1$ and $\mathbf{W}_3$ are expanded forming $FKCN_2$ and $FKCN_3$ respectively.*

2.5.2 Some remarks

Note that:

1. The training process does not necessarily lead to a balanced hierarchy, i.e. a hierarchy with equal depth in each branch. Rather, the specific distribution of the input data is modeled by a hierarchical structure, where some clusters require deeper branching than others.



(a) Structure created for the FGHSON

Figure 2.8: *Final FGHSON structure for the data example. Here, a one-dimensional dataset is used. For each level the prototypes created and the membership of the observation to that prototypes are shown. At level one, four prototypes are used to represent the dataset. At the level two, prototype W1 is split in the prototypes W5 and the W6. In a same way W3 is split in the prototypes W7, W8 and W9. Finally, prototype W7 is split in prototype W10 and W11.*

2. It does not require a priori a setup of the number of clusters.

3. The algorithm executes a self-organizing processes in parallel, so when dealing with large datasets tasks can be divided distributing computational cost.

4. Only three parameters are necessary to the setup of the algorithm, namely: τ_1, τ_2, and φ that respectively regulate: 1) the process in which are added prototypes 2) the process in which is created the hierarchical structure, and 3) the minimal membership degree of the observations to the prototypes.

2.5.3 Tuning parameters in the FGHSON

The training and growth process of the FGHSON is entirely data driven, requiring no prior knowledge or estimates for parameter specification. However, the hierarchical structure of data can be represented in different forms, allowing either: (1) shallow hierarchies with rather detailed quality presented at each subsequent layer or (2) deeper hierarchies, which provide a precise separation of the various subclusters by assigning separate nodes.

In this context, the parameter τ_1 is used to control the trade-off between shallow or deep hierarchies. In the first case, we will prefer larger Fuzzy Kohonen Clustering Networks (FKCNs) in each layer, which explain larger portions of the data, yet providing less hierarchical structuring. As an extreme example, we could consider only one FKCN, which grows in size explaining the complete structure of the data in one single level structure. It ignores all hierarchical information and tries, at best, to preserve it in the mapping of various prototypes on one FKCN. On the other hand, we might consider setting it rather large, which requires only limited growth of individual FKCNs, resulting in a deeper hierarchical structure of small FKCNs (made in few prototypes) focusing on the hierarchical structure.

Basically, the total number of prototypes at the lowest level of FKCNs is expected to be similar in both cases, because this is the number of prototypes at the required level of granularity. Thus, the smaller the parameter τ_1 is chosen the larger the resulting FKCNs will be, explaining its data at a higher granularity. For a larger value of τ_1, more detailed data representation will be delegated to additional FKCNs further down the hierarchy. Thus, the parameter τ_1 serves as the control parameter for the depth/shallowness of the resulting hierarchical FGHSON architecture.

The global stop criterion τ_2, directly influences the overall size of the resulting FGHSON, i.e., the number of units available for data space representation.

The parameter $\alpha - cut$ determines the minimal degree of membership of an observation to be member of a cluster. Hence, $\alpha - cut$ is used as selection parameter and ,obviously, affects the growth process of the hierarchical structure. When $\alpha - cut$ is high, the number of levels is minimal, conversely when $\alpha - cut$ is low the number of levels is maximal. It is because when $\alpha - cut$ is high, the clusters are made of very similar observations, thus the quantization error is easily minimized and the growth process stops. On the contrary, when $\alpha - cut$ is low, more observations are members of a cluster, and usually a new level is necessary in order to reach an optimal representation and minimize the quantization error.

Since the FGHSON is based on the Fuzzy Kohonen Clustering Networks (FKCNs), we also have to take into account the parameters $m0$ (the constant that regulates the fuzzyfication), ϵ (the termination criterion) and t_{max} (number of iterations). I used the following values: $m0$ = 4, ϵ = 0.0001, and t_{max} = 10,000, that are the values recommended in the original paper describing the FKCN [21]. As complementary information, the form of the membership functions is a generalized bell that is defined for the formula of the fuzzy c-means (FCM) used for the FKCN. However, another method to define membership functions using FCMs results, consist of discarding all the membership values and simply using the centers of the clusters to define specific shaped membership functions, such as Gaussian and triangular functions [30].

In conclusion, the selection of the three values of the parameters (τ_1, τ_2, and φ) used to train the FGHSON algorithm can lead to diverse hierarchical structures. For a complementary explanation, the appendix B explores how several combinations of these values affect the topology of the network and the quality of its prototypes.

Chapter 3

Visualization and projection of geospatial data by means of Self-Organizing Maps Component Planes

> "What this means is that we shouldn't abbreviate the truth but rather get a new method of presentation."
>
> Edward Tufte

THe Self-Organizing Map (SOM) [75] has been widely used in exploratory data analysis of geospatial data, specifically for clustering and visualization [6, 5, 119, 58, 76, 110, 7]. It has provided excellent results when dealing with high-dimensional data, datasets with not clear or an unknown distribution, and when a dataset contains several types of data representation (e.g. categorical, continuous, nominal and/or ordinal).

In order to provide more accurate ways for visualizing similarities, patterns, and relationships on the SOM, several techniques have been developed. Two of the most popular techniques are the U-matrix and the SOM component planes [118, 124, 123, 125]. These techniques highlight several characteristics of the SOM, improving the exploratory data analysis and knowledge discovery processes. These techniques allow for instance: to enhance pattern discovery, to help visualize correlations and relationships that can support the hypothesis generation and the detection of irregularities in the data.

In this chapter, techniques for the interpretation of clusters in the SOM are presented in section 3.1. New techniques to improve the analysis of the SOM's component planes are introduced in section 3.2. Finally, in section 3.3 applications in agriculture where the techniques described in previous sections are applied are shown.

3.1 Interpreting clusters in the SOM

As was described in section 2.2.1, at the end of the SOM training, prototype vectors are arranged in a grid creating a structure where prototypes with similar patterns are located nearby. This self-organized structure is very interesting in terms of understanding the observations similarities of a high-dimensional dataset. However, the definition of the groups boundaries in this structure is usually not clear. Thus, clustering techniques are necessary to highlight the groups embedded in the SOM.

An efficient approach to highlight clusters on the SOM is to use distance matrix techniques [123]. These techniques allow to visualize the distance between the prototypes, allowing to identify groups of prototypes with similar patterns (i.e., prototypes with short distance between them). The most widely used technique for this aim is the Unified Distance Matrix representation (U-matrix) [118].

Another manner to analyze similarities between prototypes is observing the distribution of the variables that compose them. Here, the *component planes* play a key role. Each component plane display a the values of a variable, usually the cells (e.g., hexagons or squares) are colored according to the values of the variables. Hence, a visual inspection of the component planes provides an idea of the distribution of the variables (e.g., which prototypes present the maximal and minimal values). They can also be used for "correlation hunting" [125] in order to find relations (e.g, correlations, partial correlations or outliers) between the variables of the prototypes. Thus, correlations between component pairs are revealed as similar patterns in identical positions of the component planes. An example of all the process is presented in 3.1.

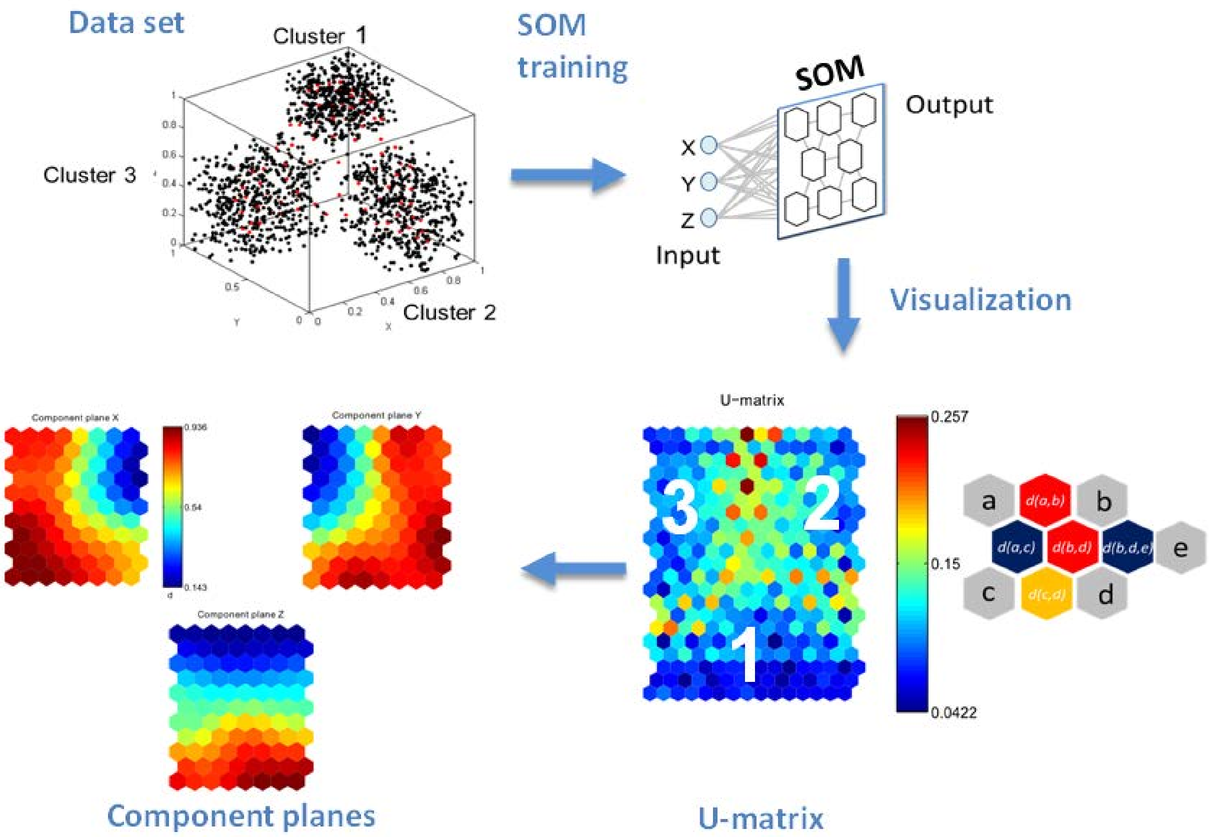


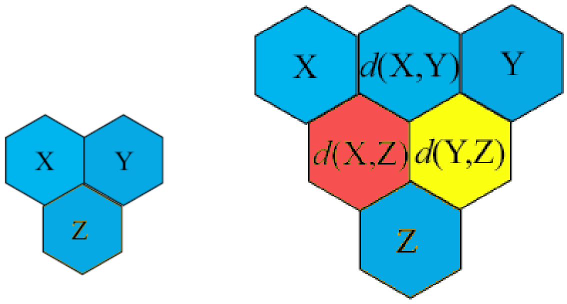
Figure 3.1: Illustration of an application of Self-Organizing Maps to a simple three-dimensional dataset. Here is shown the data and SOM prototypes in input space. Black symbols correspond to data. Red symbols correspond to prototypes. Several SOM visualizations are also shown , the U-Matrix and the component plane for each feature.

A more detailed description of the distance matrix technique and the component planes is presented below.

3.1.1 The Unified Distance Matrix representation (U-matrix)

All clustering algorithms share the problem of deciding the boundaries of the clusters. The same occurs with the SOM algorithm. In this context, Ultsch et al. [118] showed that clusters cannot be detected in a reliable manner using only the two dimensional projection created by the SOM method. In order to allow an easy and straightforward detection of the clusters in the SOM the Unified-distance Matrix, also referred to as U-Matrix was developed[118].

The U-Matrix is a representation of the SOM that visualizes the distances between cells. The idea of the U-matrix is to compute the difference between two adjacent reference vectors and illustrate the difference as an extra cell between the original cells in a U-matrix map. An example of the construction of the U-matrix is given in Figure 3.2. Here, if the distance between the neurons is small, an extra neuron depicting the distance is colored with shades of blue, and if the distance is big with shades of red.

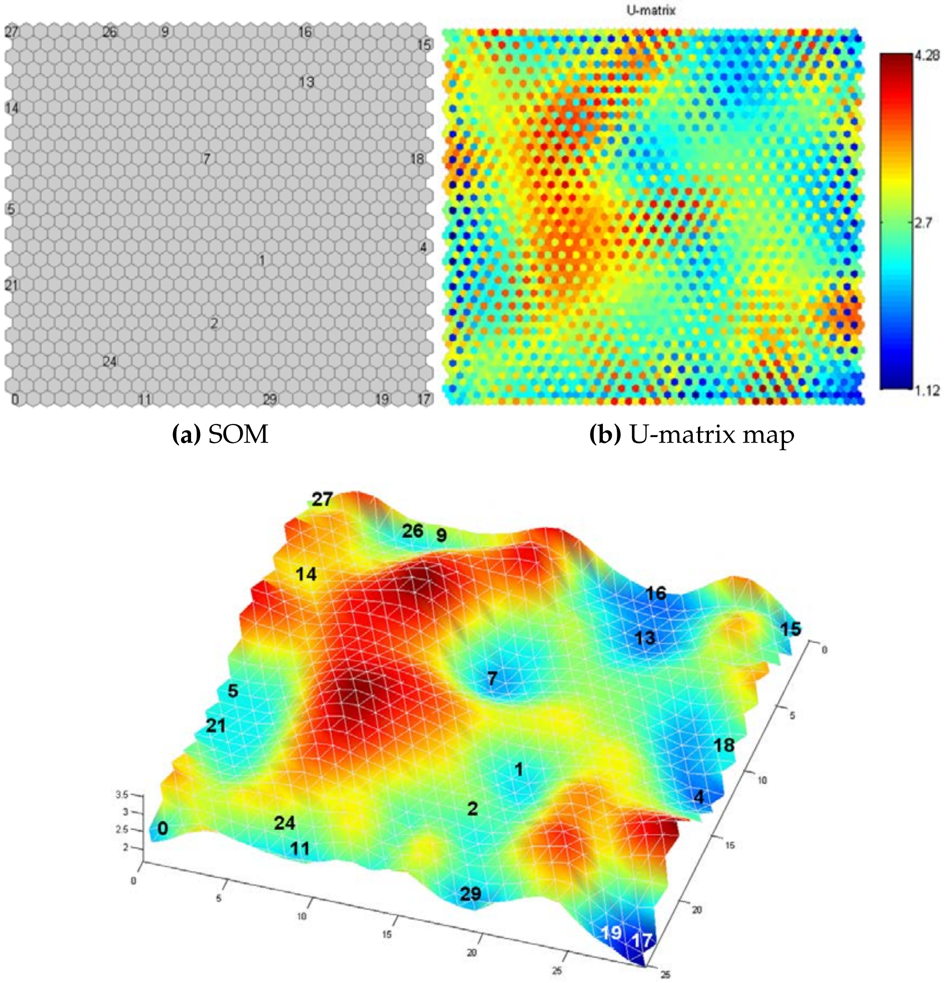


(a) Normal SOM visualization

(b) U-Matrix visualization

Figure 3.2: *Neurons in the two-dimensional graph are colored according to their similarities to adjacent neurons. The result of the U-matrix procedure is an extra element between the neurons. X and Y are similar to each other while Z has a distinct reference vector. Distance $d(X, Y)$ can be chosen arbitrarily, but usually the arithmetic mean is used.*

Another example is shown in Figure 3.3, here clusters can be seen as “valleys” (neurons colored with shades of blue) constrained by “hills” (neurons colored with shades of red). The relative locations of the clusters in the U-Matrix reflects their similarities in the high-dimensional space. The higher the hill (shades of red), the more dissimilar the clusters are.



(a) SOM **(b)** U-matrix map

(c) U-matrix map 3D

Figure 3.3: *(a) The prototypes 13, 15 and 16 located at the top right corner of the map represent vectors with similar patterns. These prototypes are close in the SOM map but when the U-matrix is calculated, shown in (b), a high distance between the couple formed by 16 and 13 compared with the prototype represented by 15 is noted. This new information allows us to conclude that even if the three vector share similar characteristics, 16 and 13 are closer in the high-dimensional space. (c) The U-matrix map represented in a 3D plot highlight the distances between prototype vectors. Here it is possible to observe the "valleys" and "hills" created by the U-matrix representation. Hence, clusters can be seen as "valleys" (neurons colored with shades of blue, for instance vectors labeled with 16 and 13) constrained by "hills" (neurons colored with shades of red) that separate for instance the cluster formed by prototypes 16 and 13 from prototype 15.*

3.1.2 The SOM components planes

The projection created by the SOM allows a straightforward visual inspection of similarities of data because prototype vectors are organized in a low-dimensional grid according to their similarity in the high-dimensional space. A way to improve this inspection is by means of the component planes representation. The relative variables' distributions of the data on the SOM can be visualized by using component planes. The component planes representation is a sliced version of the SOM. Here, each component plane reproduces the relative distribution of one of the variables that composes the prototype vector, as shown in Figure 3.4. In this example, blue neurons represent relatively small values while red neurons represent relatively large values of the variable represented by the component plane. It can be distinguished, as shown in Figure 3.4 for variables $V1$ and $V4$, if two components correlate by comparing component planes. The process of looking for correlations in component planes is called *correlation hunting* [123, 125].

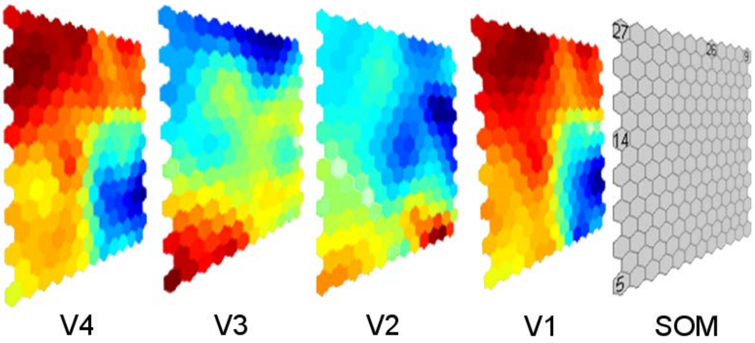


Figure 3.4: *In this example the training data is a four-dimensional dataset, the variables are namely $V1$, $V2$, $V3$ and $V4$. After training, the SOM can be sliced, creating the component planes, in order to show the relative distribution of the variables. In the component planes, blue neurons represent small values while red neurons represent high values of the variable. Information about the relationships between variables, for instance if two components correlate, can be extracted by comparing component planes (e.g.observe the similarity of $V1$ and $V4$). As an example, some prototype vectors were labeled in the SOM for further analysis, in this case prototypes labeled with 14 and 27 in the top left corner. Note that the values for these prototype vectors are respectively: high in V1 and V4, and low in V2 and V3. This kind of analysis allows to discover not only groups of vectors but also the similarities on the variables that compose them.*

3.1.3 Correlation Hunting

The component planes analysis is used to discover relations between variables. Comparing the planes, it is possible to identify analogous patterns in similar positions indicating correlation between the respective components (even local correlations can be found if two parameters resemble each other in some regions), as shown in Figure 3.5. The process of finding these relationships is called *correlation hunting*. Notice that this process does not include just linear correlations, but also nonlinear and local or partial correlations between variables [123, 125].

The correlation hunting can be done manually or automatically. However, in many cases the manual analysis is difficult because usually the component planes are not ordered. In addition, the comparison becomes more difficult when the number of components increases. In order to overcome this drawback, we might think of an way of organizing the component planes so that similar component planes could be located close to each other [124]. One of the most frequently used methodologies consist in projecting the component planes into a plane. The projection could be done using, for instance: Sammon's mapping [104], CCA [40] or another SOM [75].

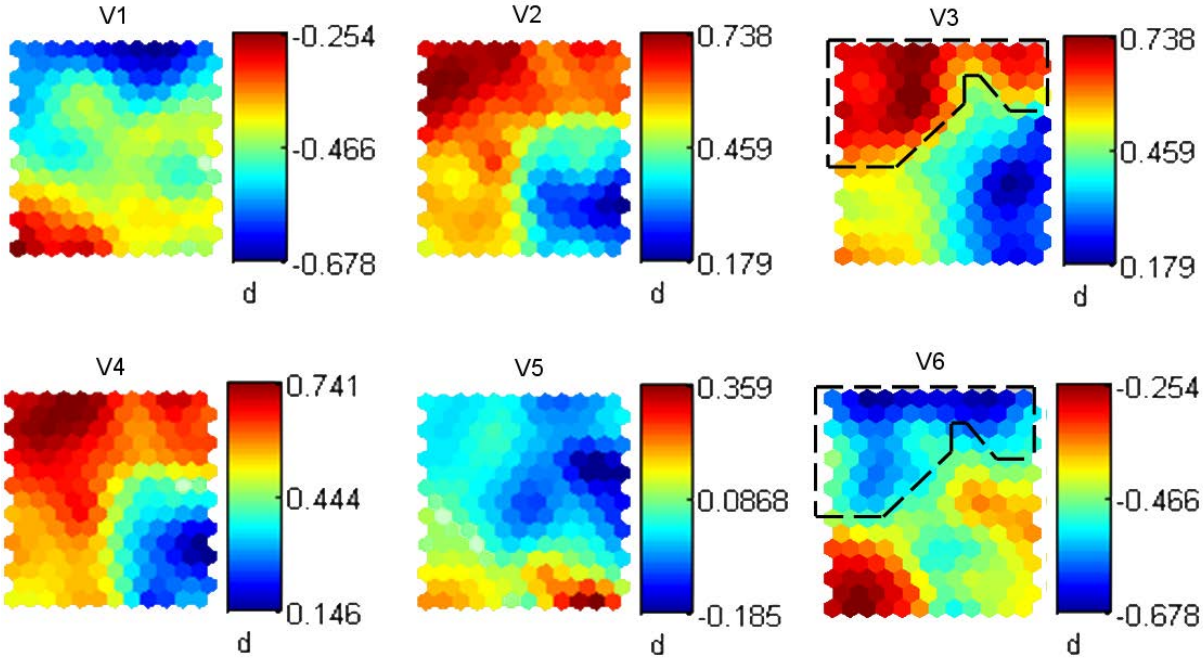


Figure 3.5: *Component planes representation. Note that component planes $V2$ and $V4$ present similar component values indicating correlation between them. On the other hand, a partial negative correlation can be noted between $V3$ and $V6$.*

3.1.4 Correlation Hunting by using SOM as a projection algorithm

The advantages of SOM over other methodologies used for component planes projection are that SOM arranges the component planes on a regular grid and it allows for an ordered presentation of similar components (see Figure 3.6 for an example). On the other hand, an important disadvantage is that the selection of clusters of component planes is left to the user, which is complicated when the number of component planes is large or similarities are not clear.

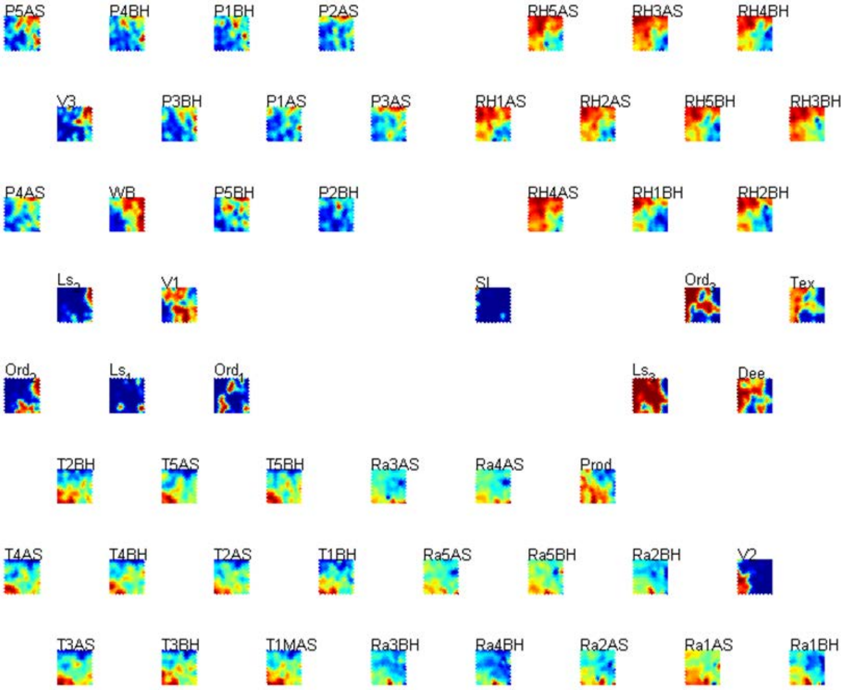


Figure 3.6: *An example of a SOM used to project the component planes. In this case component planes represent agroecological variables described in Section 3.3.1.*

Hence, when SOM is used to project component planes, a clustering technique has to be used. Nevertheless, most of the algorithms used for this aim (e.g. K-means and agglomerative hierarchical clustering) do not take into account the structure created by the SOM. Instead they use the prototype vectors as data input [124, 15].

An approach that seems to be a more natural way to group prototypes is to use a clustering algorithm that takes into account the structure created by the SOM (for instance the distance between prototypes). Thus, as mentioned in section 3.1.1 the U-matrix [118] is used as a more effective clustering approach [126]. The U-matrix is usually applied to select clusters by hand. Nonetheless, this selection is usually subjective because it is based on visual inspection that could change depending on the observer. In this context, Vellido et al. [121] proposed a clustering algorithm by using distance matrix to extract clusters automatically. In this algorithm, the U-matrix is used to identify cluster centers from the SOM. The rest of the map units are then assigned to the cluster whose center is closest. The algorithm is described as following:

1. Local minima of the distance matrix are calculated by finding the set of map units i for which:

   $$f(m_i, N_i) \leq f(m_j, N_j), \forall j \in N_i \tag{3.1}$$

   Where N_i denoted the set of neighboring map units of the map unit i (as shown in Figure 3.7a), and $f(m_i, N_i)$ is some function of the set of neighborhood distances $\left\|m_i - m_j\right\|, j \in N_i$, associated with map unit i. In the experiments, median distance was used. The set of local minima may have units which are neighbors of each other. Only one minimum from each such group is retained.

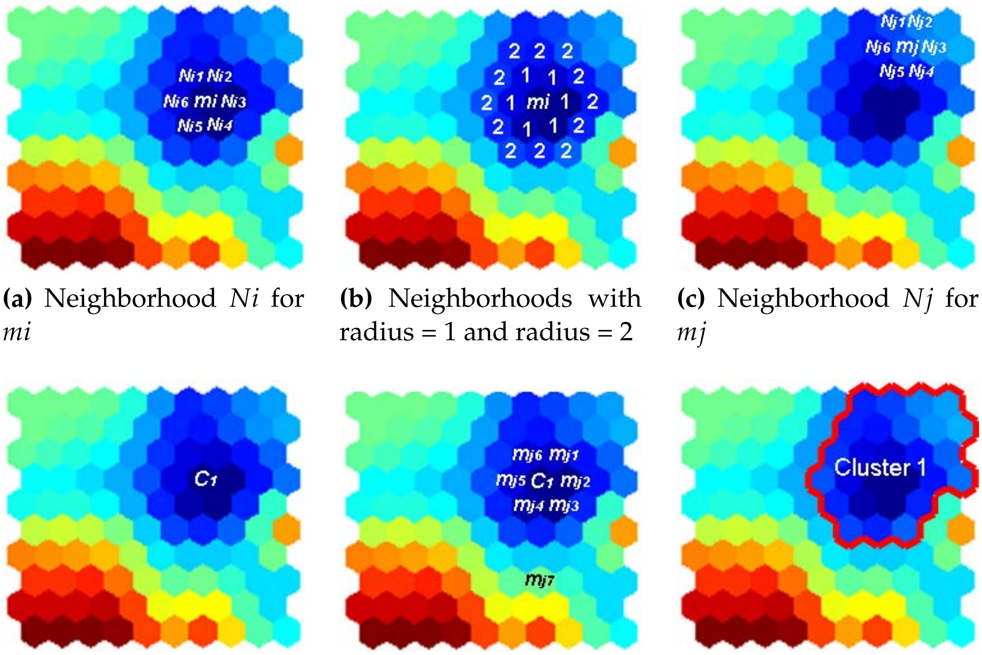
2. For the initialization, let each local minimum be one seed cluster: $C_i = m_i$. All other map units j are left unassigned (as shown is Figure 3.7d).

3. Calculate distance $d(C_i, m_j)$ from each cluster Ci to each unassigned map unit j.

4. Find the unassigned map unit with the smallest distance and assign it to the corresponding cluster.

5. Repeat from 3 until no more connections can be made.

6. If there are any unassigned map units, for example unconnected map units due to the cluster border constraint, they are assigned to the same cluster as the closest neighborhood map unit.



(a) Neighborhood *Ni* for *mi*

(b) Neighborhoods with radius = 1 and radius = 2

(c) Neighborhood *Nj* for *mj*

(d) Cluster center (local minima)

(e) Map units assigned to the corresponding C1

(f) Region of the Cluster 1

Figure 3.7: *Clustering algorithm by using distance matrix.*

This algorithm provides an automatic discrimination of clusters, thus allowing an easier exploration of similar component planes. However, when the amount of component planes is large this approach fails because many planes must be ordered in a grid with limited size. Next section presents an original technique that allows to arrange component planes in a tree-structure, thus improving its exploration when dealing with a large quantity of them.

3.2 The tree-structured component planes representation

This new algorithm uses the idea of component planes projection in a SOM, and the distance matrix based clustering to arrange the component planes in a tree-structured representation improving the correlation hunting [17, 16].

3.2.1 The algorithm

The tree-structured component planes clusters representation uses the Vellido's algorithm to obtain several granuralities of the component planes clusters. In a subsequent stage, these clusters are arranged in a tree-structure. By using the Vellido's algorithm the number of clusters obtained is equal to the number of local minima on the U-matrix. As can be noted in equation 3.1 the local minima depends of the number of neighboring map units (Ni) selected. When Ni is small the probability to find several local minima is high thus the number of clusters increases. Thus, varying Ni it is possible to obtain several cluster granuralities, as shown in Figure 3.8. These clusters and sub-clusters are used in an agglomerative way (arranging subgroups of clusters together) to build a tree-structure as shown in Figure 3.9.

An outline of the algorithm is given below.

1. Tree-generation.

   (a) Calculate the U-matrix of the SOM used for the component planes projection.

   (b) Apply Vellido's algorithm to the U-Matrix. Set the neighborhood radius to 1, and save the results (nodes and component planes clusters).

   (c) Use Vellido's algorithm to partition the map again increasing the neighborhood radius in 1.

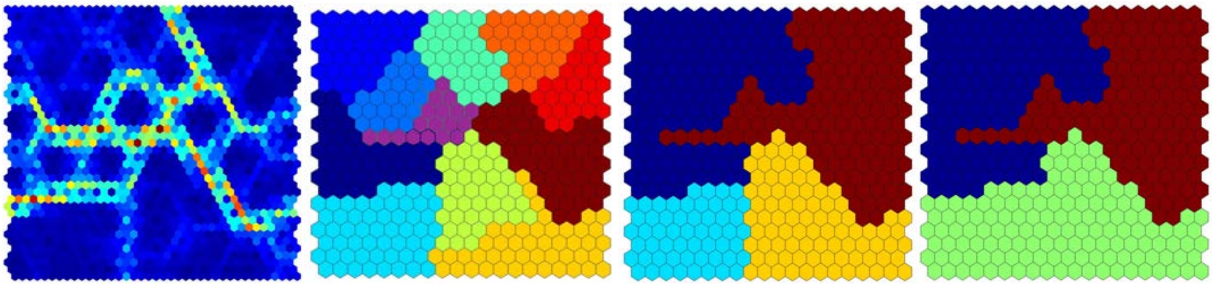
      - If the number of clusters is equal to the obtained in the previous step then repeat c.
      - If the number of clusters is different save the results and repeat c.
      - Stop when the neighborhood radius become the same as the maximal neighborhood radius used to train the SOM that was used for the projection. An examples of this visualization process is shown in Figure 3.9.

2. Visualization.

   (a) Each group of clusters found using a specific neighborhood radius compound a level of the tree, as shown in Figure 3.9.

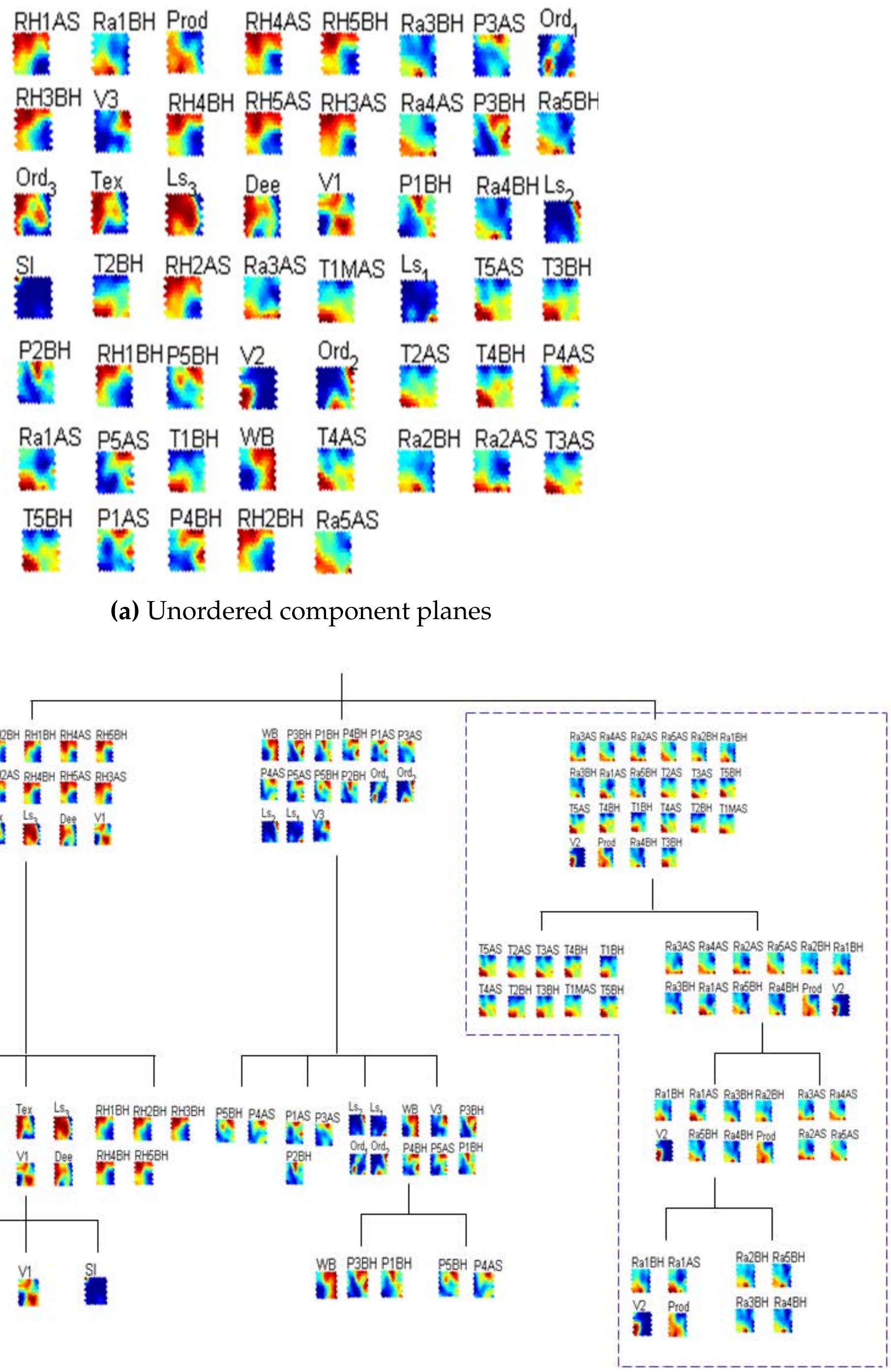
   (b) The group with the lowest number of clusters is taken the basis to construct the first level of the tree.

   (c) Based on the results obtained in the tree-generation step, clusters that were divided into new ones are selected to form the second level. This step is repeated until no more divisions can be made.



(a) U-matrix **(b)** Level 3 **(c)** Level 2 **(d)** Level 1

Figure 3.8: *By using the U-matrix shown in (a) the Vellido's algorithm is applied to obtain different cluster granuralities. (b) By using a neighborhood radius set to 3 we obtain a map with 11 clusters. The third level of the tree structure is formed by the 11 clusters of this map (c) When setting the neighborhood radius to 4, is obtained a map of 4 clusters. Based on this map is built the second level of the tree (d) With a radius set to 5 we obtain a map formed by 3 clusters. The first level of the tree is based on the clusters this map.*



(a) Unordered component planes

(b) The tree-structured component planes representation

Figure 3.9: *Component planes and the tree-structured component planes representation by using agroecological variables. (a) As can be seen, it is not a trivial task to organize by hand similar component planes when there are a large number of them. (b) Here the tree structure allows to arrange similar component planes in the same branch.*

3.3 Applications in agriculture

In this section, the techniques previously described are applied in two agroecological cases where the main goal is to find relationship between agroecological variables and productivity. The first application shows the use of the tree-structured component planes representation to support the hypothesis generation about the relationship between agroecological variables and productivity on sugarcane (Saccharum officinarum). A complete version of this article can be found in the appendix A. In the second case, the SOM component planes and other artificial neural networks methods are used to build production models for Andean blackberry (Rubus glaucus) by using information collected by small-scale growers in Colombia and publicly available meteorological data. The whole article is presented in appendix C.

3.3.1 Improving the correlation hunting in a large quantity of SOM component planes. Classification of agroecological variables related to productivity in the sugar cane culture

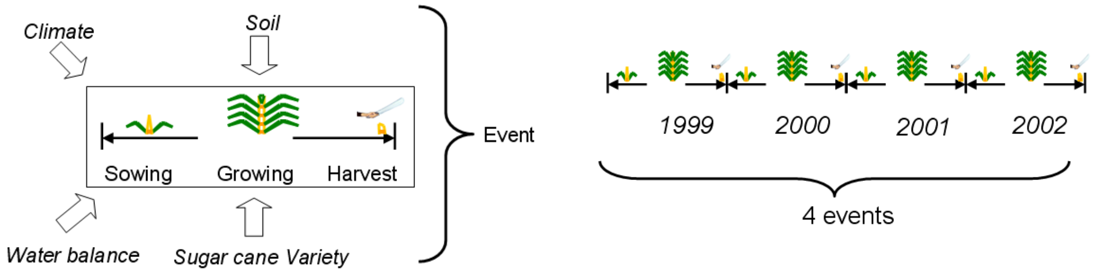
In this work a technique that sinergically combines SOM and the *tree-structured component planes representation* is used to classify agroecological events with similar productivity. Subsequently, the groups obtained are used to find relationships between agroecological variables and high productivity. A more detailed description of the problem is presented below.

Problem description

The agricultural productivity depends on many agroecological variables including soil, terrain characteristics, climatic constraints, human behavior and management. Each agroecological event is unique in time and space (an event is defined here as the period from sowing to harvest, or from harvest to the next harvest, as is illustrated in Figure 3.10). Hence, if several events are grouped and are analyzed, it will be possible to find similar agroecological characteristics between events. Thus, it can be possible to allow discovering why and how agroecological variables affect the crop development, and therefore the productivity.

In a modeling context, each time the same zone is cultivated, one can consider it as a new experiment (see Figure 3.10). This agroecological modeling approach permits to increase the number of observations used to model a certain phenomenon, since the same cultivated zone can provide several events or observations. Nonetheless, it also increases the size of the datasets to deal with. The challenges dealing with this kind of large agroecological datasets include dealing with high-dimensional data, datasets with unclear or unknown data distribution, and datasets containing several types of data representations (e.g. categorical, continuous, nominal and/or ordinal). In this context, techniques based on SOM are appropriate to deal with the presented challenges.

As a case study, we describe the use of SOM and the *tree-structured component planes representation* to group agroecological events with similar productivity, and in a posterior stage, we analyze the variables related with high productivity.



(a) An agroecological event for a sugar cane plantation

(b) Agroecologic events for a sugar cane plantation in four years

Figure 3.10: *(a) An agroecological event is defined as the period from sowing to harvest, or from harvest to the next harvest. Agroecological variables such as soil, climate or water balance have particular effects on the plant at different moments of its development (e.g., germination, flowering). Moreover, the combination and/or change of these variables in specific moments determine the developmental states of the plant. (b) From a modeling point of view, each event can be seen as an experiment or a observation. Every harvest is effectively an unreplicated experiment. If it were possible to characterize the production system in terms of management and the environmental conditions, and if we were able to collect information on the harvested product of a large number of harvesting events under varied conditions, it should be possible to develop best fit models for the production system.*

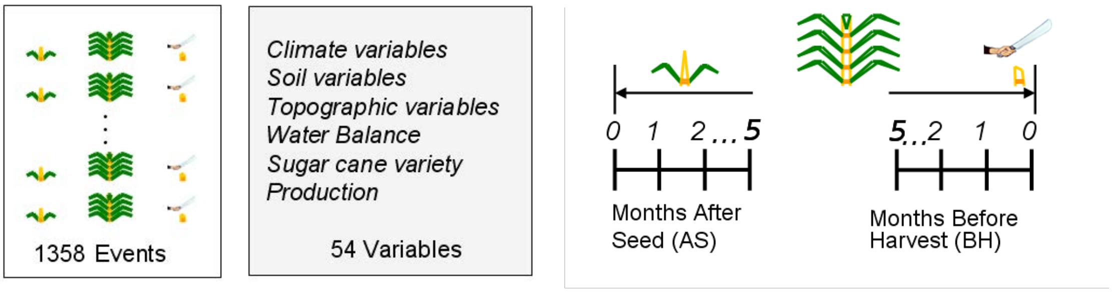
The dataset

The database used was provided by a sugar cane research center (CENICAÑA) located in the region under study. It contains information of agroecological variables from sites were sugar cane is cultivated. The database contains information collected during seven years (1999 to 2005). Considering that the time from the sugar cane development to harvest takes approximatively one year, it means that a maximum six events from a same region were collected. The number of events used as observations was 1328 and the number of associated variables to those events was 54, there are listed in Table 3.1.

Table 3.1: *Variables associated to the zones where sugar cane is cultivated. In the climate group the acronym AS indicates months After Sowing, and BH months Before Harvest.*

Variable	Group	Acronym	Data Type	Classes
Temperature Average	Climate	T	Continuous	AS, BH
Relative Humidity Average	Climate	RH	Continuous	AS, BH
Radiation	Climate	Ra	Continuous	AS, BH
Precipitation	Climate	P	Continuous	AS, BH
Order	Soil	Ord	Nominal	Ord1, Ord2, Ord3
Texture	Soil	Tex	Ordinal	-
Depth	Soil	Dee	Ordinal	-
Landscape	Topographic	Ord	Nominal	Ls1, Ls2, Ls3
Slope	Topographic	Sl	Ordinal	-
Water Balance	Water Balance	WB	Ordinal	-
Variety	Variety	V	Nominal	V1, V2, V3
Productivity	Productivity	P	Continuous	-

As far as the variables related to climate are concerned, only the data from a group of x months after sowing (denoted by xAS) and y months before harvest (denoted by yBH) were used (as it is illustrated in Figure 3.11). This decision is based on the fact that, in the sugar cane production, expert knowledge indicates that the most relevant periods are the beginning and the end of plant development: in the first months (after sowing or harvest) the vegetative structure is formed (e.g., leaves grow allowing the photosynthesis process); during the last months (approximately thirteen months after sowing) the plant accumulates the largest amount of saccharose. In this case study, five months after sowing and five months before harvest were taken into account. Thus, creating a set of ten variables for each climate variable. For instance in the case of the radiation the set is composed of: $Ra1BH$ (radiation-the-first-month-before-harvest) , ..., $Ra5BH$ (radiation-the-fifth-month-before-harvest) and $Ra1AS$ (radiation-the-first-month-after sowing), ..., $Ra5AS$ (radiation-the-fifth-month-after-sowing). The variables related to the soil and to the sugar cane variety were ordered using a presence/absence coding, where 1 represents presence and 0 absence. As a result, the vector which defines an agroecological event is composed of 54 variables.



(a) (b)

Figure 3.11: *(a) The database used contains information of agroecological variables from sites were sugar cane is cultivated. The database contains information collected during seven years (1999 to 2005), it means that from a same region several events are collected. (b) Regarding the variables related to climate, only the data from a specific group of months after sowing and specific months before harvest were used. The schema illustrates the case when three months are taken.*

Training and results

All the variables were scaled between [-1,1] in order to allow a comparison in magnitude. Then, an input matrix with 1328 vectors, corresponding to each event, and its 54 agroecological variables associated was created. Note that the output of this sugar cane model is the productivity. In this experiment the productivity was used as input in order to obtain its component plane to be compared with the component planes of the agroecological variables.

The matrix composed of events and agroecological variables was used to train a SOM of 400 neurons (20x20). The component planes were projected into a new SOM composed of 400 neurons (20x20). Finally, the tree-structured component planes representation was applied to the last SOM, obtaining the structure shown in Figure 3.12.

Analysis

Several interesting observations can be made by analyzing the branches of the tree-structure, where the production plane is located (see Figure 3.12). In the first level it is possible to find in a same cluster the planes associated with temperature, radiation and production. In level 4 radiation-of-first-month-after-sowing, radiation-of-first-month-before-harvest and productivity present similar patterns.

The tree-structured representation facilitates the visual inspection of local correlations in component planes since components with similar patterns are grouped together. As an example, regarding the cluster in level 4 where the production is located (see the dotted clusters in Figure 3.12) it can be noted that when productivity is high most of the values of radiation $Ra1BH$ and radiation $Ra1AS$ are also high. This indicates that high radiation at the beginning and at the end of the sugar cane development plays a key role in the productivity.

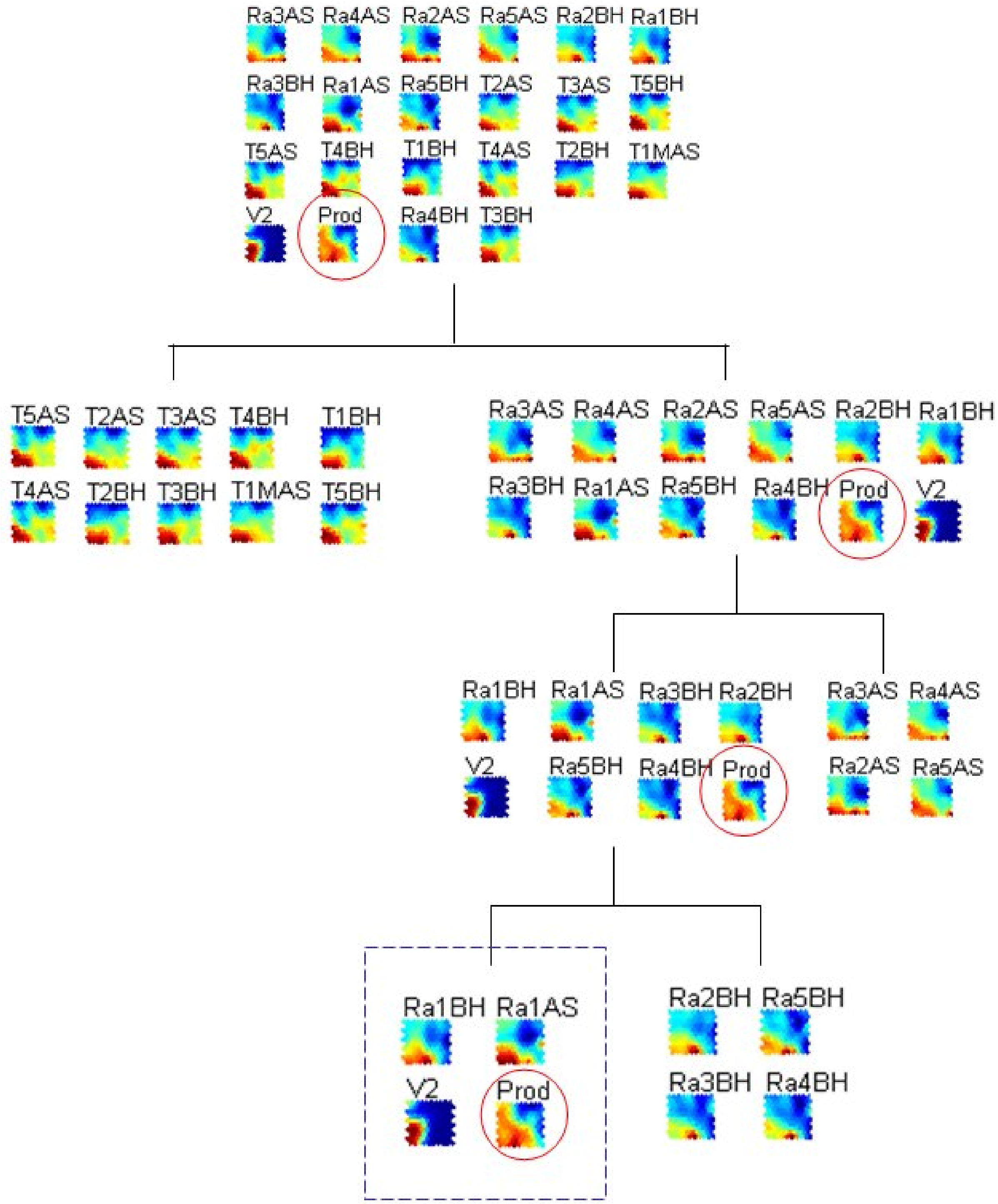

Ra3AS Ra4AS Ra2AS Ra5AS Ra2BH Ra1BH
Ra3BH Ra1AS Ra5BH T2AS T3AS T5BH
T5AS T4BH T1BH T4AS T2BH T1MAS
V2 Prod Ra4BH T3BH
T5AS T2AS T3AS T4BH T1BH
T4AS T2BH T3BH T1MAS T5BH
Ra3AS Ra4AS Ra2AS Ra5AS Ra2BH Ra1BH
Ra3BH Ra1AS Ra5BH Ra4BH Prod V2
Ra1BH Ra1AS Ra3BH Ra2BH
V2 Ra5BH Ra4BH Prod
Ra3AS Ra4AS
Ra2AS Ra5AS
Ra1BH Ra1AS
V2 Prod
Ra2BH Ra5BH
Ra3BH Ra4BH

Figure 3.12: *Productivity (highlighted with a red circle) in each branch of the tree-structure is presented here. In the forth level, radiation-of-first-month-after-sowing (Ra1AS), radiation-of-first-month-before harvest (Ra1BH), sugar cane variety 2 (V2) and productivity (Prod) are grouped in the same cluster.*

By plotting the best matching units (BMU) of the component planes productivity, Ra1BH and Ra1AS in a scatter plot, it is possible to detect high values of productivity when there are high values of Ra1BH and Ra1AS. As a conclusion, the radiation of the first month after sowing and the radiation of the first month before harvest are more correlated with the productivity in comparison with the other variables. In addition, a local correlation is observed between a majority of high values of radiation and high productivity.

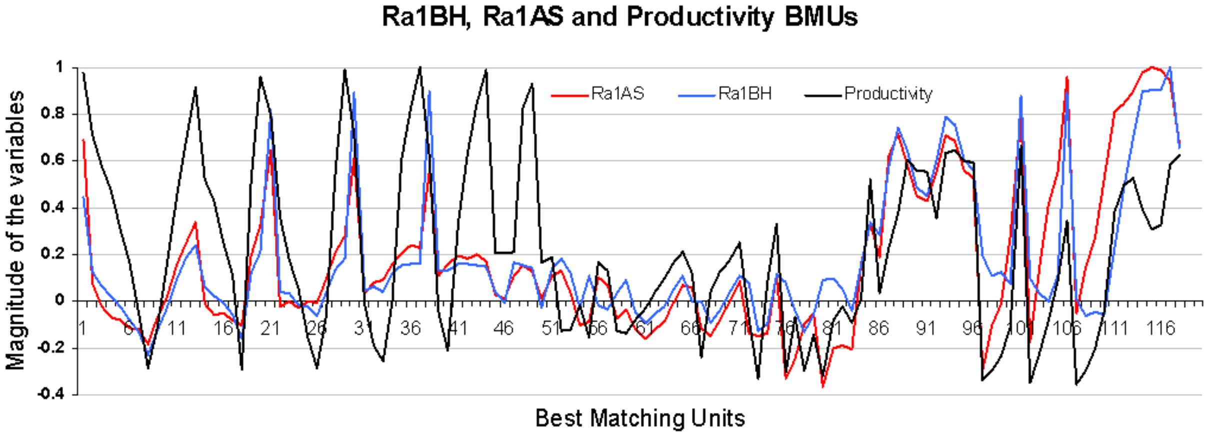


Figure 3.13: *BMUs of the component planes: productivity, radiation 1 month before harvest (Ra1BH) and radiation 1 month-after-sowing (Ra1AS).*

3.3.2 Analysis of Andean blackberry (*Rubus glaucus*) production models obtained by means of artificial neural networks exploiting information collected by small-scale growers in Colombia and publicly available meteorological data

The Andean blackberry (*Rubus glaucus*) is an important source of income in hillside regions of Colombia. However, growers have little reliable information on the factors that affect the development and yield of the crop therefore, there is a dearth of information on how to effectively manage the crop. Site specific information recorded by small-scale producers of the Andean blackberry on their production systems and soils coupled with publicly available meteorological data was used to develop models of such production systems.

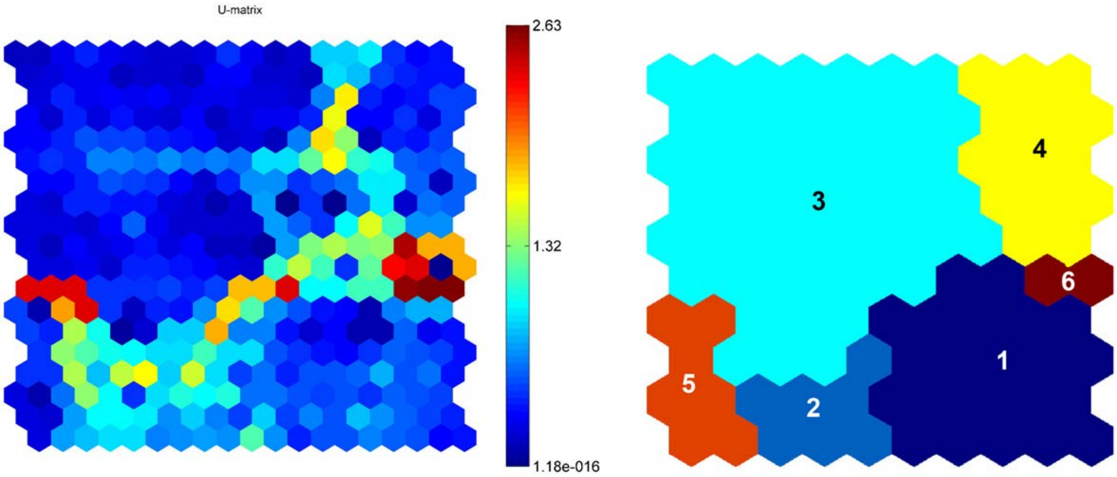
Multilayer perceptrons, Self-organizing maps and SOM component planes were used in the identification and visualization of the most important variables for modeling the production of Andean blackberry. In the present study, a SOM and its component planes were used in order to facilitate the visualization of the relations among the productivity and the environmental and geographical variables, and to establish the ranges of values of these variables, associated with high, medium and low yield.

Modeling and analysis of the variables' relevance

As a first step, artificial neural networks were trained with information from 20 sites in Colombia where the Andean blackberry is cultivated. Multilayer perceptrons predicted with a reasonable degree of accuracy the production response of the crop. In the second part of this study the relevance of the variables obtained by the neural network model were analyzed. We assessed the yield response to changes in the 28 variables used in the model by obtaining the sensitivity of the model output with respect to each one of the inputs. We used the relevance metric based on sensitivity described in Satizábal and Pérez-Uribe [107], which expresses the amount of change of the output with the variations of the inputs. The nine most important variables identified by the sensitivity metric were: soil depth, the average-temperature-of-the-first-month before-harvest, the specific geographical areas Nariño-la-union-chical-alto and Nariño-la-union-cusillo bajo, the average-temperature-of-the-harvest-month, the average-temperature-of-the-second-month-before -harvest, the average-temperature-of-the-third-month-before-harvest, external-drainage and the accumulated- precipitation-of-the-first-month-before-harvest.

Visualization of the relations between the variables found as relevant by the sensitivity metric and clusters with similar productivity of Andean blackberry

To further analyze the effects of the nine variables, a Kohonen map was trained with the same observations we employed to train the multilayer perceptron. The resulting bi-dimensional map is composed of vector prototypes which associate topological information of the original 28 variables with Andean blackberry yield (Figure 3.14a). These prototypes were clustered by using the K-means algorithm [81]. According to the Davies-Bouldin index [37], the map was divided into 6 clusters exhibiting similar features that influence Andean blackberry productivity (Figure 3.14b).



(a) a

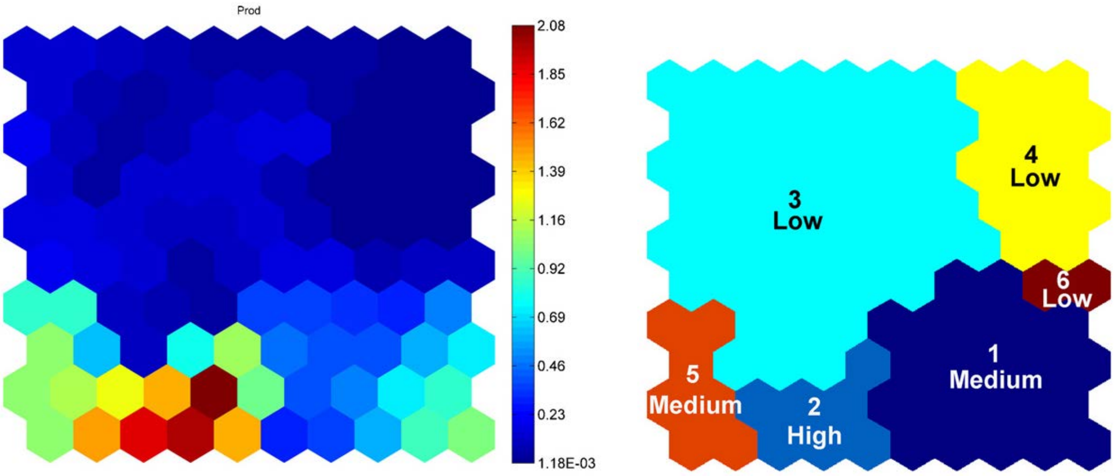
(b) b

Figure 3.14: *Kohonen map showing the resulting clusters. (a) U-matrix displaying the distance among prototypes. The scale bar (right) indicates the values of distance. The upper side exhibits high distances, whilst the lower displays low distances. (b) Kohonen map displaying the 6 clusters obtained after using the K-means algorithm and the Davies-Bouldin index.*

Component planes and variable dependencies

In order to improve the visualization of the dependencies between the clusters shown in the Kohonen map in Figure 3.14b, the component planes of Andean blackberry productivity (Figure 3.15a) and the variables previously identified as the most relevant for modeling Andean blackberry yield: effective-soil-depth (Figure 3.16), the average-temperature-of- the-harvest-month, the average-temperature-of-the-first, second and third months before harvest (Figure 3.17), two specific geographic areas (Figure 3.18 and Figure 3.19) and external drainage (Figure 3.20) were separated from the Kohonen map and displayed as lattices.

- **Productivity component plane**: Yields greater than 1.16 kg/plant/week were associated with regions in cluster 2 on the Kohonen map (Figure 3.15). Yield values between 0.0018 and 1.16 kg/plant/week correspond to clusters 1, 3, 4, 5 and 6 in the Kohonen map. Being 3, 4, 6 the clusters with lowest yields.



(a) a

(b) b

Figure 3.15: *(a) Component plane of Andean blackberry yield, the scale bar (right) indicates the range value of productivity in kg/plant/week. The upper side exhibits high values of yield, whereas the lower displays low values. (b) Kohonen map displaying the resultant 6 clusters and their labels according to yield values.*

- **Effective-soil-depth component plane** : Values of soil-depth greater than 70 centimeters (cm) are associated with clusters 3, 4 and 6 (Figure 3.16) which are all associated with low yields. In contrast, a soil depth between 40 and 70 cm appears to be related to medium to high yield clusters (1, 2, 5). The cluster with the highest yields had soil depths in the range of 60-70 cm suggesting that this level of soil depth is optimal, and that an effective soil depth greater than 70 cm is not necessary to obtain high yields.

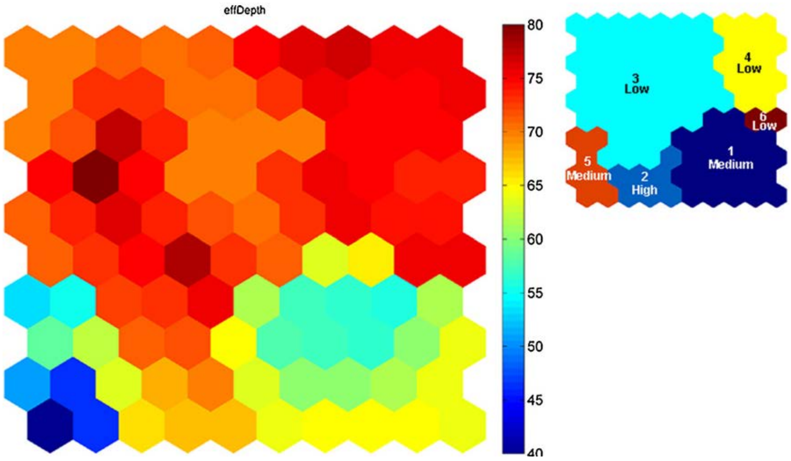


Figure 3.16: *Component plane of effective soil depth. The scale bar (right) indicates the range value in centimeters of soil depth, the upper side of the scale exhibits high values, whereas the lower displays low values*

- **Average temperature of the harvest month and average temperatures of the first, second and third months before harvest component planes**: The component planes for temperature of the first, second and third months before harvest were similar (Figure 3.17). The multilayer perceptron showed that the average temperature of the first month before harvest was more important than the other temperatures (that occurs due to small differences captured to better fit the output). However, the similarity of the components of temperature is probably due to the low monthly variation in temperature under the equatorial conditions of this study. The similarity of the temperature patterns induced us to analyze them as a group rather than separately. It is immediately evident that cluster 6 with temperatures of about 24 °C is not suitable for high yields of blackberries (Figure 3.17). Clusters 1, 2 and 5 with medium to high yields are related to temperatures between 16 and 18 °C (Figure 3.17) and low yields appear to be associated with temperatures in the range of 14-15 °C. Andean blackberry experts suggest the optimal temperature for a healthy growth of this crop is between 11 and 18 °C. We suggest a narrower temperature range with 16-18 °C associated with high yields and lower yields as the temperature moves above or below this range.

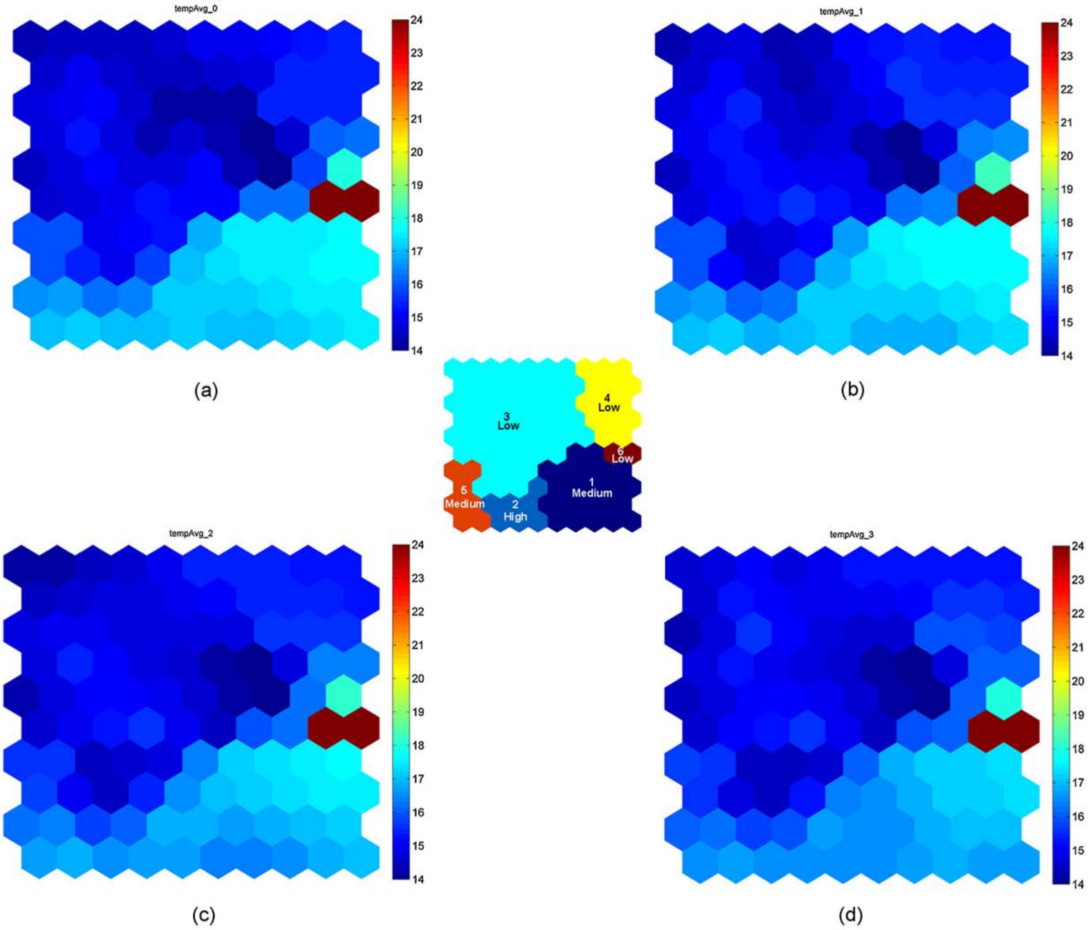


Figure 3.17: *Component planes of the average temperature: (a) temperature of the harvest month, (b) average temperature of the first month before harvest, (c) average temperature of the second month before harvest, and (d) average temperature of the third month before harvest. In all figures, the scale bar (right) indicates the range value in °C of temperature. The upper side exhibits high values, whereas the lower displays low values*

- **Geographic areas as proxy for crop management**: Proxies can be used to estimate the effect of either immeasurable or unobservable variables on a given phenomenon. In our study, geographical areas were integrated into the model with the aim of capturing the effect of variables that were not measured. The geographical proxies were added to the analysis specifically to take into account management and social factors which were not captured by the data collection process and which are likely to be related to the geographic location of a site. For example, farmers from a given locality are likely to use similar management practices that will differ from those used by other communities living in distant localities. The localities Nariño-la union-chical alto (Figure 3.18), and Nariño-la union-cusillo bajo (Figure 3.19) were associated with cluster 2 which is characterized by the highest yields. Whilst the association with high yields could be a consequence of specific local environmental conditions not accounted for by the environmental variables used in the model, we suggest that is more likely that they are due to particular crop management practices related to local knowledge and socio-economic circumstances.

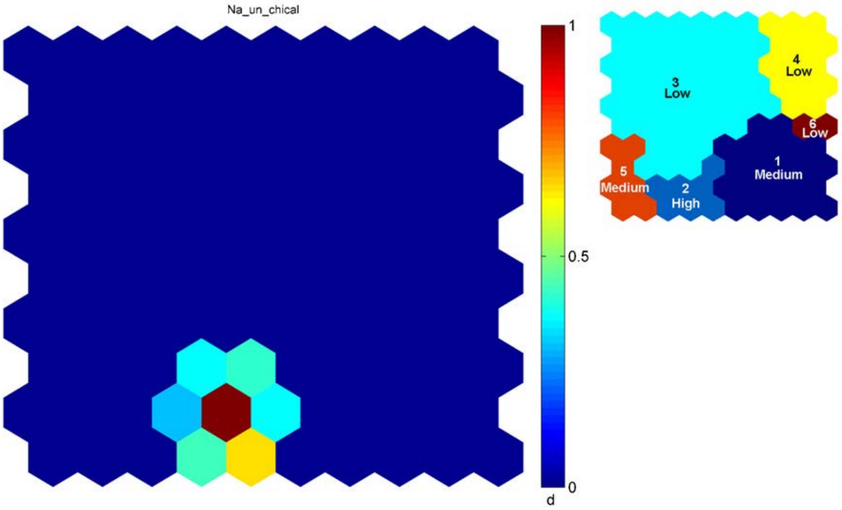


Figure 3.18: *Component plane of the specific geographic area Nariño-la union-chical alto. The highest values indicate presence and the lowest absence, as they are categorical variables.*

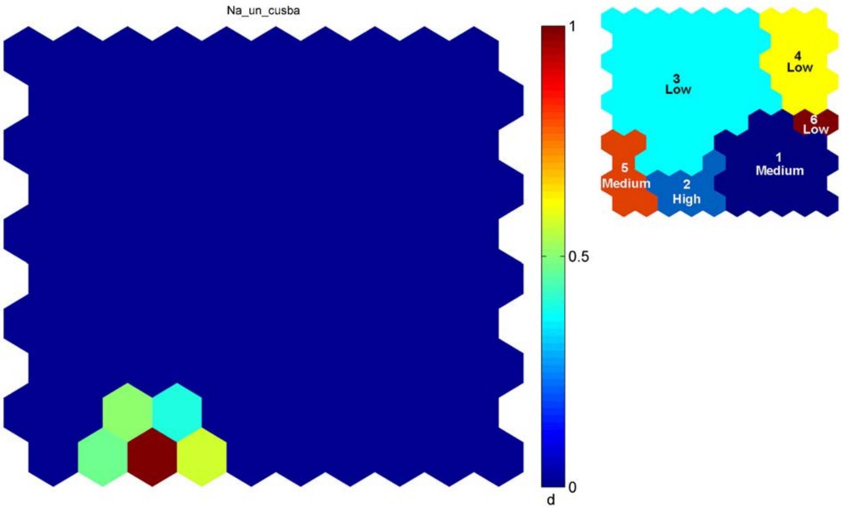


Figure 3.19: *Component plane of the specific geographic area Nariño-la union-cusillo bajo. The highest values indicate presence and the lowest absence, as they are categorical variables*

- **External drainage and accumulated precipitation of the first month before harvest**: Scrutiny of the external drainage lattice (Figure 3.20) gave no obvious clues as to how drainage affects the yield of blackberries. In fact medium yield in cluster 5 is associated with poor external drainage and in cluster 2 with high yields the external drainage is highly variable. However, in all clusters with medium or high yields poor external drainage is associated with low precipitation of the first month before harvest (Figure 3.21): not only does this appear to be true from the component planes, but it also makes agronomic sense. Good external drainage is evidently more important when rainfall is greater. This example clearly indicates how the visual inspection of the the SOM and the component planes can assist in understanding how various factors affect the growth and development of the crop and the interactions between them. Further inspection of Figures 3.20 and 3.21 indicate that excellent external drainage is not sufficient to overcome the effects of high or moderate precipitation with moderate external drainage in cluster 3. Overall, there was a tendency for low rainfall to be advantageous but there were exceptions. However, when the two variables, precipitation of the first month before harvest and external drainage are taken together it is clear that low rainfall accompanied with varied external drainage conditions can provide good yields, but that heavier precipitation of the first month before harvest with poor drainage is not conducive to high levels of productivity.

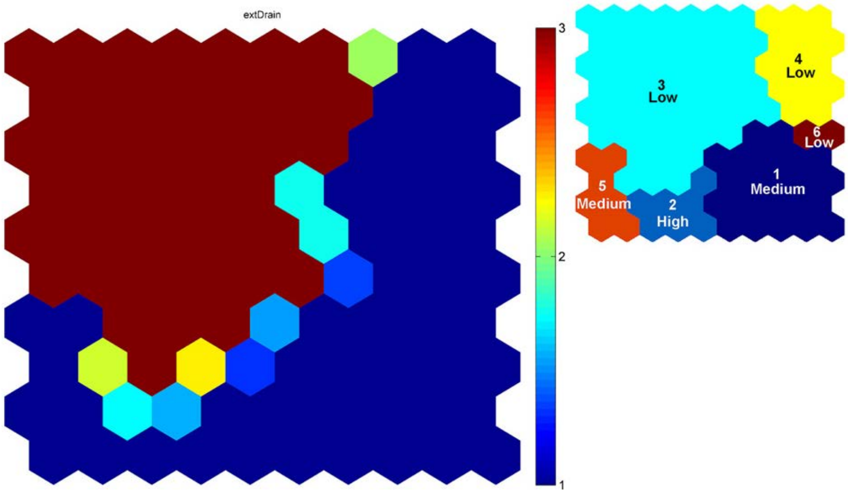


Figure 3.20: *Component plane of external drainage. In the scale bar (right), the highest value 3 indicates excellent or fast drainage, 2 moderate drainage, and 1 poor or slow drainage.*

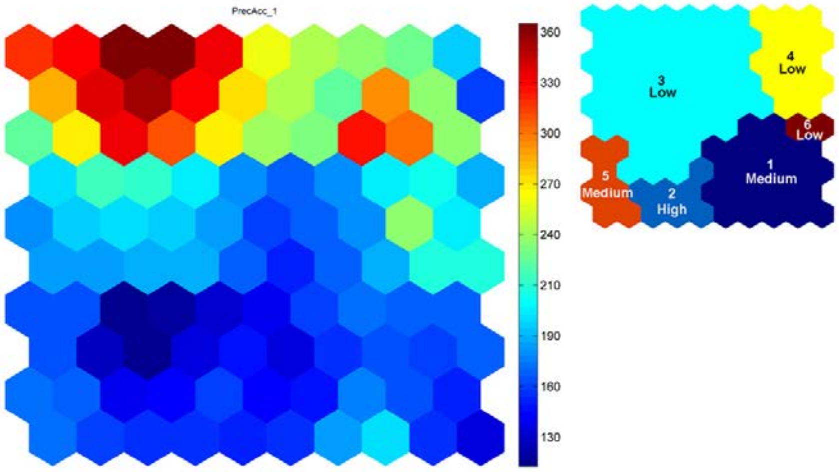


Figure 3.21: *Component plane of the accumulated precipitation of the first month before harvest. The scale bar (right) indicates the range value in millimeters of rainfall, the upper side of the scale exhibits high values, whereas the lower displays low values.*

Conclusions

Data collected by small farmers in the Andes coupled with information from existing data bases was successfully used to characterize specific production events and to relate production to site and time specific events. The analysis approach focused first on identifying those variables that explain most of the yield variability by means of artificial neural networks (multilayer perceptron), and then using the Self-Organizing Maps and SOM component planes as a tool for dimensionality reduction and visualization of input-input and input-output dependencies.

Sensitivity analysis was used to identify the most important variables in determining variation in yield. Self-Organizing Maps were then used to group Andean blackberry yield from different sites according to similarity of growth conditions and management. Data was not available to directly evaluate management practices, so localities were used as a proxy for management. The SOM provided a straightforward manner to visualize the distribution of the variables that affected yield. Component planes generated by SOM illustrated the association of these variables with yield and identified two geographic areas as highly productive. The optimal conditions for high yields are an average temperature between 16 and 18 °C, an effective soil depth between 60 and 70 cm, and low rainfall during the first month before harvest in poor external drainage locations or moderate to low rainfall in better drained areas.

The identification of geographic areas with higher yields than those that would be expected solely from the environmental conditions suggests that the farmers in those geographical areas were managing their crops particularly effectively. However, there was not sufficient information to precisely determine which management factors led to the high yields. At the same time the mere identification of areas with farmers that properly manage their crops, offers the chance for these farmers to disseminate their knowledge to other farmers with similar environmental conditions so that they too can improve yields.

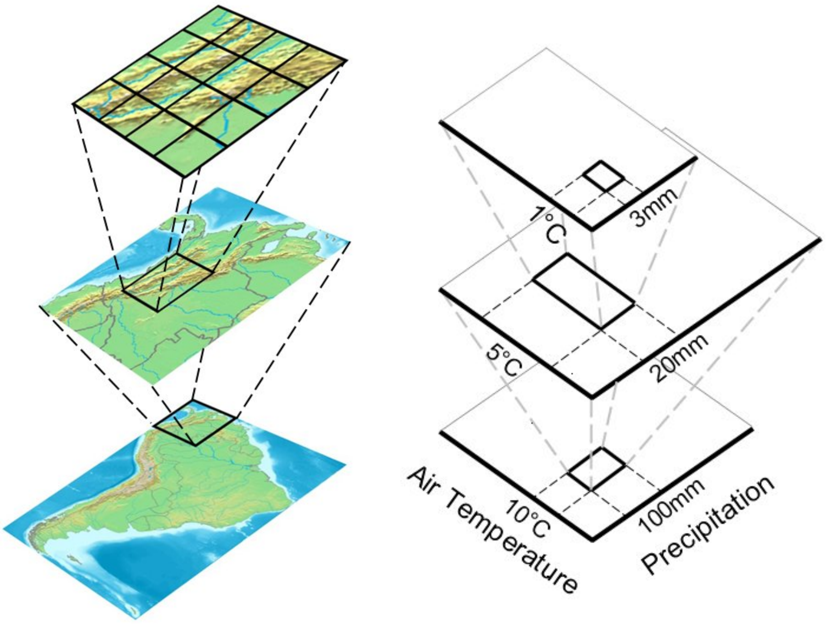
Chapter 4

Clustering of geospatial data by using soft competitive learning approaches

IN many research fields large volumes of geospatial data are regularly captured, stored and continually updated. The value of this information depends on the ability to extract useful knowledge from these data [36, 100]. A key issue in the exploratory data analysis of geospatial data concerns the development of efficient techniques for clustering and visualization. These techniques allow for instance: to help discover relationships among variables (i.e., between climatic variables and productivity of a crop), to find zones with similar patterns (e.g, zones with analogous environmental patterns), and data abstraction [132, 100, 101] (i.e, to extract a simple and compact representation of a dataset). However, several challenges must be faced when working with geospatial data, namely:

- **The visualization of clusters in both geographic and feature space.** The geographic space refers to the bi-dimensional cartographic representation of a geographical area. The feature space is the n-dimensional space of variables associated to the georeferenced points (e.g. temperature, precipitation and solar radiation). Although the geographic space and the feature space are strongly related, they present several differences. Regarding the visualization of zones with similar patterns, it is simpler to visualize them in a geographic space since they can be projected in two dimensional map. However, visualization of similar patterns in the feature space is not trivial when the number of variables are more than three, in this case we have to deal with a high-dimensional visualization problem.

- **The fact that of geospatial data can be represented at several resolutions.** Regions with similar characteristics can be visualized at several scales in both the geographical space and the feature space. In the geographical space, different granularities can be adopted, so the region of interest can vary in resolution depending on the application or the context (e.g., countries, states, cities, parcels) as shown in Figure 4.1a. Concerning the feature space, several variables' ranges can be used. For instance, in Figure 4.1b the couple temperature-precipitation is represented at several levels.



(a) **(b)**

Figure 4.1: *Several granularities in both geographic and feature space can be used in order to look for appropriate representations of clusters. (a) In the geographical space. (b) As an example of several data representation in the feature space, we can imagine that we are looking for clusters with a temperature range of 10 °C , and precipitation range of 100 mm, then to find this resolution we have to move at the first level of the hierarchy. At the same time, we can observe a smaller ranges if we move in the hierarchy until the third level, where ranges are of 1 °C for temperature and 3 mm for precipitation.*

- **The handling of fuzzy boundaries in geospatial clusters.** Geographical areas with similar characteristics (e.g. climate, soil, landscape) are often represented with rigid boundaries. However, geografical boundaries do not necessarily correspond to such rigid frontiers, [78].
- **The large quantity of data.** Usually when dealing with geospatial datasets we have to work with extensive geographical regions that are represented by a large quantity of points (e.g., a common dataset contains millions or even more observations)
- **The unlabeled data.** In many cases, classification must be done without a priori knowledge of the classes in which are divided the dataset (unlabeled pattern).

Recently, soft competitive learning algorithms have been used with a great level of success in geosciences [5, 6, 7, 58, 76]. This success is due to their capability to tackle the aforementioned challenges when clustering and visualizing geospatial data. For example, these algorithms are very useful in the following tasks:

- **Discovering natural clusters in unlabeled datasets**. Many clustering algorithms have been developed with this aim, however the continuous updating and the large volumes of data available today make it difficult to label observations without previous knowledge of the datasets. In this context, soft competitive learning algorithms allow the grouping of similar observations without having an a-priori knowledge of the group or class to which they belong;

- **Reduction of information redundancy contained in the data**. They create prototypes of the observations of a dataset. Hence, large datasets can be reduced without, or with a minimal, loss of information. Data abstraction is useful when working with large datasets, as is the case in agroecology. Using data abstraction large datasets can be represented by a few prototypes. Therefore, data abstraction help to improve representation, interpretation and processing of geospatial datasets.

- **Visualization of high-dimensional data** The high-dimensionality of the geospatial data renders difficult the task of discovering patterns among variables. Several soft competitive learning algorithms allow the projection of high-dimensional space in a two dimensional grid. Thus, allowing the visual exploratory analysis of data, facilitating the discovering of non linear, local, or partial correlations;

- **To work with data with unknown distribution**. Many clustering algorithms have been developed to deal with certain data distribution (e.g. Gaussian distributions). Usually when working with geospatial data it is not possible to have an a-priori knowledge of the data distribution, since in many cases the data sets are updated constantly. Thus, the continuous updating of observations and variables renders difficult to make assumptions about the data distribution. Soft competitive learning algorithms are very useful when working with geospatial data because they do not need to assume any data distribution.

- **Hierarchical representation of spatial and temporal relationships**

  Hierarchical representation techniques are used to reveal the inner organization of spatio-temporal datasets, providing very informative description and visualization of data structures. These representations enable the analysis of similarities in the observations as well as the exploration of relationships and patterns at several resolutions [101].

- **Fuzzy representation**

  Geographical areas with analogous agroecologic conditions (e.g. climate, soil, landscape) are often represented with static and rigid boundaries because the techniques used for this aim are usually hard clustering algorithms. These algorithms allocate each observation to a particular cluster, creating rough boundaries as shown in Figure 4.2 (a). However, agroecological boundaries do not necessarily correspond to such rigid frontiers. A fuzzy representation, where one region can belong to several clusters with a membership level, as shown in Figure 4.2 (b) might be more appropriate. Therefore, delimitation of zones with similar agroecologic conditions using fuzzy logic techniques are being recently adopted as a common practice [78]. In summary, it is essential to integrate a fuzzy representation as an additional feature to the techniques used for clustering spatio-temporal geospatial data.

In this chapter we explore three algorithms: the Self-Organizing Maps (SOMs) [75], the Growing Hierarchical Self-organizing Map (GHSOM) [98], and the Fuzzy Growing Hierarchical Self-organizing Networks (FGHSON) [18], in order to find analogous environmental zones in Colombia. This application serves as a basis to show the advantages and some disadvantages of soft competitive learning algorithms when dealing with geospatial data.

Section 4.1 presents the SOM, and GHSOM algorithms applied to the problem of finding analogous environmental zones in Colombia. Finally, in section 4.2, the FGHSON algorithm is presented as an alternative to tackle some of the limitations of SOM and GHSOM when clustering geospatial data.

4.1 Unsupervised soft competitive learning algorithms for clustering and projection of geospatial data

Three soft competitive learning algorithms are used in this section to find analogous environmental areas in Colombia. Two of them, namely the SOM and the GHSOM have been widely used in many fields proving to work effectively when large and high-dimensional datasets have to be clustered [58, 76, 58, 5, 6, 7], and relationship between observations visualized. The third algorithm is the FGHSON which is presented as an option to tackle several limitations of the SOM, and GHSOM when working with geospatial data. The results obtained after training the algorithms with the same dataset are presented at the end of the sections. In addition, its advantages and disadvantages when dealing with geospatial data.

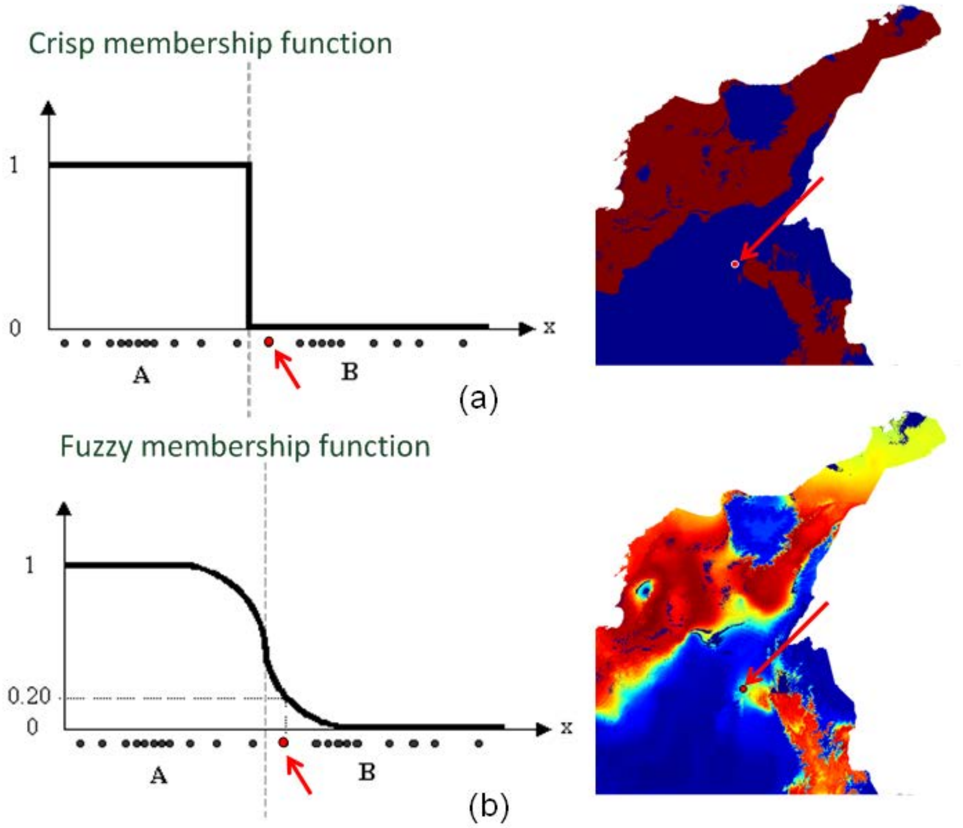


__Figure 4.2:__ Crisp and fuzzy representations. This example shows two representations, crisp and fuzzy, of a same cluster. This cluster gather together zones with similar air temperature and precipitation. In (a) Shown a crisp representation. Here, all the points belong to the cluster and have the same membership value (1 or 0), hence they are creating rigid boundaries when the cluster is projected in the geographical space (as can be observed in the map). In the fuzzy representation illustrated in (b), each point has a membership value depending on how close it is to the cluster prototype. The colors in the fuzzy representation correspond to different degrees of membership of the observation to the given cluster, i.e. red shades represent a high membership and blue shades a low membership. This representation allows to create more natural boundaries when the cluster is projected in the geographical space

The dataset

I used the same dataset for all the algorithms. This dataset contains $1,336,025$ georeferenced points covering all the Colombian geography, and 19 bioclimatic variables (shown in Table 4.1) associated to each point. The resolution of each point is one kilometer by one kilometer.

Table 4.1: *Variables used as input to the SOM, GHSOM, and FGHSON algorithms. These bioclimatic variables are derived from the monthly temperature and rainfall values in order to generate more biologically meaningful variables. They represent annual trends (e.g., mean annual temperature, annual precipitation), seasonality (e.g., annual range in temperature and precipitation) and extreme or limiting environmental factors (e.g., temperature of the coldest and warmest month, and precipitation of the wet and dry quarters). [60]*

Acronym	Variable
V1	Annual Mean Temperature
V2	Mean Diurnal Range Temperature (Mean (period max-min))
V3	Isothermality (V2/V7)
V4	Temperature Seasonality (Coefficient of Variation)
V5	Max Temperature of Warmest Period
V6	Min Temperature of Coldest Period
V7	Temperature Annual Range (V5-V6)
V8	Mean Temperature of Wettest Quarter
V9	Mean Temperature of Driest Quarter
V10	Mean Temperature of Warmest Quarter
V11	Mean Temperature of Coldest Quarter
V12	Annual Precipitation
V13	Precipitation of Wettest Period
V14	Precipitation of Driest Period
V15	Precipitation Seasonality (Coefficient of Variation)
V16	Precipitation of Wettest Quarter
V17	Precipitation of Driest Quarter
V18	Precipitation of Warmest Quarter
V19	Precipitation of Coldest Quarter

4.1.1 Self-Organizing Maps (SOMs)

Since the dimensionality of the datasets is high, it is often a challenge to search for representative patterns. To tackle this problem, a SOM can be used to project high-dimensional input data into an alternative low dimensional space (usually a grid of two dimensions). This projection is performed based on regularities, similarities and frequencies of the input data, which projected in a low dimensional space, can aid the search for interesting patterns [112].

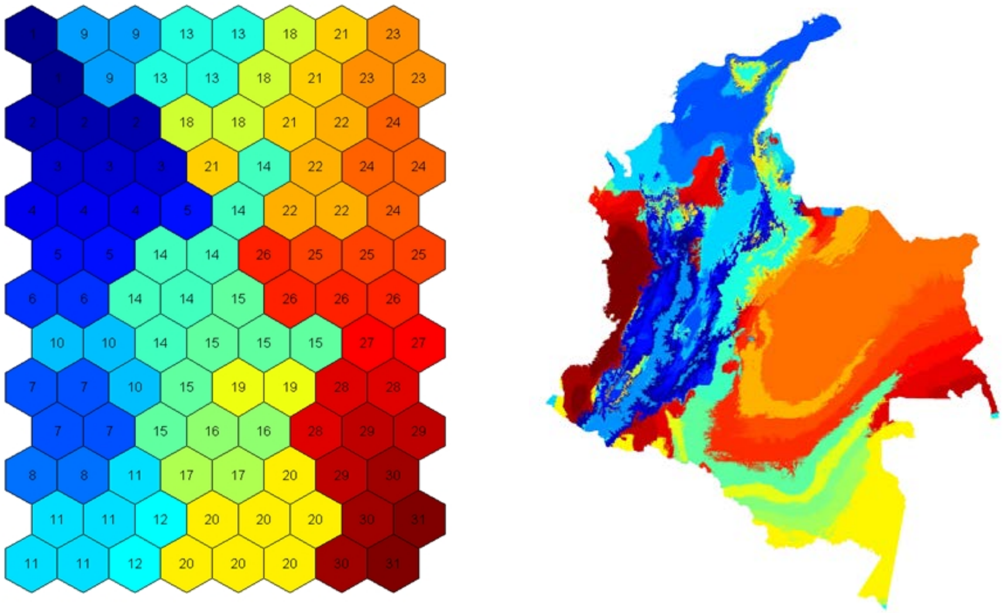
Training parameters and results

A SOM was trained setting its parameters with the values shown in Table 4.2, and using as input the dataset described at the beginning of this Chapter.

Table 4.2: *SOM parameters. The SOM training is divided in two phases. The first phase is the rough training. Here map units are adapted to obtain an approximate representation of the data. It utilizes a large neighborhood radius. The second phase is the fine-tunning phase that uses a small neighborhood radius for more precise adjustment of the map to the data. In both of these phases the neighborhood radius decreases as the training progresses.*

Parameter	Value
Map size	13 x 8
Lattice	Hexagonal
Shape	Rectangular
Training Type	Batch
Neighborhood	Gaussian
Rough training radius initial	3 map units
Rough training radius final	1 map unit
Rough training length	1 epoch
Finetune training radius initial	2 map units
Finetune training radius final	1 map unit
Finetune training length	20 epochs

After training, the prototypes arranged in the SOM grid must be grouped in clusters. However, SOM, as in many clustering algorithms, share the problem of deciding boundaries of the clusters. In order to address this problem, standard clustering methods are used to cluster the prototypes [123] of the SOM grid. For this aim, the K-means algorithm was used to group the prototypes into a given number of K clusters. Nonetheless, one of the limitations of K-means is that the number of clusters has to be defined a-priori. To tackle this drawback, different K values were tested. An optimal value of $K = 31$ was then derived using the Davies-Bouldin index [37]. The 31 clusters obtained are shown in Figure 4.3.



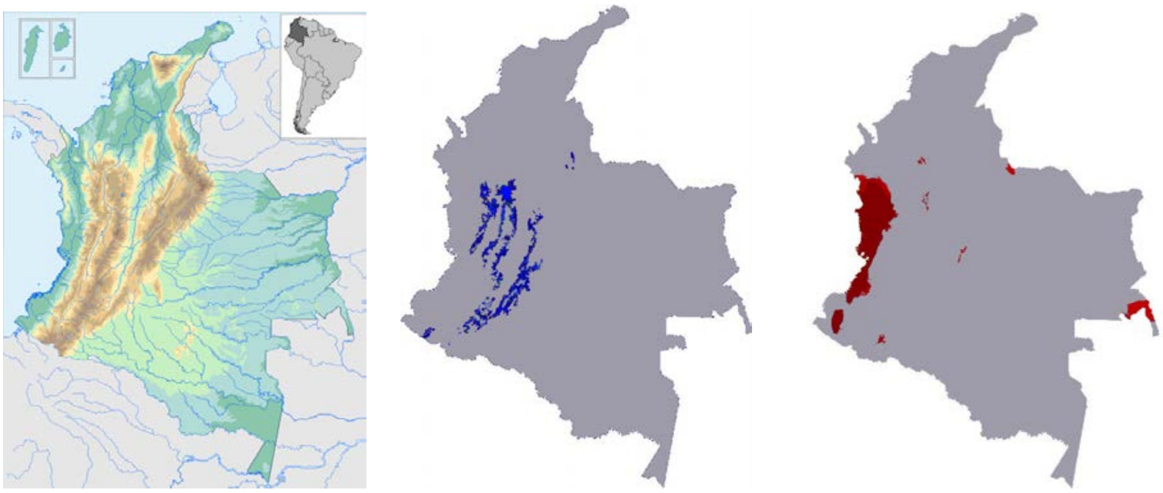
(a) SOM prototypes clustered using K-means and the Davies-Bouldin index [37]

(b) The clusters obtained map of clusters displayed in the geographic space. The same color code used to represent the clusters in the SOM grid was used in the geographic space.

Figure 4.3: *Clusters of analogous environmental zones in Colombia using 19 bioclimatic variables.*

Analyzing the results

In order to show the visualization properties of a SOM in exploratory analysis tasks, the clusters 1 and 2 (upper left corner in Figure 4.3a), and 30 and 31 (bottom right corner in Figure 4.3a) were selected for further analysis. These clusters in the geographical space are shown in Figure 4.4. The clusters 1 and 2 are located mainly in the mountains, as can be seen by comparing the elevation map of Colombia shown in Figure 4.4a with clusters 1 and 2 in Figure 4.4b. On the other hand, clusters 30 and 31 are mainly located in regions close to the Colombian Pacific coast. Even if elevation is not used as an input variable in our clustering test, the elevation map shown in Figure 4.4a can give an idea of where clusters are located in geographic space.



(a) Colombia elevation map **(b)** Clusters 1 and 2 **(c)** Clusters 30 and 31

Figure 4.4: *Four clusters representing analogous environmental zones in Colombia. Elevation map is shown as a guide to locate the clusters in the geographical space.*

The Unified Distance Matrix representation (U-matrix) [118] and the SOM component planes [75, 123] are used to do the exploratory analysis in the feature space. Both are shown in Figure 4.5. In this context, the U-matrix allows the examination of the overall cluster patterns of the input dataset. On the other hand, the component planes allow for a visual exploratory analysis of the clusters in the feature space.

By observing the U-matrix, it is possible to note that clusters 1 and 2 (upper left corner in the U-matrix), represent particular patterns which are indicated by the large distance between them and its neighborhoods (note the red hexagons indicating a large distance between the clusters 1 and 2 and the prototypes surrounding them). Similarly, in clusters 30 and 31 (bottom right corner in the U-matrix), the distances between those clusters and its neighbors are also high, which indicates that the patterns of these clusters are very different to those of their neighbors. In a visual exploratory analysis, the U-matrix allows to highlight clusters that present particular patterns.

The component planes allow to explore variables in order to look for relationships or patterns among clusters since neurons are colored according to the relative values of the variables that they represent. Thus, maximal and minimal values, particularities, similarities, or partial correlations can be easily visualized. As an example, clusters 1, 2, 30 and 31 are highlighted in the component planes shown in Figure 4.5. For instance, consider the component plane of the variable V12 (Annual Precipitation). In this case, clusters 1, 2, present the lowest values of V12. On the contrary, clusters 30 and 31 present the highest values. A similar behavior can be found for V13 (Precipitation of Wettest Period).

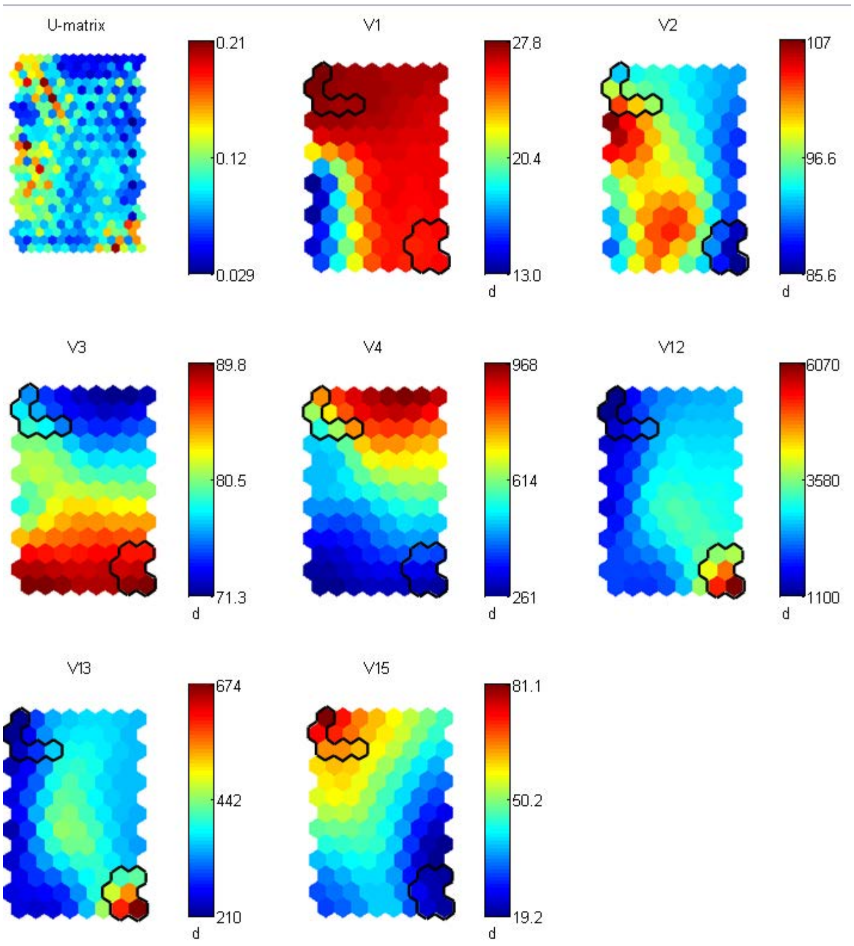


Figure 4.5: *The U-matrix (upper left corner of the Figure) and the component planes of variables V1, V2, V3, V4, V12, V13 and V15 described in Table 4.1. The placement of clusters 1 and 2 (upper left corner of the component planes), and clusters 30 and 31 (bottom right corner of the component planes) in the Self-organizing map are highlighted in order to show the value of the variables associated with them.*

4.1.2 Growing Hierarchical Self-Organizing Map (GHSOM)

The GHSOM [98] allows overcoming some limitations of the SOMs, for instance: its fixed architecture and the exploration of the SOM at multiple levels of detail. The GHSOM creates hierarchical topologies that are able to adapt to the data distribution. This hierarchical representation provides a convenient interface for exploration and visualization of the structures of the clusters.

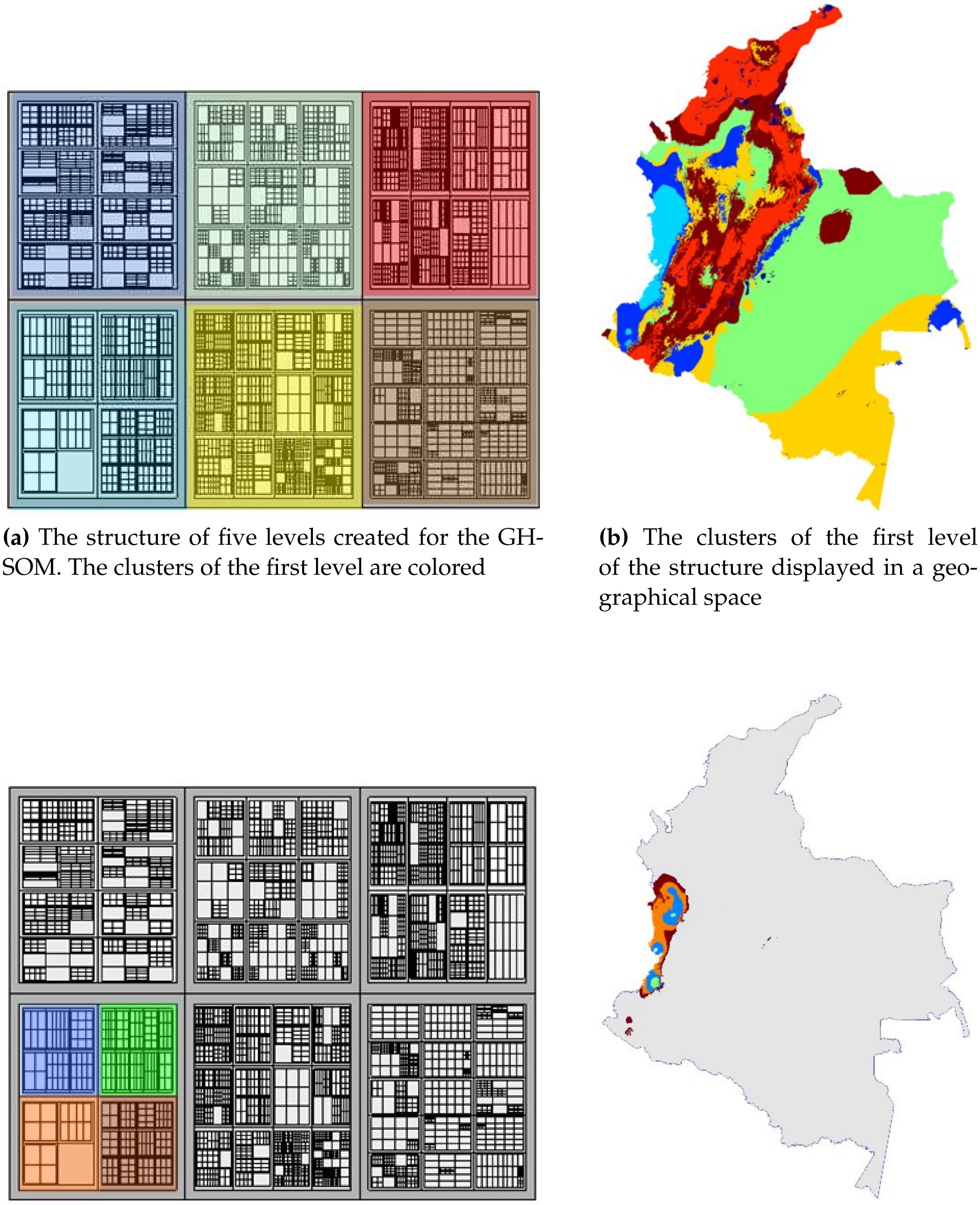
When working with geospatial data, GHSOM allows to evaluate regions with similar patterns at different levels of abstraction [106].

Training parameters and results

With the objective of finding similar environmental regions in Colombia, a GHSOM was trained using the parameters $\tau_m = 0.3$ (which is a fixed percentage that control the growing process of the SOMs on each layer), and $\tau_u = 0.03$ (Which is a parameter used to describe the desired level of granularity in input data discrimination in the final maps). The input dataset used to training the GHSOM was presented at the beginning of this chapter. After training, a hierarchical structure of five levels was obtained, as in shown in Figure 4.6a. In this structure, each cluster is represented by a box. The bigger boxes are split into smaller boxes, hence creating levels. For instance, in Figure 4.6a, the first level is composed of six clusters but then each box is further composed of multiple boxes of lower levels. They are also displayed in geographic space in Figure 4.6b, maintaining the same color code shown in the hierarchical structure.

Analyzing the results

In this first level it is possible to distinguish clear analogue environmental zones. For example: areas of the eastern plains (colored in green), the Amazon region (colored in yellow) and the Pacific Coast (colored in light blue). However, these areas are very large for a detailed study of similar environmental zones. Hence, taking advantage of the hierarchical representation of clusters one can move from one level to the following level in order to obtain a more detailed granularity.



(a) The structure of five levels created for the GHSOM. The clusters of the first level are colored

(b) The clusters of the first level of the structure displayed in a geographical space

(c) Clusters of the second level for a selected first level cluster.

(d) Clusters of the second level represented in the a geographical space.

Figure 4.6: *The hierarchical structure created for the GHSOM. One of the clusters of the first level was selected as an example in order to show the navigability of the structure*

As an example, the cluster located at the bottom left of the map (colored in light blue), is selected for further analysis. This cluster is composed by four sub-clusters, as shown in Figure 4.6c. The cluster selected represents similar environmental zones in the Colombian Pacific Coast area as shown in Figure 4.6 d. In order to analyze the characteristic patterns of the environmental variables of this area, the component planes of the GHSOM are displayed in Figure 4.7. Here, it is possible to observe the 19 climate variables, described in Table 4.1, and their values for each cluster. For instance, note that the cluster under study presents the higher values for V1, V17, V18 and V19 (these are variables related with precipitation).

In conclusion, the GHSOM presents several characteristics that make it very useful for clustering and exploratory analysis when dealing with geospatial data.

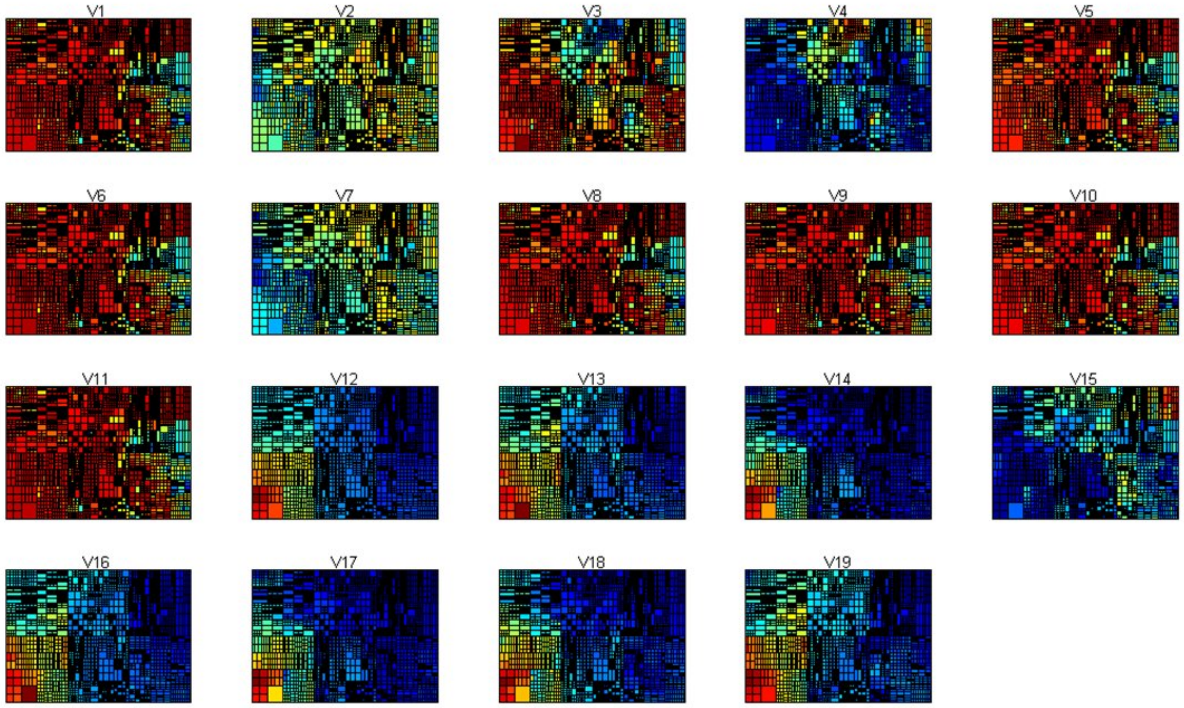


Figure 4.7: *GHSOM component planes. Each component represent one of the 19 climatic variables described in Table 4.1.*

4.2 Fuzzy Growing Hierarchical Self-Organizing Networks (FGHSON) as a tool to explore geospatial data

In this section FGHSON is used to find analogous environmental zones in South America and Colombia. In the first example, we used a bi-dimensional dataset containing information of precipitation and air temperature in South America. The low-dimensionality of this dataset will allow us to graphically illustrate the characteristics of the FGHSON when dealing with geospatial data. The second dataset is more complex and contains bioclimatic variables based on environmental variables from Colombia.

4.2.1 Finding zones with analogous precipitation and air temperature in South America

This example is a simple but real world application that will help to conceptualize the features of the FGHSON when dealing with geospatial data. With this aim, we have selected a case study whose objective is to find analogous environmental zones in South America in the month of January. Here, we used only two environmental variables to facilitate the visualization of some concepts that are difficult to visualize when high dimensionality is involved.

The dataset

The original data used is made of 24 datasets in an ASCII grid format [128]. They contain: total monthly precipitation (12 datasets, one per month), and mean air temperature (12 datasets, also one per month). The observations are referenced to a geographic coordinate system in a resolution of 0.5 degrees per pixel. The precipitation is given in millimeters per year, and the temperature in degrees Celsius. Both variables were combined forming datasets by month containing the couple temperature-precipitation. As it was previously mentioned, only for the month of the January dataset was used.

FGHSON training and results

The parameters of the algorithm were set to τ_1 = 0.3 (the parameter τ_1 represent a fixed percentage which control the growing process of the network in each layer, indicating if more prototypes must be added), τ_2 = 0.03 (the parameter τ_2 is a percentage that regulates the granularity of the data representation in the depth growth process, indicating if more levels must be created) and φ = 0.3 (the parameter φ represents the minimal membership degree of a observation to a cluster to be selected for further expansion of the network). After training, a structure of three levels and 28 prototypes was obtained.

FGHSON results in feature space and geographical space

As it was aforementioned in this chapter, when we cluster geospatial datasets we have to take into account the geographical and the feature spaces. The geographic space refers to the cartographic representation of a geographical area, and the feature space includes the n-dimensional space created for the variables associated to the georeferenced points (e.g. temperature and precipitation). In this example the feature space is bi-dimensional and it is made of air temperature and precipitation (as shown in Figure 4.8).

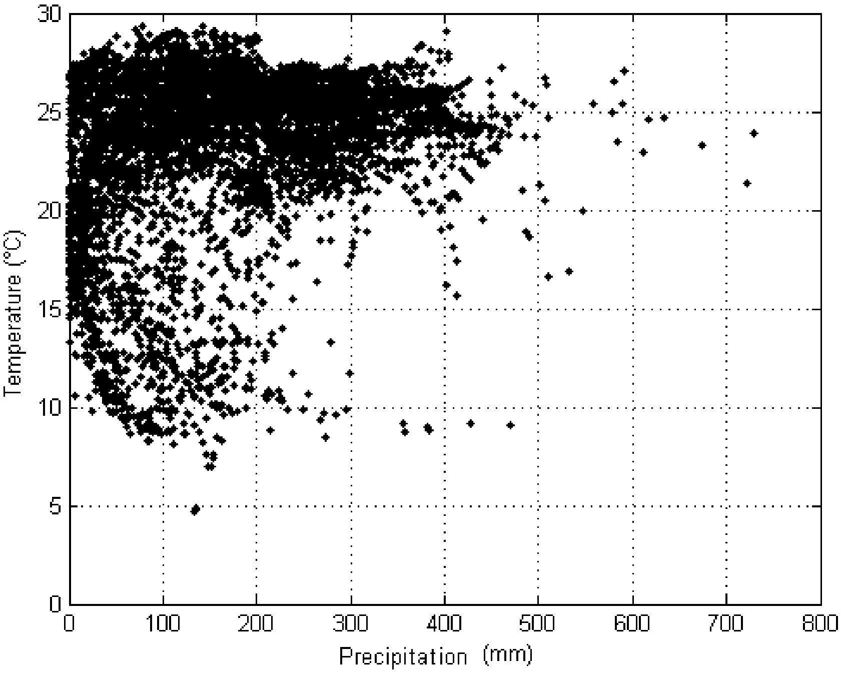


Figure 4.8: *Dataset used to train the FGHSON algorithm*

The reason to use the FGHSON algorithm is to obtain a cluster structure that allows to find zones with similar air temperature and precipitation. A reduced representation of the hierarchical structure created by the FGHSON in the feature space is shown in Figure 4.9.

In the feature space hierarchy, each scatter plot shows the membership of the observations to a single prototype. The observations with a higher membership to the given prototype are colored in light red, observations with a low membership in dark red, and observations with zero membership in black.

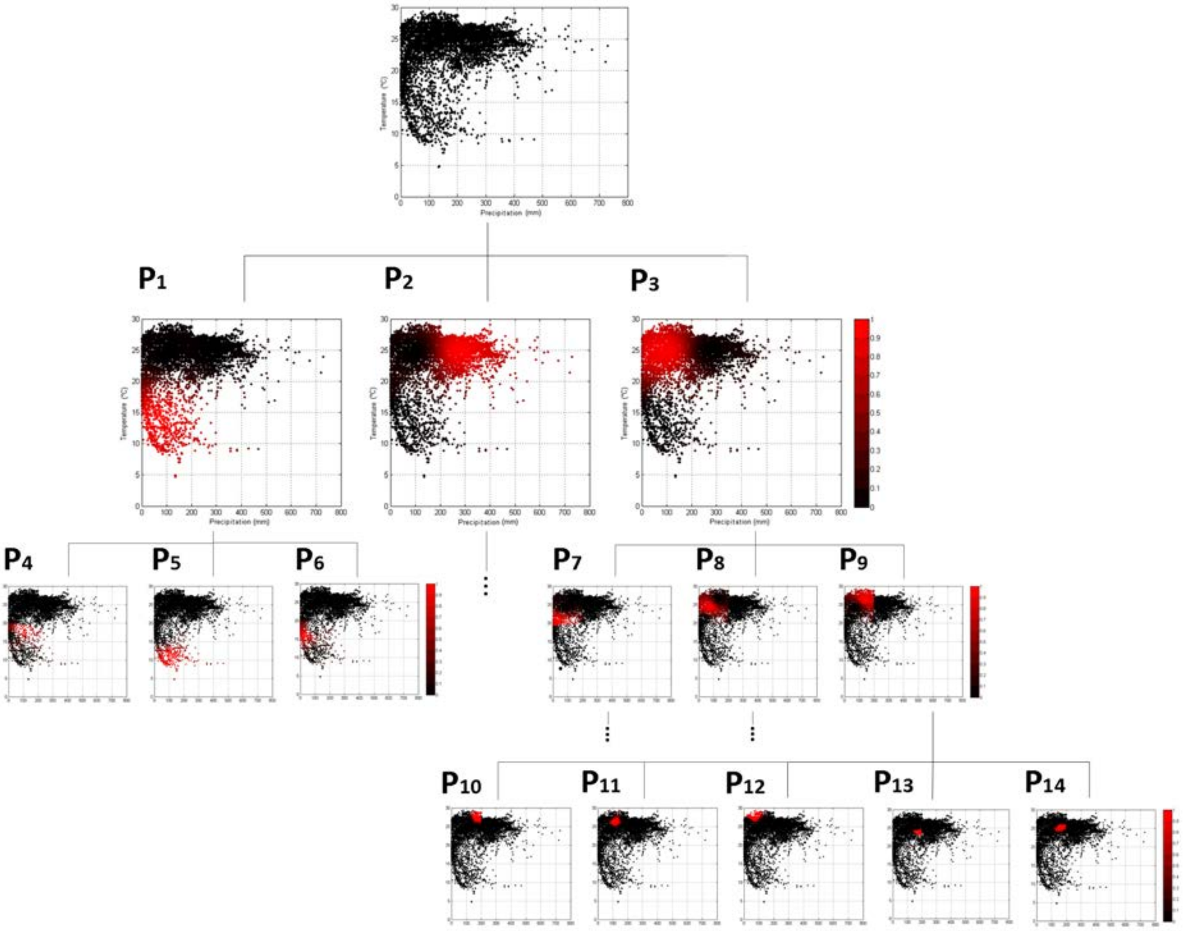


Figure 4.9: Hierarchical structure created by the FGHSON, representing the feature space of the dataset under study. Each scatter plot shows the membership of the observations to a single prototype created at each level. The scatter plot in the top level illustrates the dataset used. The x-axis represents temperature, and the y-axis precipitation. Some branches of the hierarchy were eliminated in order to facilitate the visualization.

In Figure 4.10 the structure created by the FGHSON is shown in a geographical space. For each prototype a map is displayed where the membership of the observations to this prototype are colored. Thus, geographical zones with a higher membership to the given prototype are colored in red, and observations with zero membership in dark blue.

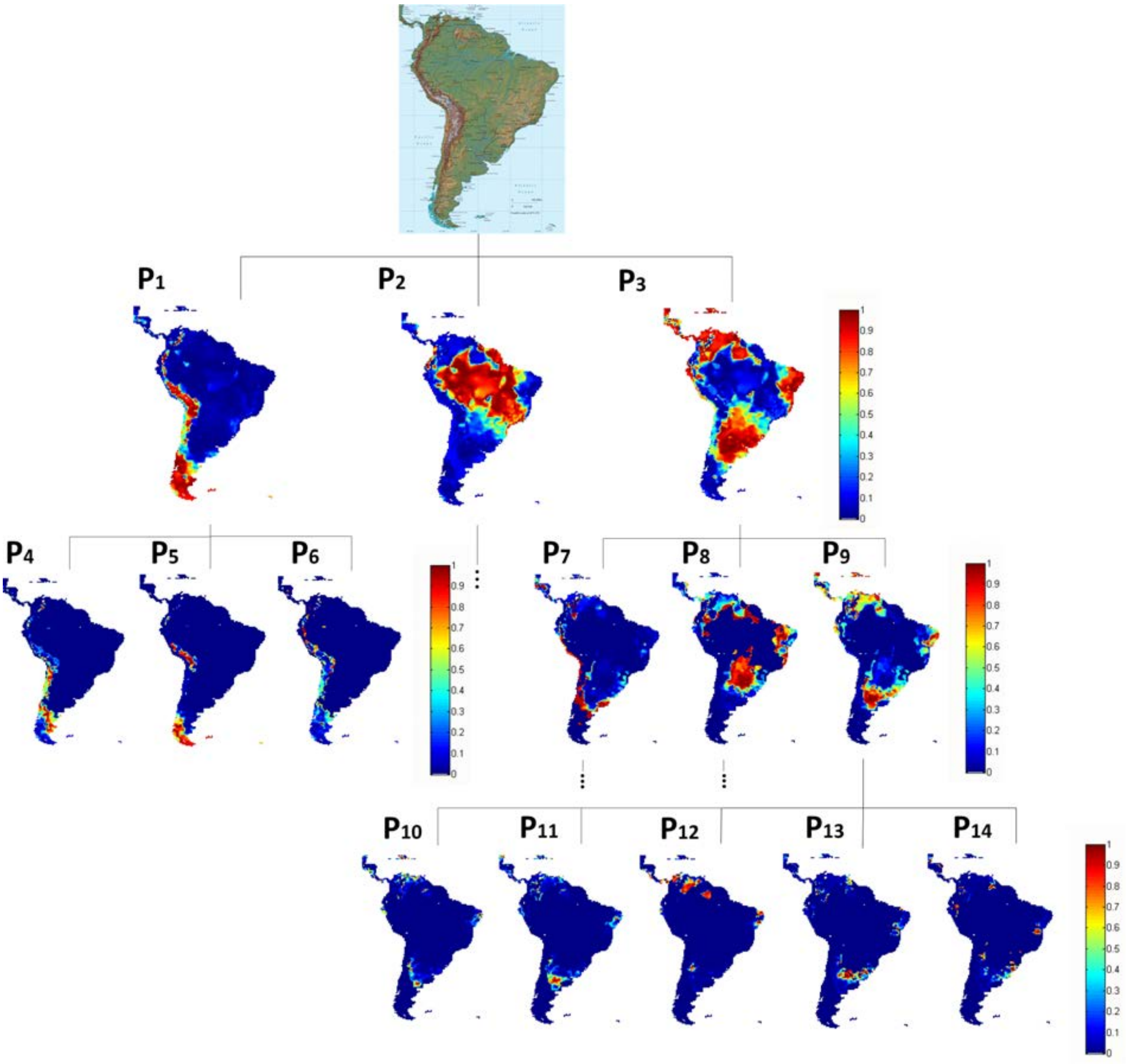


Figure 4.10: *Geographic space hierarchy. For each prototype Pi the membership of the observations to this prototype is given using a color code (dark red correspond to a membership of 1 while dark blue to a membership of 0).*

Analyzing the hierarchical structure

As can be observed in Figures 4.9 and 4.10 the first level of the structure is composed of three prototypes, namely P_1, P_2, P_3. At this level, the feature space is covered by three prototypes with the following characteristics:

- The first prototype, P_1, represent observations with precipitation levels between 0 mm and 500 mm and air temperature values between 0 °C and 21 °C;

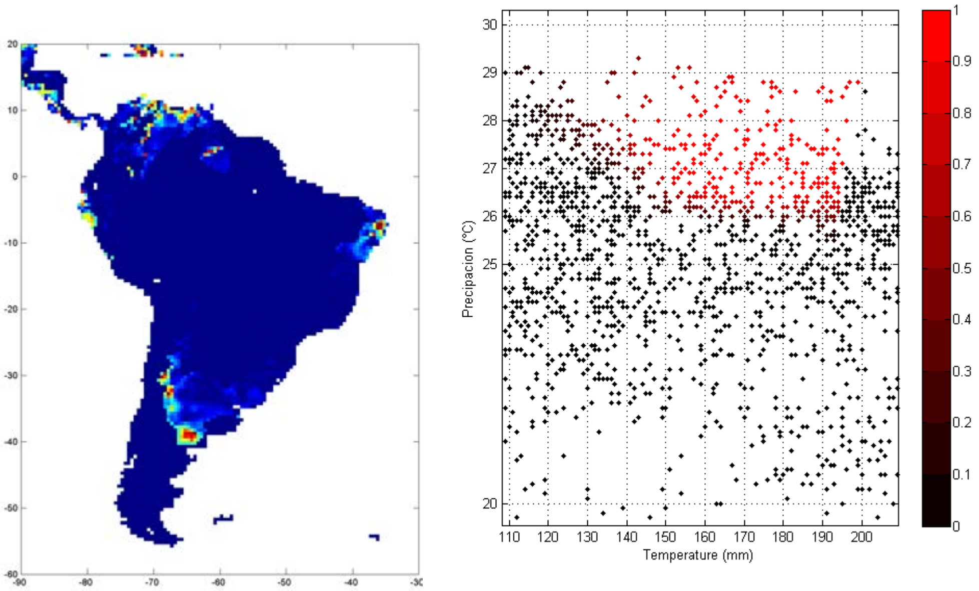
- The second prototype, P_2, includes observations with precipitation levels between 200 mm and 800 mm, and air temperature between 15 °C and 30 °C;

- Finally, the third prototype represents observations with precipitation levels between 0 mm and 200 mm, and air temperature values between 20 °C and 30 °C.

As an example, one can observe in the geographical space (Figure 4.10) that the Amazonian region is represented by P_2 where precipitation is represented between 200 mm and 800 mm, and air temperature between 15 °C and 30 °C, presenting the higher values of temperature and precipitation. The roughly division of the Amazonian region can be improved by selecting a next level in the hierarchy. It is out of the scope of this example to analyze all the prototypes. Usually the interesting prototypes in the structure are selected based on the user needs.

As an illustration, consider that we are looking for localities with mean monthly precipitation between 130 mm and 200 mm, and air temperature between 26 °C and 29 °C. The prototype that correspond to the characteristics indicated using the structure that was previously created for the FGHSON.

In this context, when selecting prototypes it is important to consider the coverage range, and to move in trough the hierarchy to the appropriate level which can represent the right range. Thus, the prototype that fulfills those parameters is P_{10}. Figure 4.11 shows P_{10} in the feature and the geographical space. Figure 4.11 shows the geographical zones with those characteristics together with their memberships.



(a) P_{10} in the geographical space

(b) P_{10} in the feature space

Figure 4.11: *Prototype selected which corresponds to the following characteristics: mean monthly precipitation between 130 mm and 200 mm, and air temperature between 26* °C *and 29* °C

4.2.2 Case study: finding analogous environmental regions in Colombia

In this case study we used the dataset described at the beginning of this chapter. A FGHSON algorithm was trained using as setup parameters τ_1 = 0.03, τ_2 = 0.003 and φ = 0.5. At the end of the training, was obtained a hierarchical structure of four levels and 22 prototypes. A reduced structure in the feature space is shown in Figure 4.12. In those maps, one can observe the fuzzy representation of the membership of the zones to the prototypes.

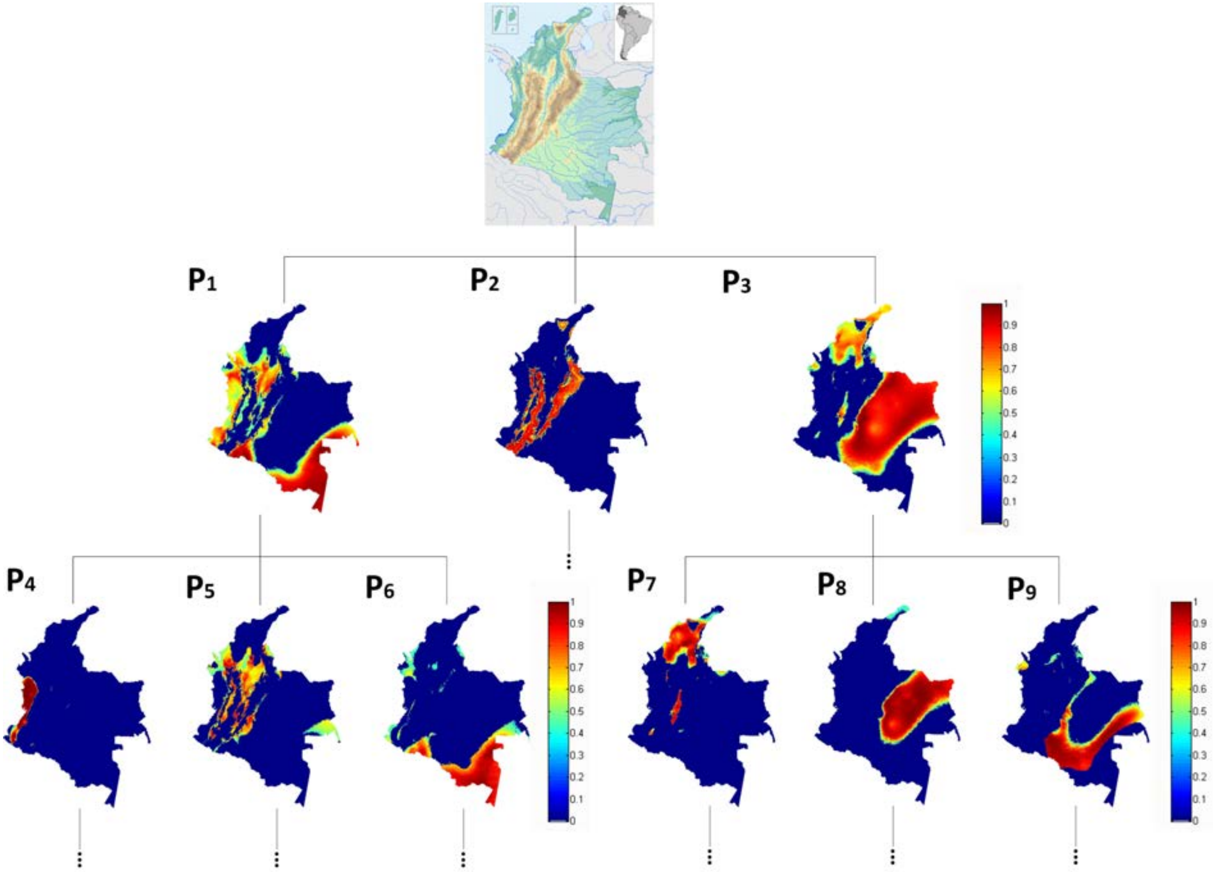


Figure 4.12: Hierarchical structure in the geographical space

4.2.3 Conclusions

In many fields including agroecology, environmental sciences and geosciences, large volumes of geospatial data must be analyzed in order to extract useful information embedded in those datasets. A first step in the analysis of geospatial data is the exploratory data analysis. This allows for instance: to maximize the insight into a data set, uncover underlying structure, extract important variables, to find relationships among variables and detect outliers.

In this section, three algorithms, namely the SOM, GHSOM, and FGHSON, were used for the exploratory data analysis of geospatial datasets. The results obtained allowed to compare their performances facing the challenges imposed when mining geospatial datasets.

As conclusions we can mention the following:

The three algorithms can deal with a large quantity of data. They were tested with 1,336,025 georeferenced points with 19 bioclimatic variables. The algorithms allowed to create prototypes of the observations helping to improve representation, interpretation of the geospatial dataset used as input.

The SOM and the GHSOM showed to be useful in the visualization of clusters in the feature space, since they allowed to project the 19 dimensions of the dataset used as input into two dimensions. Thus, allowing the visual exploratory analysis of relationships on the data.

Due to the ability of GHSOM and the FGHSON algorithms to create hierarchical structures they showed to be useful in the representation of geospatial data at several resolutions. It allowed to visualize regions with similar characteristics at several scales in both the geographical space and the feature space.

The FGHSON algorithm proved to be useful to handle the fuzzy boundaries of the geospatial clusters. It is true for environmental variables (e.g., temperature and precipitation) because boundaries are changing and they are rarely sharp or crisp in nature.

Finally, I present bellow a summary of the advantages and disadvantages of the SOM, GSOM, and FGHSON algorithms as tools for the exploratory data analysis of geospatial datasets.

Advantages of the SOM

1. SOM is becoming a well-known technique dealing with geospatial data [58, 76, 119, 110]. Hence, results in similar research fields can be compared. In addition, literature about methodologies dealing with SOM drawbacks is published continuously;

2. SOM allow for the projection of a high-dimensional space into a low dimensional space. Hence, improving the visual exploratory analysis of patterns in data;

3. There are several popular software used in geosciencies where SOM is implemented as a clustering and visualization technique. [58, 5, 6, 7].

Disadvantages of the SOM

1. A SOM does not create a hierarchical cluster structure;

2. The membership of the observations with respect to the clusters is crisp;

3. A SOM has a fixed network architecture, in terms of the number and the way of interconnecting the neural processing elements, which has to be defined before training. Obviously, in case of largely unknown input data characteristics, as is usually the case with geospatial data, it remains far from trivial to determine the network architecture that allows to obtain satisfactory results;

4. Supplementary techniques have to be used to cluster the SOM prototypes.

Advantages of the GHSOM

1. No additional techniques are needed to group the prototypes in the map;

2. The GHSOM allows for the projection of high-dimensional space into a two dimensional space;

3. Compared with the SOM, the overall training time is largely reduced since only the necessary number of units is developed

4. The GHSOM uncovers the hierarchical structure of the data by its very architecture, thus allowing the user to understand and analyze large amounts of data in an exploratory way to organize the data with a certain degree of detail.

5. The emerging maps at each layer of the hierarchy are rather small, it is much easier for the user to keep an overview of the various clusters.

6. Last but not least, by ensuring a consistent global orientation of the individual maps in the respective layers, the topological similarities of neighboring maps are preserved. Thus, navigation across map boundaries is facilitated, allowing the exploration of similar clusters that are represented by neighboring branches in the GHSOM structure.

Disadvantages of the GHSOM

1. The memberships of the observations of the clusters are crisp;

2. There is not available software to enable using GHSOM in geosciences [58].

3. The maps on individual layers cannot grow irregularly in shape and they cannot remove connections between neighboring units.

Advantages of the FGHSON

1. FGHSON does not require a-priori setup of the number of clusters. This feature is critical when dealing with geospatial data, because usually it is not possible to estimate a-priori the optimal number of clusters that can best represent a dataset;

2. The membership of the observations of the clusters is fuzzy;

3. The final structure does not necessarily lead to a balanced hierarchy (i.e. a hierarchy with equal depth in each branch). Therefore, areas in the input space that require more units for appropriate data representation create deeper branches than others. It is important when dealing with geographical-based data, because in many cases, there are regions that must be better represented;

4. The algorithm executes a self-organizing process that can be parallelized. Hence, when dealing whith large datasets, the tasks can be divided, distributing computational cost. This is an actual research issue in spatio-temporal data mining [99];

5. A software using the FGHSON algorithm in geosciences is in development;

Disadvantages of the FGHSON algorithm

1. The FGHSON model does not allow to project a high-dimensional space into a two dimensional one;

2. The FGHSON algorithm is a new algorithm. So, there is a need for further analysis;

Chapter 5

Finding similar patterns through time in spatio-temporal geospatial data

> "Time and Space... It is not nature which imposes them upon us, it is we who impose them upon nature because we find them convenient."
>
> Henri Poincaré

To find similar patterns is one of the most common tasks when mining spatio-temporal geospatial data. It is usually applied to find zones with similar environmental characteristics. Those results are used in several applications, for instance:

- **Agriculture:** knowing which zones share similar characteristics allows to enhance agricultural decision-making and management. For instance, to migrate successful crops to other zones with similar characteristics [3, 129].
- **Ecological triage:** finding similar regions to those were protected species live allows, for example, to relocate them to new regions or to expand protected areas.
- **Species migration:** several animal species migrate looking for places with similar environmental conditions to those where they live. Hence, finding zones with similar environmental characteristics allows the discovering of migration patterns. This is an actual research topic since global warming and devastation of the migratory routes is forcing many animals to seek for alternative habitats.
- **Plant hardiness zones:** they are defined as zones with specific climatic conditions in which a specific plant is capable of growing.

Techniques to cluster multivariate time series are used to group zones with similar variables' patterns. Frequently, the same techniques used to cluster discrete objects are used to cluster mutivariate time series. However, they share the same weaknesses discussed in chapter 4, when dealing for instance with: fuzzy boundaries in geospatial clusters, fuzzy hierarchical representation, and data with unknown distribution.

In order to better deal with spatio-temporal geospatial data, we introduced methodologies based on Self-Organizing Maps (SOMs) and the Fuzzy Growing Hierarchical Self-Organizing Networks (FGHSON) in order to find similar patterns through time in multivariate time series. We also present several examples of application in agroecology.

This chapter is organized as follows: section 5.2 introduces notation and definitions used throughout the chapter. Section 5.3 presents the problem of finding agroecozones through time. Here, Self-Organizing Maps are applied by using static and dynamic approaches to group similar agroecological patterns. Section 5.4 concerns the use of FGHSON for clustering spatio-temporal data in order to discover similar patterns shifted in time.

5.1 Related work

In diverse fields including agriculture, ecology and environmental sciences, clustering techniques are employed to find geographic areas with similar environmental conditions. These areas (ecoregions) are used to analyze patterns across similar environmental factors. This information is used for management, legislation, or ecological triage. When ecoregions are used for agricultural purposes they are called agro-ecozones. The challenges finding agroecoregions and agro-ecozones are similar, so techniques developed to find ecoregions are also used to find agro-ecozones. The difference between delineating agro-ecozones and ecoregions lies in that for delineating agro-ecozones, we must include variables related to crop production, like soil, landscape, crop management, and productivity.

To our knowledge, clustering techniques considering a "dynamic approach" to delineate ecoregions or/and agro-ecozones had been only reported by Hoffman and Hargrove [64, 129]. They developed a spatio-temporal clustering technique based on K-means ,including time as an additional spatial dimension, in order to delineate ecoregions. However, K-means presents several disadvantages. For instance, the number of clusters has to be defined prior to training, and usually, in geographic-based applications it is impossible to know that in advance. In addition, another limitation of this technique is the limited capabilities to represent hierarchical relations on the dataset.

Therefore, there is a clear need for techniques capable of dealing with large spatio-temporal datasets, that can automatically define the number of clusters while creating hierarchical structures to represent the data, and providing a fuzzy membership of the regions to ecoregions or agro-ecozones.

5.2 Notation and definitions

In order to frame the methodologies we used in the proper context, we begin with a review of the necessary background material.

- **Time series:** A time series $T = t_1, ..., t_m$ is an ordered set of m real-valued variables.
- **Subsequence:** Given a time series T of length m, a subsequence C_p of T is a sample of length $w < m$ of contiguous data from T, that is $C = t_p, ..., t_{p+w-1}$ for $1 \leq p \leq m - w + 1$. When the objective is to extract subsequences to be clustered we use a sliding window.
- **Sliding Windows:** Given a time series T of length m, and a user-defined subsequence of length w, a matrix S of all the possible subsequences can be built by "Sliding a window" across T and making the p^{th} row of S equal to subsequence "C_p". The size of the matrix S is $(m - w + 1)$ by w

Figure 5.1 illustrates all the above definitions and notations.

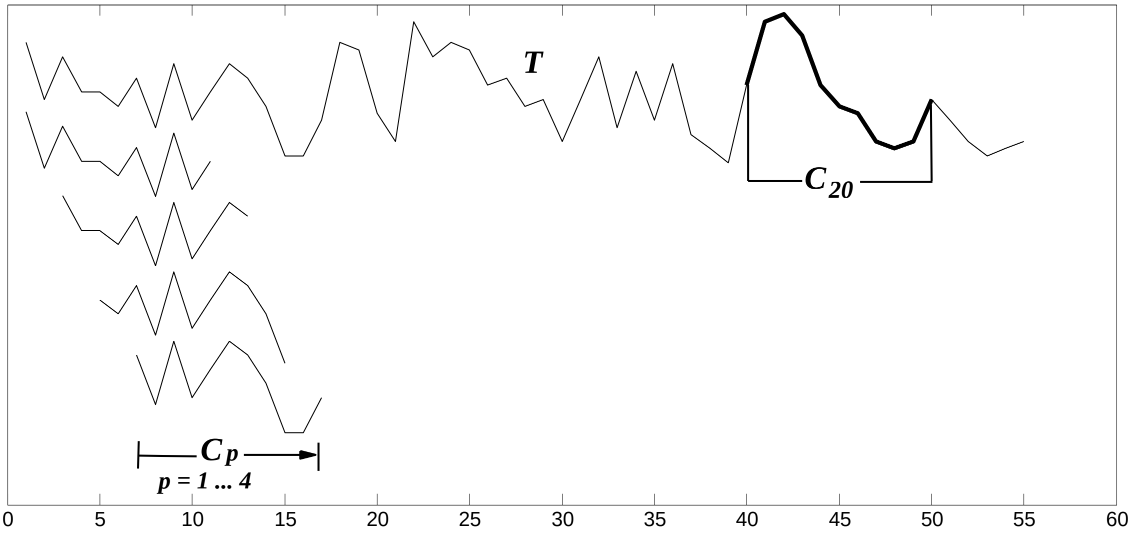


Figure 5.1: *An illustration of the notation introduced in this section: a time series T of length 55, the 20th subsequence of length $w = 10$, and the first 4 subsequences.*

Approaches for clustering time series can be broadly classified into two categories [73]:

- **Whole Clustering:** The notion of clustering here is similar to that of conventional clustering of discrete objects. Given a set of individual time series data, the objective is to group similar time series into the same cluster.
- **Subsequence Clustering:** Given a single time series, subsequences time series are extracted with a sliding window. Clustering is then performed on the extracted subsequence time series.

5.3 Finding agroecozones through time

Agroecozones are defined as geographic areas sharing similar characteristics for crop production. They allow to enhance agricultural decision-making and management. They allow, for instance:

- To migrating successful crops to zones with similar agroecological properties.
- Using crop management from regions with high productivity as examples for low productivity regions sharing similar conditions [3, 129].

Currently, agroecozones are found by using *whole clustering* techniques, which create static and rigid delineations [29, 78]. Nevertheless, environmental characteristics change over time, thus the agroecozones delineation ask for a dynamic process.

In this context, two approaches for delineating agroecozones can be distinguished :

- **Static delineation:** It uses a *whole clustering* approach to find similar zones. It considers a global vision of the time series. Hence, it compares whole time series ignoring information embedded in a temporal context, for instance variables' transitions and dynamics.
- **Dynamic delineation:** It takes into account the temporal context by using a *subsequence clustering* approach. In this case time series are split by using time windows in order to capture more detailed temporal similarities and variations.

In this section we present a case study highlighting the advantages in exploratory data analysis when using a dynamic delineation approach compared to the static one (e.g. to help discovering relationships and patterns embedded in a temporal context as similar patterns through time, relationships among variables and cluster dynamics). In addition, we present the advantages of using SOMs as a tool for exploratory data analysis when clustering multivariate time series.

5.3.1 Delineation of sugar cane agroecozones

In this case study, we used climate time series from the database of CENICAÑA (Colombian Sugar Cane Research Center [1]) to find similar agroecozones by using SOMs. Here, we present the results when *static* and *dynamic* approaches are used to define agroecozones.

5.3.1.1 Previous work

CENICAÑA has realized a detailed work characterizing agroecozones of the areas cultivated with sugarcane in the Cauca River Valley [28] based on physico-chemical properties of soil and climate. This agro-regionalization is crucial to establish the production potential of crop-specific sites and also to determine its agronomic crop management. The CENICAÑA study is very detailed and includes digital mapping and distribution of soil groups. The CENICAÑA agroecozones are based on semi-detailed studies of soil and climatic information collected between 1970 and 2001 from several sources (e.g. automatic weather stations, satellite imagery) and processed by a pool of experts from diverse fields (e.g. statisticians, agronomists, geographers).

It is not in the scope of this case study to mimic the CENICAÑA agroecoregionalizacion process. Our approach focused on finding similar climate agroecozones using static and dynamic approaches. For this aim we take advantage of the SOM's features as a tool for exploratory data analysis. For instance, its capability to deal with incomplete datasets [103], high-dimensional data, noisy data, and is less sensible to outliers than other clustering methodologies [124]. This approach is practical in cases where we want to perform exploratory data analysis of large volumes of high-dimensional and incomplete datasets. For example, Jiménez et al. show how using publicly available meteorological data coupled with information recorded by small-scale producers can be used to develop models for production systems [69].

5.3.1.2 The dataset

The zones under study are located in Colombia in the departments of Valle del Cauca, Cauca and Risaralda (see Figure 5.2). The variables were measured and collected daily from 1999 to 2006 by using 20 Automatic Weather Stations (AWS), distributed along the region under study. Each AWS has a geographical coverage area identified with a unique index, as shown in Figure 5.2. The variables stored by the AWS are described in Table 5.1.

The dataset used to train the algorithms is composed of time series of the eight climate variables described in Table 5.1. For this case study weekly averages were used, so, we had data from 366 weeks per variable. Thus, the final dataset is composed of 20 vectors (one for each AWS) and 2928 columns (8 climate variables x 366 weeks). The variables were normalized using the so-called Z-normalization (normalization to zero mean and standard deviation one) [55]. It ensures that all elements of the input vector are transformed into the output vector whose mean is zero while the standard deviation (and variance) are in a range of one.

In the next sections this dataset will be used to find ageoecozones by using static and dynamic clustering approaches.

Table 5.1: *Variables used in a first approximation to define agroecozones.*

Variable	Acronym	Units	Missing values (%)
Maximum Temperature	T_{max}	°C	8.29
Minimum Temperature	T_{min}	°C	8.30
Average Temperature	T_{avg}	°C	8.32
Maximum Relative Humidity	RH_{max}	%	8.29
Minimum Relative Humidity	RH_{min}	%	8.29
Average Relative Humidity	RH_{avg}	%	8.29
Solar Radiation	$Rad.$	Cal/cm^2	8.50
Precipitation	$Prec.$	mm	0

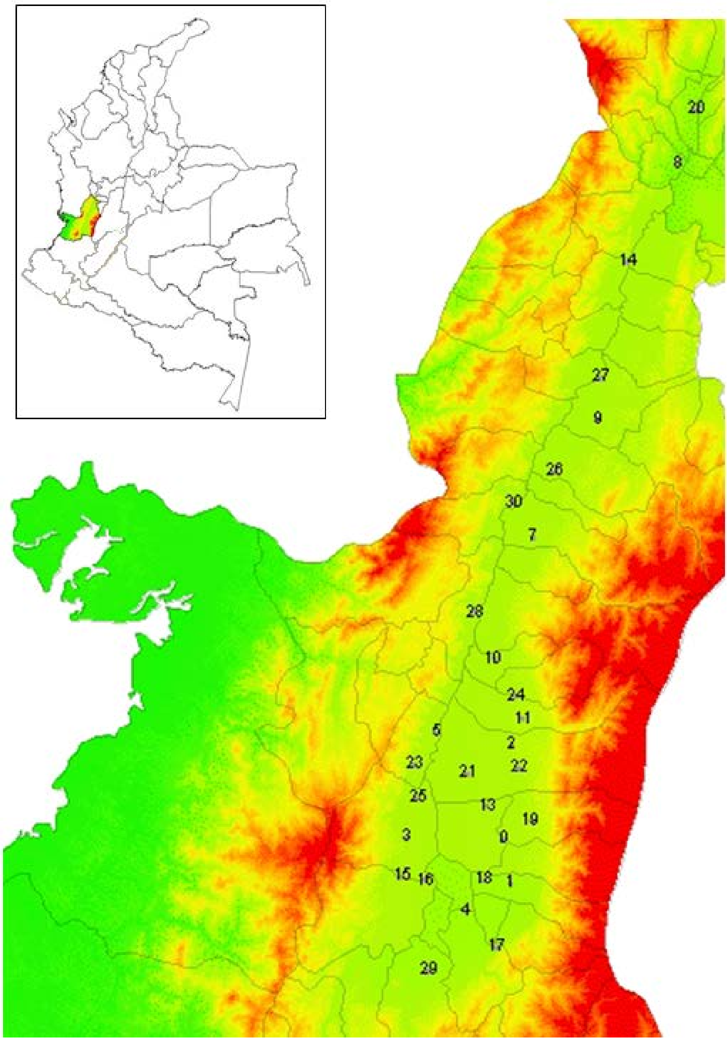


Figure 5.2: *Automatic weather stations (AWS) distributed along the Valley of the Cauca River. Each AWS is identified with a unique index.*

5.3.2 Static delineation approach

The objective of this experiment is to find agroecozones with similar climatic patterns using a *whole clustering* approach. For this aim we used the whole time series from 1999 to 2006 from each AWS. Thus, we will compare times series of seven years. An example of the multivariate time series for a specific AWS is presented in Figure 5.3.

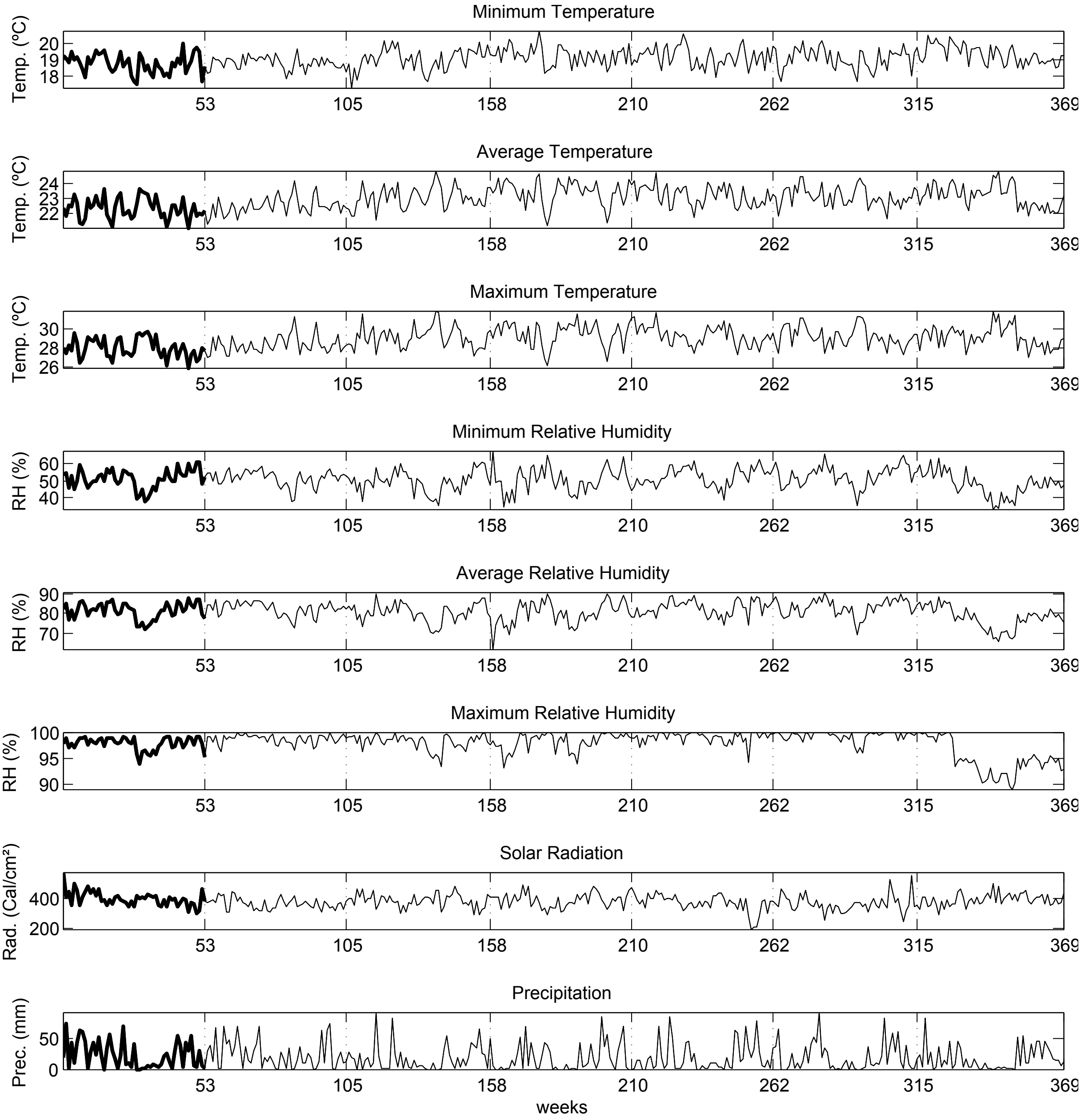


__Figure 5.3:__ Time series of the variables measured by the Automatic Weather Station labeled with the index 0. The time series contains weekly observations (non- normalized) of the eighth variables presented in Table 5.1 from 1999 to 2006. In the X-axis is shown the last week for each year. For instance the year 1999 (highlighted at the begin of the time series) ends in 53th week, the year 2000 in 105th week, and 2001 in 158th week. When using a static approach the entire time series are used; in the case of a dynamic approach the time series are split by years, obtaining in this case six subsequences per variable.

5.3.2.1 Training and results

A Self-Organizing Map was set with the parameters shown in Table 5.2 and trained with the dataset described in the previous section.

Table 5.2: *SOM parameters. The SOM training is divided into two phases. The first phase is the rough training. In this phase map units are adapted to obtain an approximate representation of the data. It utilizes a large neighborhood radius. The second phase is the fine-tunning phase that uses a small neighborhood radius for a more precise adjustment of the map to the data. In both of those phases the neighborhood radius decreases as the training progresses.*

Parameter	Value
Map size	32 x 32
Lattice	Hexagonal
Shape	Rectangular
Training Type	Batch
Neighborhood	Gaussian
Rough training radius initial	15 map units
Rough training radius final	5 map units
Rough training length	10 epochs
Fine-tuning training radius initial	5 map units
Fine-tuning training radius final	1 map units
Fine-tuning training length	30 epochs

After training, in order to visualize the groups obtained, the U-matrix was calculated and the labels corresponding to the code of each AWS were ploted, as it is shown in Figure 5.4.

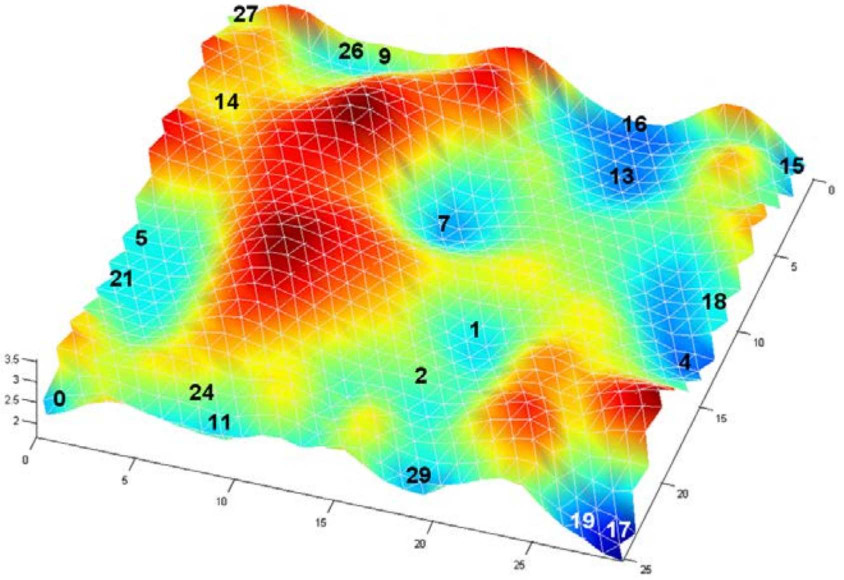


Figure 5.4: *The U-matrix represented in a 3D plot highlights the distances between prototype vectors. We can note the "valleys" and "hills" created by the U-matrix. Clusters can be seen as "valleys" (neurons colored with shades of blue, for instance vectors labeled with 16 and 13) constrained by "hills" (neurons colored with shades of red) that separate for instance the cluster formed by prototypes 16 and 13 from prototype 15.*

5.3.2.2 Analysis

Based on the results shown by the U-matrix, clusters of AWS where colored with the same hue (see Figure 5.5). As a result, zones with similar climate patters can be distinguished. Some areas despite being close in a geographical space are distant in the SOM (feature space), indicating that although they are close geographically they present different climate patterns. For instance zones 4 and 17 in Figure 5.5. Conversely, there are zones geographically distant, but close in the feature space because of their similarity in climate patterns, for example zones 17 and 19.

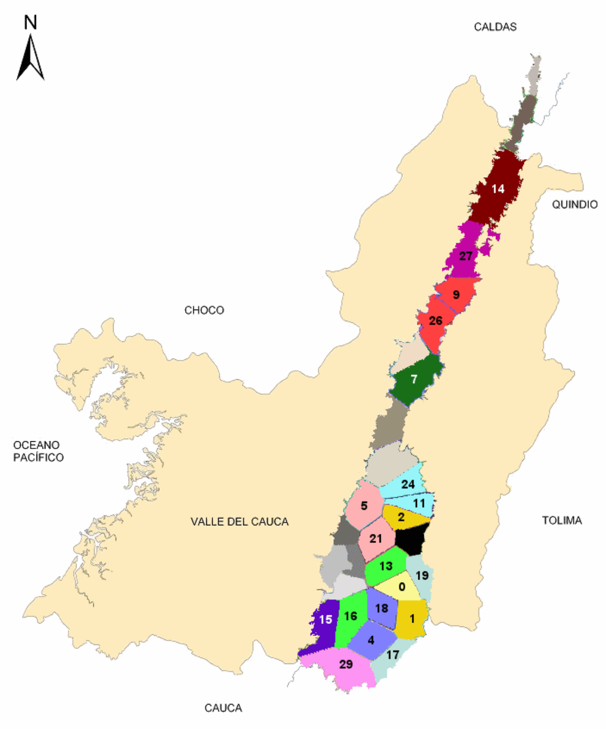


Figure 5.5: Agroecozones using a static delineation approach. Zones located in the same cluster in the SOM are colored with the same hue. Note that some areas close in geographical space are distant in the feature space, hence they present different climate patterns. On the other hand, there exist geographically distant zones that are close on feature space, for example zones 17 and 19 located at the bottom right corner of the SOM and colored in light blue in the geographical space.

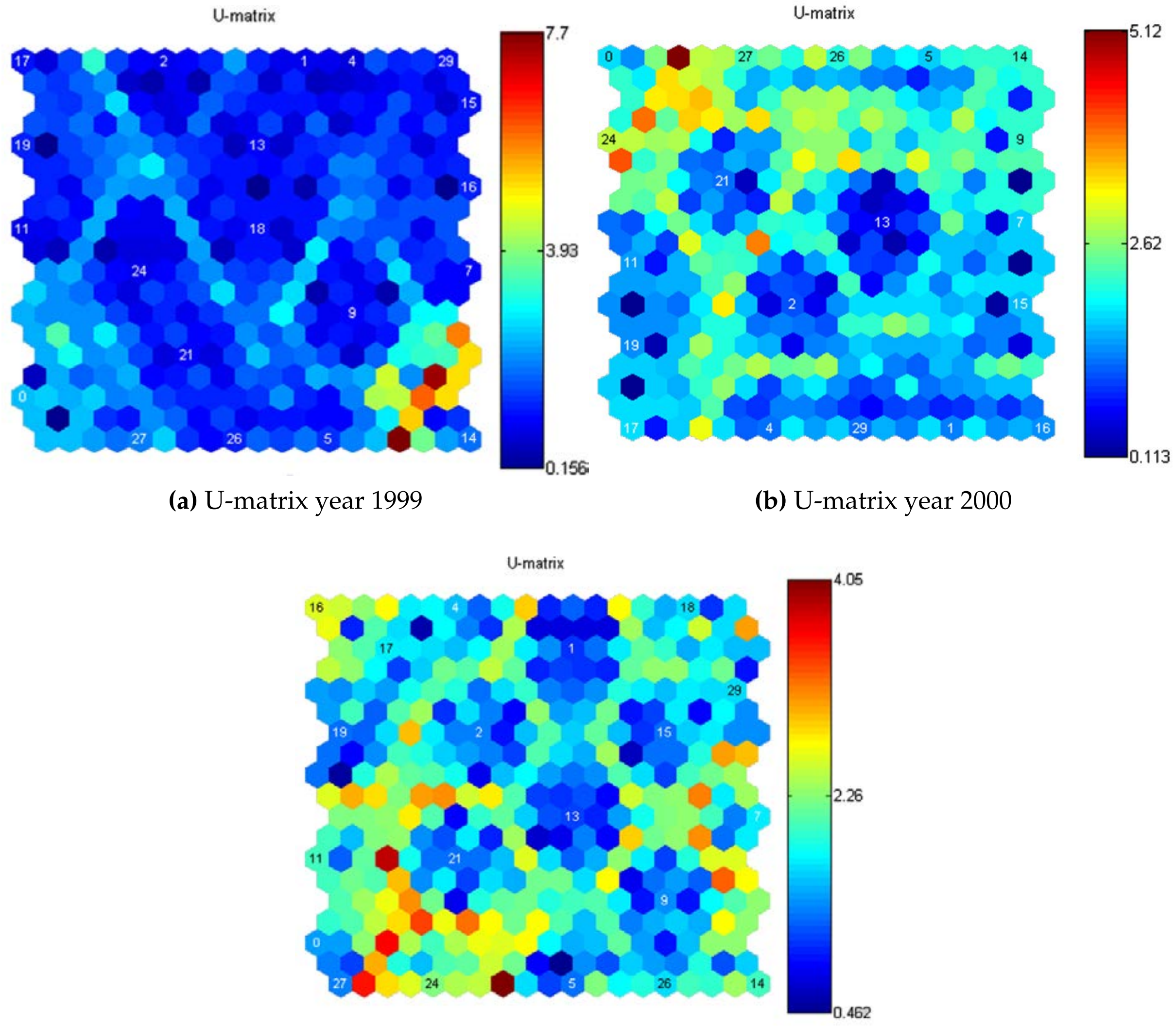
This approach groups regions with similar patterns during the whole period of seven years. Nevertheless, if subsequences of time series are used they will allow us to analyze similar patterns at several time resolutions. The following section presents a dynamic approach clustering in which time series are split by using time windows in order to capture more detailed temporal similarities, variations and dynamics in agroecozones.

5.3.3 Dynamic delineation approach

This approach is used to analyze the characteristics of the zones under study considering subsequences of time series using time windows of one year. The dynamic delineation approach allows to extract knowledge about the dynamics of the agroecozones. An example of a dynamic behavior is an agroecozone that changes from one group to another each year.

5.3.3.1 Training and results

In this experiment, the dataset described in section 5.3.1.2 from 1999 to 2006 was split into seven datasets, one for each year. Thus, seven SOMs where trained with each of the seven datasets. Each SOM was set with the parameters described in Table 5.2. The U-matrix obtained after training for the years 1999, 2000 and 2001 are shown in Figure 5.6.



(a) U-matrix year 1999

(b) U-matrix year 2000

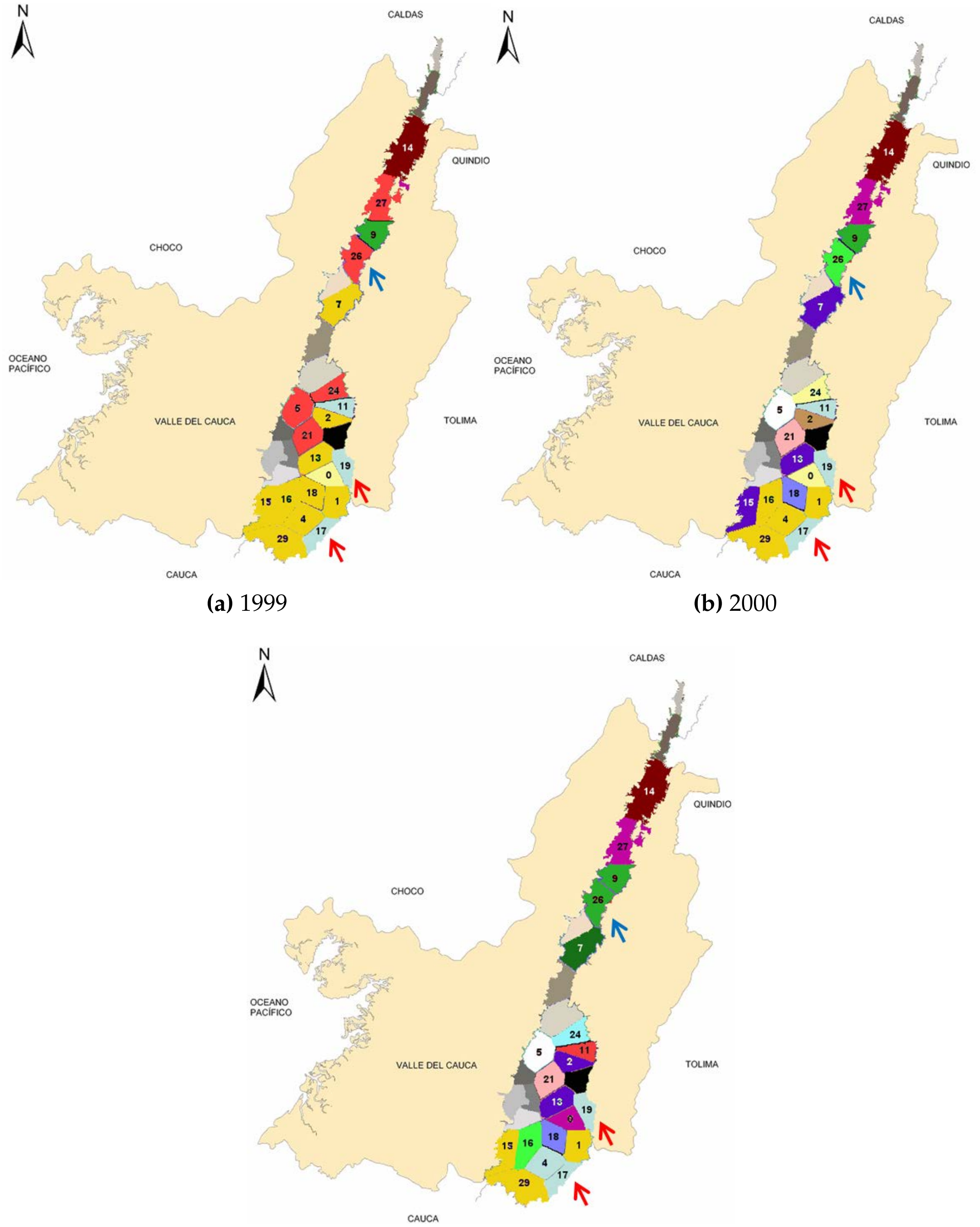
(c) U-matrix year 2001

Figure 5.6: *U-matrix from years 1999, 2000 and 2001. In this sequence of U-matrix we can follow the dynamics of the clusters year by year. As an example we will follow the AWS number 11, 17, and 19. If we compare the U-matrix we can observe that in 1999 and 2000 AWS 11, 17, and 19 still near in the map and with a low distance between them (this can be seen as a blue valleys). But in 2001 AWS 11 appears with a high distance from AWS 17 and 19 (it can seen as a orange hill).*

5.3.3.2 Analysis

Comparing the clusters obtained, we can observe that some groups do not change through time, while others change. As an example, one can observe the clusters obtained using the data from 1999, 2000 and 2001 in Figure 5.7. Group 1 (indicated with the blue arrow) gathers together different ecoregions every year, while group 2 (indicated with the red arrow) gathers the same members over the years.

This indicates that agro-ecoregionalization should not be static. Therefore, we propose a new approach that involves agroecological zoning in space-time. Thus, management crop decisions are less subject to uncertainty related to the agricultural suitability of a particular area, despite the occurrence of seasonal weather phenomena, as for instance “El Niño” or “la Niña”.



(a) 1999

(b) 2000

(c) 2001

Figure 5.7: *The dynamics of the variables affect the agroecozones delineations; for instance two regions can be part of the same agro-ecozone one year but the following year belong to different agroecozones (zones indicated with blue arrows). On the contrary, there are zones which maintain a similar behavior year after year as for instance the zones indicated with the red arrows.*

5.4 Using Spatio-temporal clustering to find similar patterns shifted in time

As it was shown in the previous section, the delineation of similar environmental regions can be done in a dynamical way. In this context we had shown that a *dynamic approach* using subsequence clustering allows to discover interesting dynamic patterns. The methodology presented in the previous section consisted on dividing the time series of seven years in subsequences of one year by using a shift of a year.

Taking into account that similar behaviors may occur at any time (not only each year), in this section we use smaller values for the shift in order to detect patterns translated in time.

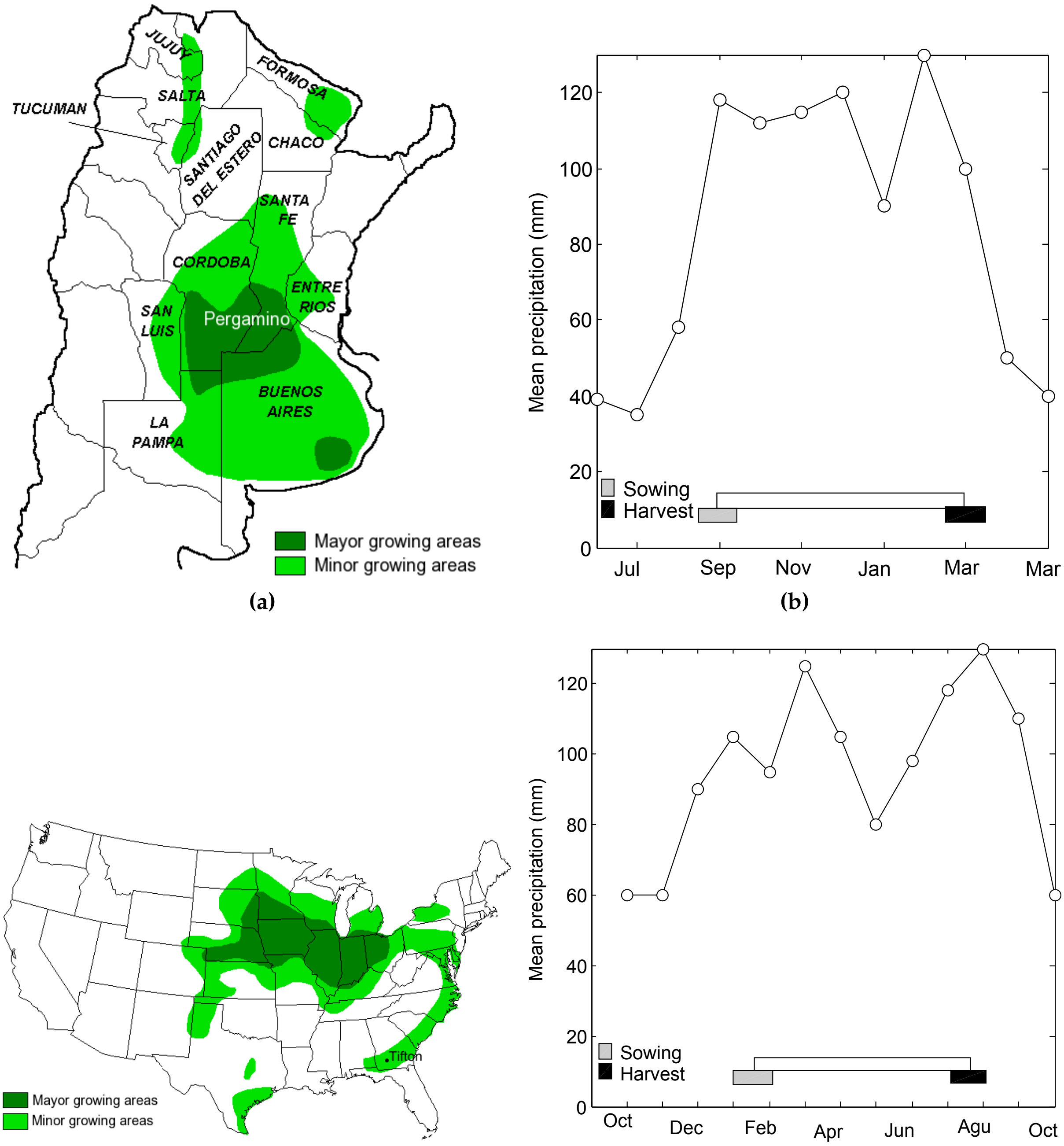
A well known example of zones with similar agroecologic characteristics shifted in time is presented in the maize culture. Maize (*Zea maize* L.) is grown in the Southeast of the USA and in the Pampas region of Argentina at different time of the year but under similar agroecological conditions [72], as shown in Figure 5.8.

To our knowledge, multivariate time series clustering techniques considering a *dynamic approach* to delineate zones with similar environmental patterns had only been reported by Hoffman and Hargrove [64, 129]. They developed a multivariate spatio-temporal clustering technique based on K-means, including time as an additional spatial dimension, in order to delineate ecoregions. Nonetheless, K-means presents several disadvantages:

- The number of clusters has to be defined a-priori, and usually, in geographic-based applications it is impossible to know this parameter in advance.

- K-means cannot represent the hierarchical fuzzy relationships of the data.

In this section, Fuzzy Growing Hierarchical Self-Organizing Networks are applied to find geographical areas with analogous environmental conditions shifted in time.



(a) (b)

(c) (d)

Figure 5.8: (a) Regions in the Argentina where maize is grown [2]. (b) Monthly mean precipitation for Pergamino, Argentina based on historical data from 1931 through 1996. The crop calendars at the bottom of the graph illustrate the growing seasons as well as planting and harvesting periods [72]. (c) Regions in the United States where maize is grown [2]. (d) Monthly mean precipitation in Tifton, Georgia based on historical data from 1922 through 1998. The crop calendars at the bottom of the plot show the growing, planting, and harvesting periods. [72].

5.4.1 Using FGHSON to find similar ecoregions shifted in time

In this case study our objective is to find similar environmental zones trough time in South America. In this experience we are looking for regions with similar patterns in time windows of three months.

5.4.2 The dataset

The original data is made of 24 datasets in ASCII grid format [128]. They contain: total precipitation per month (12 datasets, one per month), and mean air temperature (12 datasets, also one per month) from 6477 georeferenced points in South America and part of Central America. The precipitation is given in millimeters per year, and the temperature in degrees Celsius. Observations are referenced to a geographic coordinate system in a resolution of 0.5 degrees per pixel.

In order to obtain multivariate time series of three months for the couple temperature-precipitation shifted by one month, we created a dataset as follows:

1. We split temperature and precipitation datasets into three months time series, as shown in Figure 5.9.

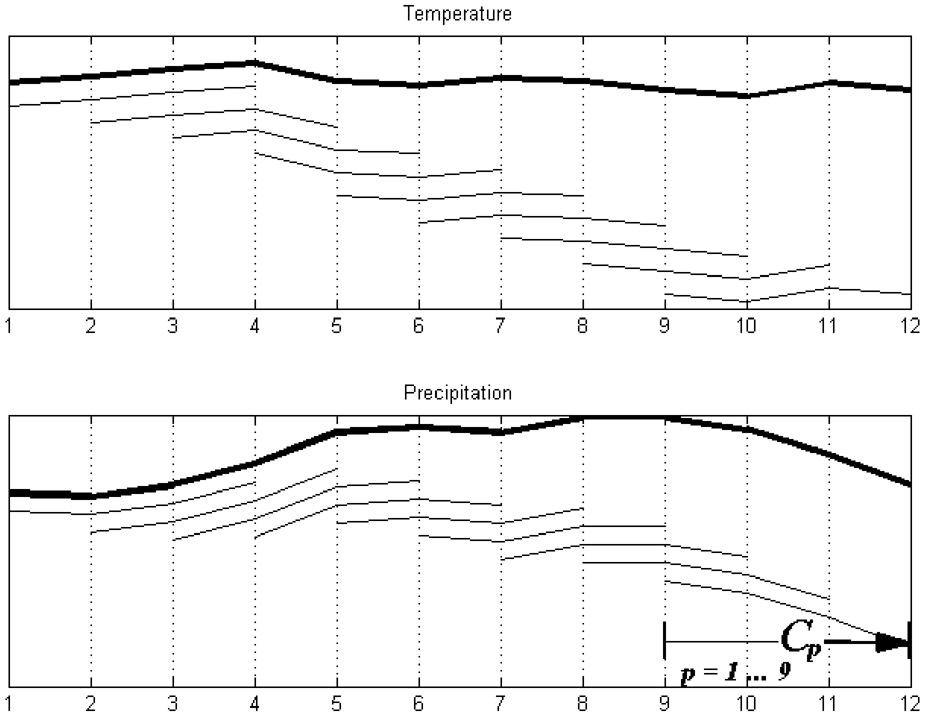


Figure 5.9: *Time series of temperature and precipitation with a length of twelve months, $w = 3$ months, and the first 9 subsequences of the three months data.*

2. Using a shift of one month we obtain twelve vectors of three months of temperature and precipitation for a georeferenced point, as illustrated in Figure 5.10.

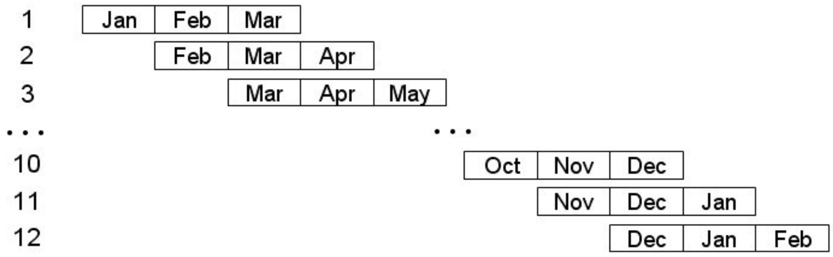


Figure 5.10: *Vectors corresponding to one point. For each georeferenced point we have twelve vectors. Taking into account that only data from twelve months is available, the vectors number twelve and eleven were created using the months of January and February from the the same dataset.*

3. The final dataset is made of six columns (temperature and precipitation per trimester) and 77724 rows (12 vectors multiplied by 6477 georeferenced points), as illustrated in Figure 5.11.

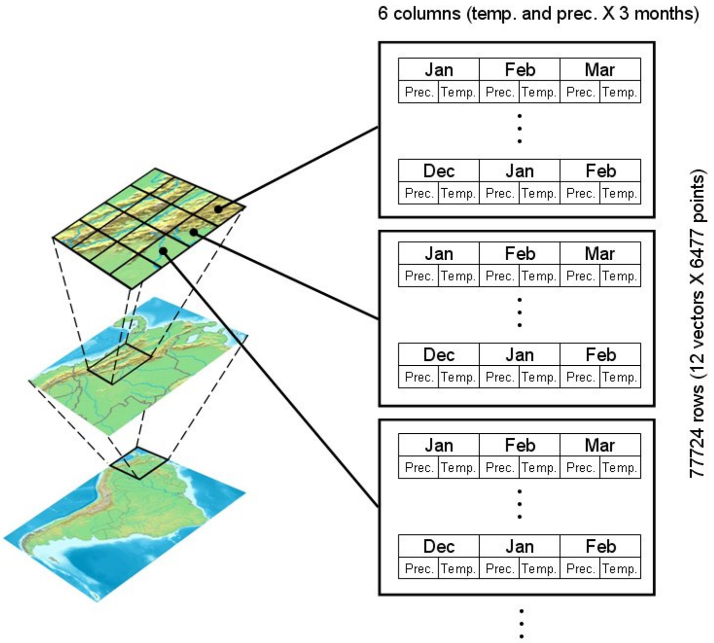

6 columns (temp. and prec. X 3 months)
Jan Feb Mar
Prec. Temp. Prec. Temp. Prec. Temp.
Dec Jan Feb
Prec. Temp. Prec. Temp. Prec. Temp.
77724 rows (12 vectors X 6477 points)

Figure 5.11: *Final dataset. This schema illustrates how the final dataset was created. It is made of 6 columns and 77724 rows.*

Thus, using a clustering technique we can obtain groups of zones with similar quarterly patterns trough time. For instance zones that in the period January-February-March present similar patterns to other zones in the period May-June-July.

The next section presents the results obtained after training a FGHSON with the dataset presented above.

5.4.3 Training and results

The dataset previously presented is used to train a FGHSON set with the following parameters: $\tau_1 = 0.3$ (the parameter τ_1 represent a fixed percentage of the quantization error which control the growing process of the network in each layer, indicating if more prototypes must be added), $\tau_2 = 0.08$ (the parameter τ_2 is a percentage of the quantization error that regulates the granularity of the data representation in the depth growth process, indicating if more levels must be created) and $\varphi = 0.1$ (the parameter φ represents the minimal membership degree of a sample to a cluster to be selected for further expansion of the network). After training, a tree structure of three levels was obtained (see Figure 5.12).

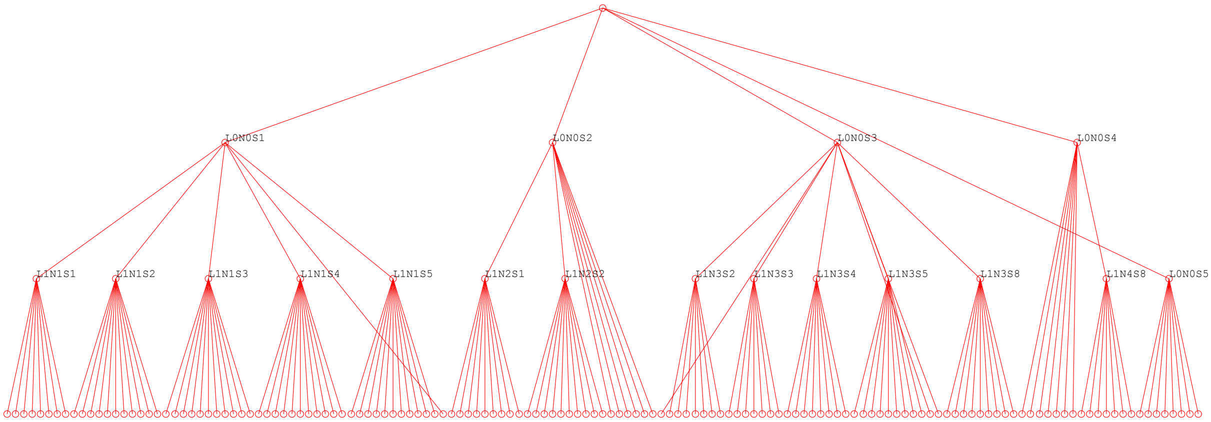


Figure 5.12: *Structure created by the FGHSON. A hierarchical structure of three levels was created. The first level is composed of five prototypes, namely: L0N0S1, L0N0S2, L0N0S3, L0N0S4 and L0N0S5. The nomenclature used to label the prototypes indicates that they are "sons" (S1, S2, S3, S4 and S5) of the so-called node "father" zero (N0) from the level zero (L0). In the same way at the second level we can observe the "sons" of L0N0S1 namely L1N1S1, L1N1S2, L1N1S3, L1N1S4, L1N1S5 and L1N1S6.*

The patterns represented by the five prototypes in the first level : L0N0S1, L0N0S2, L0N0S3, L0N0S4 and L0N0S5; are shown in Figure 5.13. Note that each combination of temperature-precipitation for each prototype presents a particular pattern.

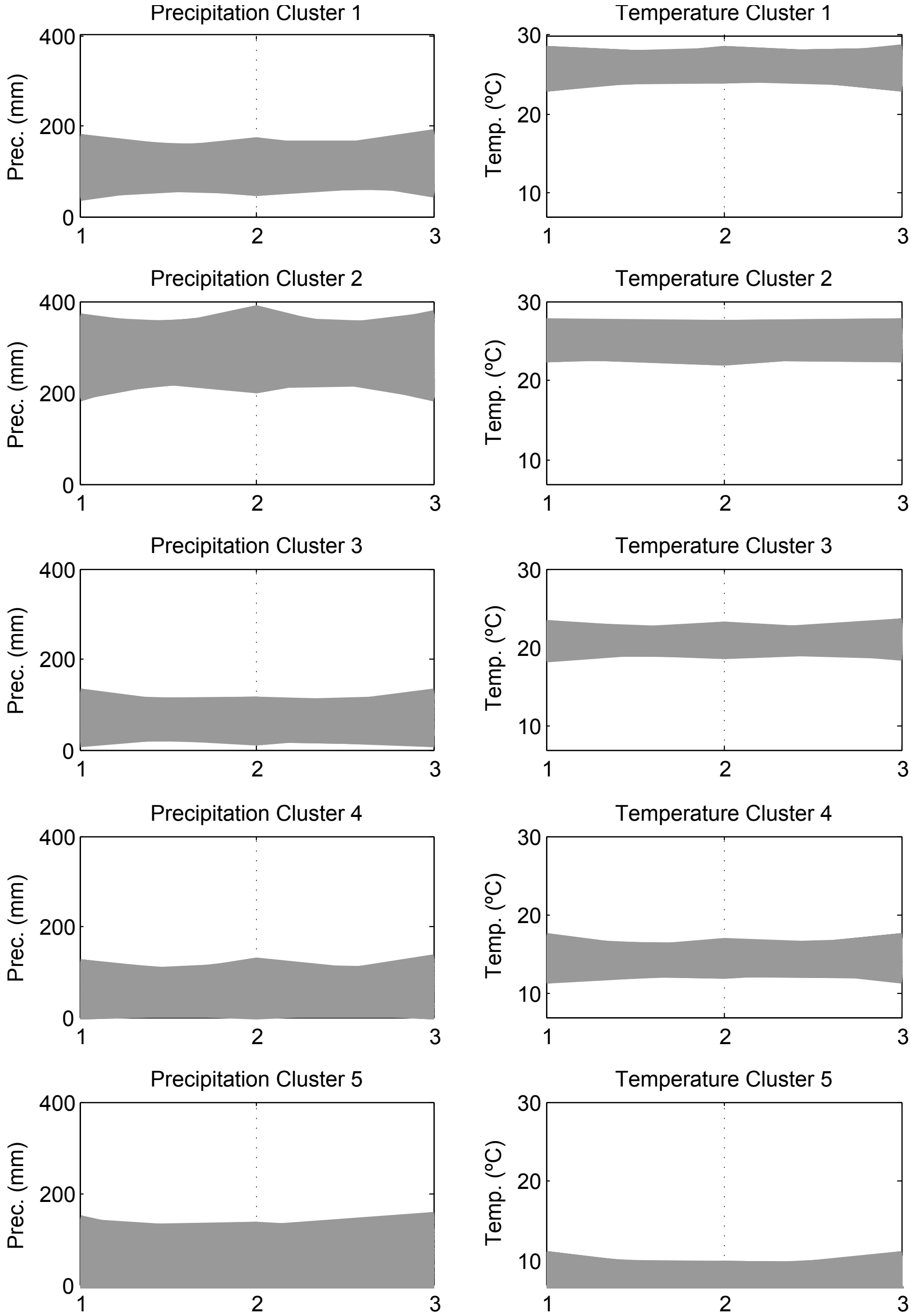


Figure 5.13: *Temperature and precipitation patterns of the prototypes from the first level. Only the vectors with a membership higher that 0.9 were plotted. Note that for each cluster a particular pattern temperature-precipitation was obtained.*

More specific patterns are found when we move into lower levels of the tree-structure, as illustrated in Figure 5.14.

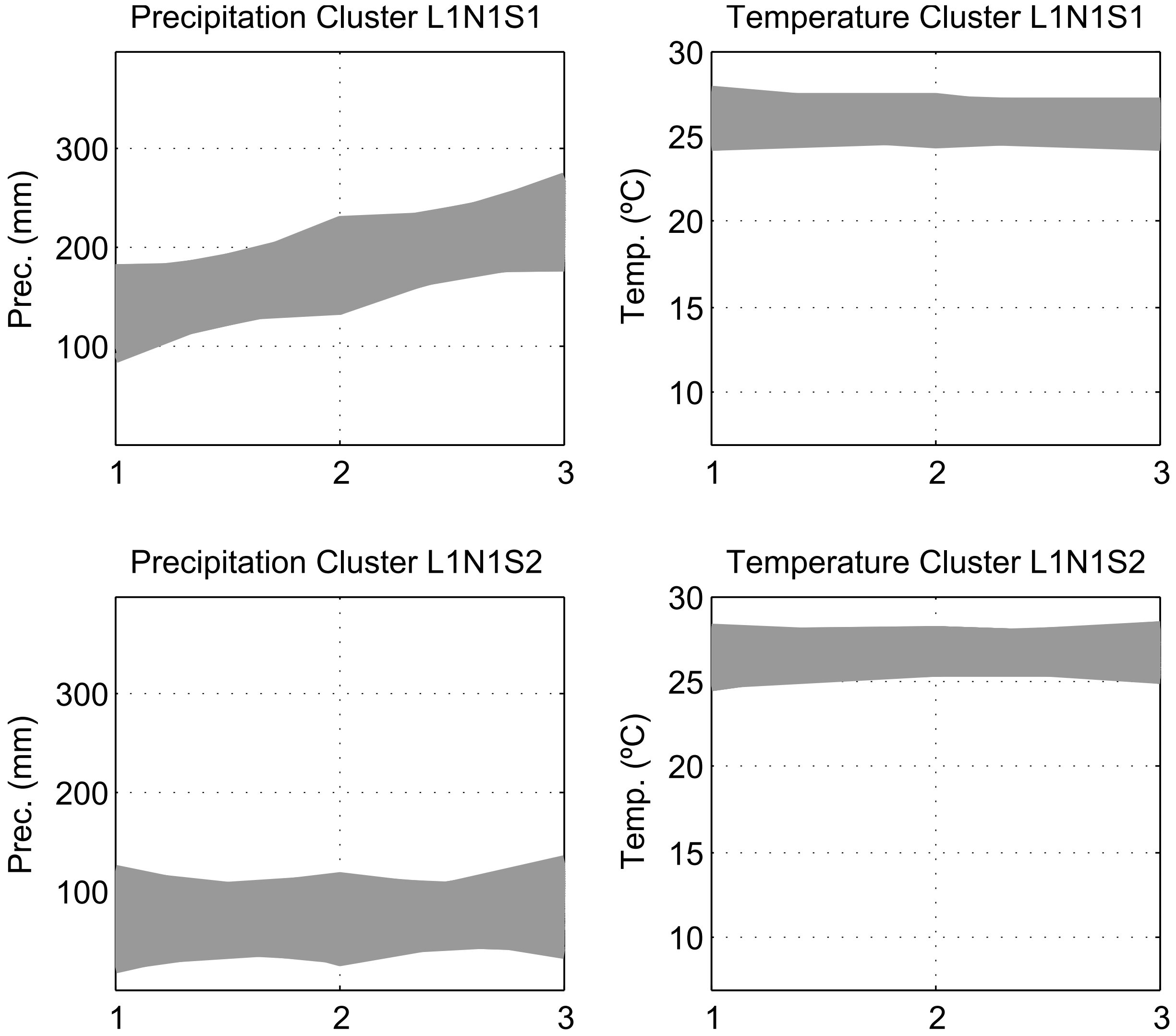


Figure 5.14: Two "sons" of the prototype L0N0S1 (prototypes L1N1S1 and L1N1S1). Only the vectors with a membership higher that 0.9 for each cluster are plotted.

5.4.4 Analysis

The FGHSON algorithm creates a hierarchical structure that allows to arrange zones with similar quarterly patterns. As an example, we selected the cluster L1N1S1. This cluster gathers together the vectors with similar trimestrial temperature and precipitation patterns, as shown in Figure 5.15. In this group of vectors we can find a wide range of values. Nevertheles, if we apply a filter taking only the vectors with a membership higher than 0.9 (represented in red and dark red in Figure 5.15) we obtain the ranges described in Table 5.3 and shown in Figure 5.14.

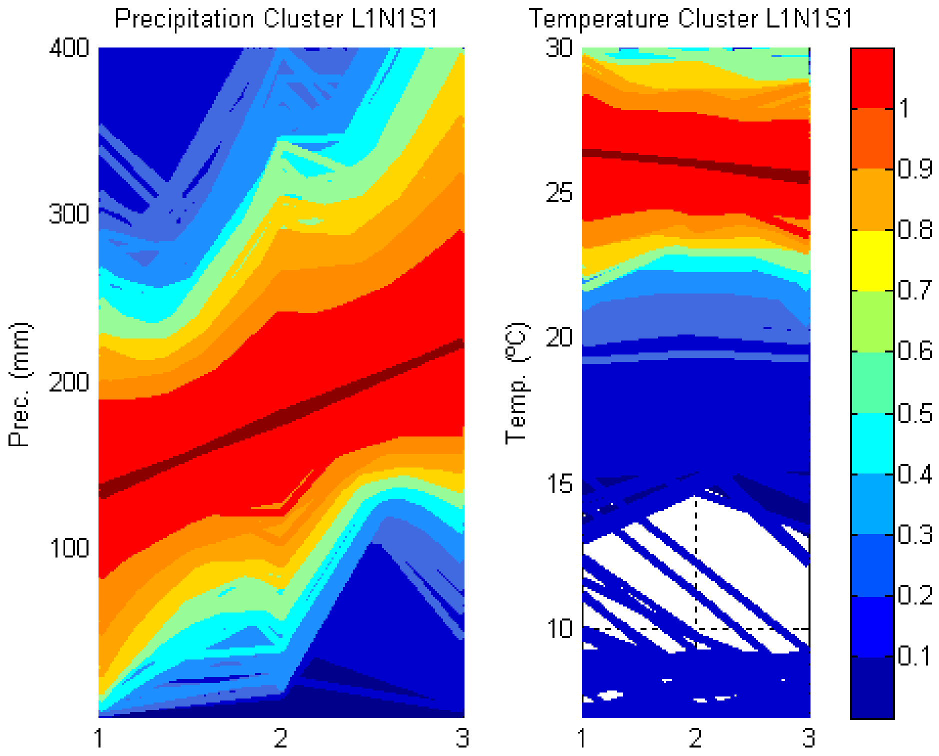


Figure 5.15: *Cluster L1N1S1 with the membership of its vectors.*

Table 5.3: *Ranges of temperature and precipitation for the cluster L1N1S1. The memberships higher that 0.9 for each cluster were selected*

	First month range	Second month range	Third month range
Precipitation(mm)	90-180	110-230	150-290
Temperature (°C)	24-29	24-28	23-27

We have plotted in Figure 5.16 the geographical zones with similar patterns from cluster L1N1S1. The color code used to represent the memberships (i.e. red for high membership and blue for low memberships) is the same as the one used in Figure 5.15. Each map represents similar zones for the each one of the twelve triplets. Hence, we can distinguish zones with analogous environmental conditions shifted on time. For instance, we can distinguish zones that in the period February-March-April present similar characteristics to other zones in the period October-November-December.

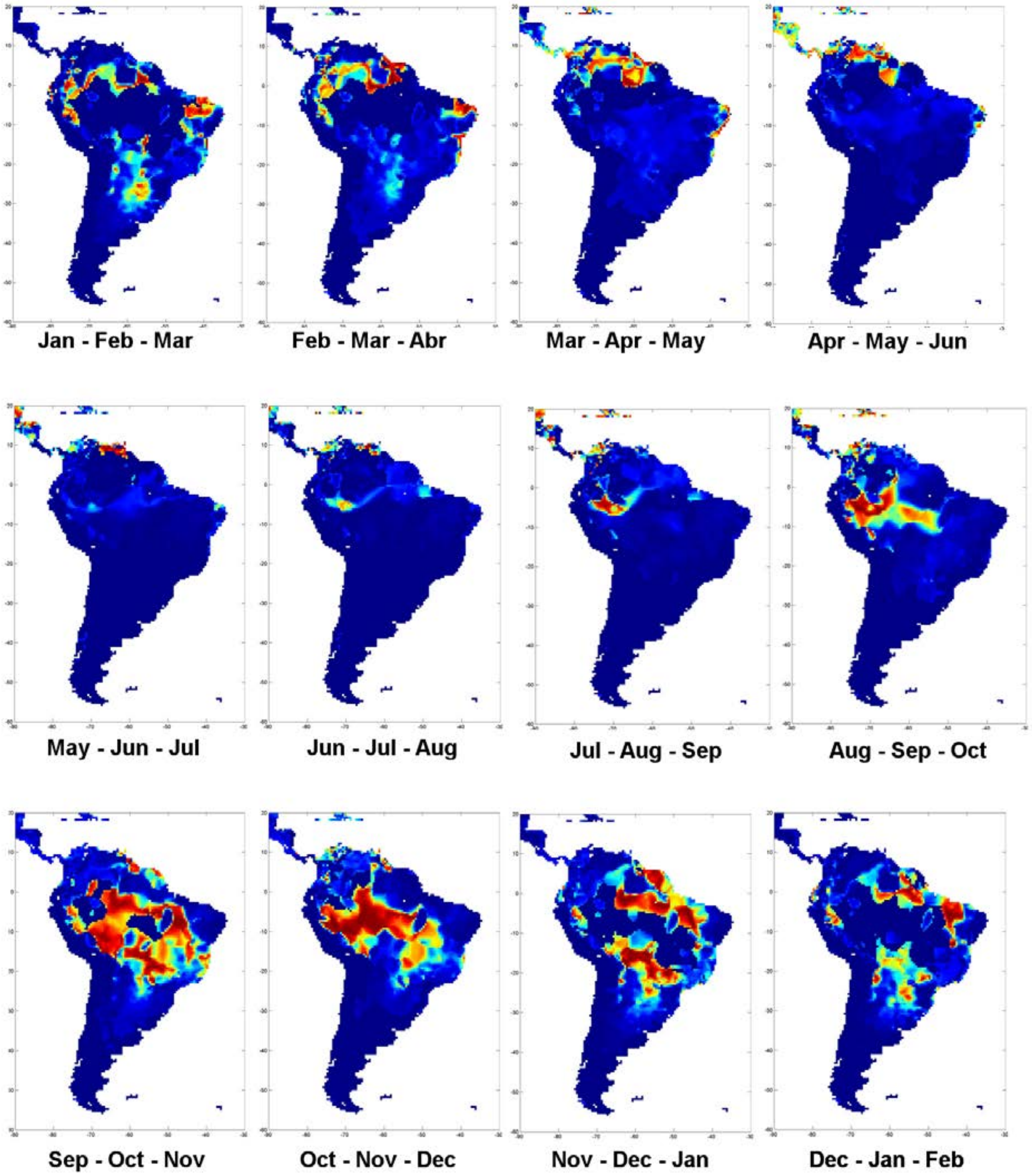


Figure 5.16: *Geographical zones belonging to the cluster L1N1S1 and its memberships for each quarter of the year. These maps show the zones that present similar patterns shifted one month. The similar zones are colored in red.*

Thus, we can use this information to migrate successful crops to other zones with similar behavior. In this example, three months were used as time window, so in this specific case we can use this results in fast growing crops such as beans, or berries.

For crops with a longer growing process larger time windows should be used. In this context, the analysis of spatio-temporal datasets in agroecology is inherent, because biological process occurs in time and space. Then, to extract knowledge from agroecological datasets implicates not only to understand the biological processes related with the physical environment (the spatial point of view) but also the environmental changes (the temporal point of view) that affects the life cycles of organisms. The sites with similar environmental conditions are generally taken as static in time, thus they suffer from the problem of not capturing the effects of environmental change through time. The methodology used in the aforementioned example, using FGHSON as clustering algorithm, is a new approach that allows to find sites with similar environmental conditions in the spatio-temporal context. This similarities can be defined as a set of frequently environmental states or regimes that present the combination of factors seen within that geographical area during the given time interval. This vision offers an accounting procedure that can track changes as a geographical location shifts from one group of environmental conditions to another through time. This dynamic tracking aspect is new to the similar environmental conditions concept and it should be of great potential utility.

5.5 Conclusions

As discussed in previous chapters the georeferenced data present specific challenges that must be overcome when they are being analyzed in a process of knowledge extraction. It was also mentioned the advantages of using soft competitive learning algorithms in the exploratory data analysis of these datasets, specifically for clustering and visualization. One of the most common applications in which clustering of georeferenced data is used is to find places with similar environmental characteristics. These similar places are used for several applications in agriculture, ecological triage and species migration. The finding of places with similar characteristics is a process that can be performed based on two main approaches. The first approach is called *whole clustering* and consists of grouping a set of individual time series without splitting them. For instance, if we have time series of ten years we use the full data without split it in years or months. Usually, the objective of this approach is to find similar time series in a long period of time. The second approach is the *subsequence clustering*. In this approach given a single time series, subsequences time series are extracted with a sliding window. Clustering is then performed on the extracted subsequence time series. This approach is used to find similar periods in the time series (e.g, similar years, months or seasons).

In this chapter we used self-organizing maps for clustering similar environmental zones by using both the whole clustering approach and the subsequence clustering approach. The results of both approaches were analyzed in order to find interesting patterns. The similar zones in a high-dimensional space were projected into a two dimensional space using the SOM algorithm. This projection allowed the visualization of the dynamics of the clusters, year by year. It helped to discover that some agroecological zones tend to stay in the same clusters, thus indicating the existence of stable clusters though time, while others change of clusters in time. Hence, this result suggests that the delineation of agroecozones should be adaptive given the changing dynamics of agroecozones in time, and that we should not follow the classical view of assuming an unique agroregionalizacion to define similar zones.

The second part of the chapter was devoted to present the FGHSON as an efficient algorithm to find similar patterns shifted in time. Several challenges are presented when clustering times series shifted in time, for instance: the overlap of clusters, the unknown number of clusters and to find the best resolution to visualize the clusters.

The FGHSON algorithm presents several features that render it more adequate than other algorithms to find similar patterns shifted in time. Usually, multivariate time series clustering techniques are based on K-means, nonetheless, K-means presents several disadvantages, namely: (1) the number of clusters has to be defined a-priori; (2) the K-means algorithm cannot represent the hierarchical relationships of the data, and (3) since K-means is a hard clustering algorithm it cannot represent overlapped clusters.

In this context, the FGHSON algorithm proved to be an useful tool to describe clusters of time series shifted in time, their hierarchical structure, and to visualize overlapping clusters using a fuzzy representation.

Chapter 6

Conclusions and future work

> “The most exciting phrase to hear in science, the one that heralds new discoveries, is not “Eureka!”, but “That’s funny...” ”
>
> Isaac Asimov

6.1 Summary

Clustering and visualization play a key role in the exploratory data analysis and the extraction of knowledge embedded in spatio-temporal geospatial datasets. However, new challenges are posed when dealing with the special characteristics of these datasets (as it is described in chapter 1 and chapter 1). For instance, their complex structures, large quantity of observations, high dimensionality, large variability in cluster shapes, and in many cases their unknown distribution.

The central aim of my thesis was to propose new algorithms and methodologies to clustering and visualization, in order to assist the knowledge extraction from spatio-temporal geo-referenced data with the objective of improving the decision making processes.

In this context, I proposed two original algorithms and methodologies in order to tackle the challenges aforementioned. One for clustering: the Fuzzy Growing Hierarchical Self-Organizing Networks (FGHSON) (described in section 2.5) and another used for exploratory visual data analysis: the tree-structured SOM component planes (discussed in section 3.2).

In addition, in section 4 we tested several well-known unsupervised soft competitive algorithms to cluster geo-referenced data. Although the techniques presented here have been used in several areas, to my knowledge there is not a single published study applying and comparing the performance of those techniques when dealing with spatio-temporal geospatial data as presented in this thesis.

In this thesis I also presented the implementation of a methodology to cluster spatio-temporal geo-referenced data through time (described in section 5.4). This methodology was applied to a real world problem whose aim was to find similar environmental patterns shifted in time.

Several results presented in this thesis have led to new contributions to agroecological knowledge, for instance in sugar cane (sections 3.3.1, and 5.3.1) and blackberry production (section 3.3.2).

Finally, in the framework of this thesis I developed several software tools: (1) a Matlab toolbox that implements the FGHSON algorithm, and (2) a program called BIS (Bio-inspired Identification of Similar agroecozones) an interactive graphical user interface which integrates the FGHSON algorithm with Google Earth in order to show zones with similar agroecologic characteristics.

6.2 Original contributions

In this thesis I present several original contributions related to the knowledge extraction from spatio-temporal geospatial data. They include the develop of new clustering and visualization algorithms, the comparison of the performance of existing algorithms, the development of methodologies to clustering spatio-temporal datasets through time, and contributions in the agroecological domain.

6.2.1 Algorithms contribution for clustering and visualization

In this thesis, I presented two original algorithms, the FGHSON and the tree-structured SOM component planes.

- **FGHSON**.

  The originality of the FGHSON lies in its capability to reflect the underlying structure of a dataset in a hierarchical fuzzy way. A hierarchical fuzzy representation of clusters is crucial in many areas where datasets include complex structures with large variability of cluster shapes, variances, densities and number of data points in each cluster.

  The most important characteristics of the FGHSON are described below:

  1. The training process does not necessarily lead to a balanced hierarchy, i.e. a hierarchy with equal depth in each branch. Rather, the specific distribution of the input data is modeled by a hierarchical structure, where some clusters require deeper branching than others.

2. It does not require an a-priori definition of the number of clusters.
3. The algorithm executes self-organizing processes in parallel. Hence, when dealing with large datasets the processes can be distributed reducing the computational time.
4. Only three parameters are necessary to the setup of the algorithm.

The capabilities of the FGHSON were tested with several benchmarks [18] and real world datasets obtaining satisfactory results in both cases. Namely, FGHSON was used in a trivial case (presented in section 4.2.1) in order to conceptualize the features of FGHSON when dealing with geospatial data. In a second case, it was used to find analogous environmental regions in Colombia (section 4.2.2). Finally, in section 5.4 FGHSON was applied to find similar environmental patterns shifted in time.

- **Tree-structured SOM component planes**

  The novelty of this algorithm lies in its ability to create a structure that allows the visual exploratory data analysis of large high-dimensional datasets. This algorithm creates a hierarchical structure of SOM component planes, arranging similar variables' projections in the same branches of the tree. Hence, similarities of variables' behavior can be easily detected (e.g. local correlations, maximum and minimum values and outliers). The tree-structured SOM Component planes was applied in a real world agroecologic problem (described in section 3.3) allowing to extract new knowledge about the sugar cane production.

6.2.2 Test of methodologies for clustering and visualization of georeferenced data

In this thesis, I tested three soft competitive learning algorithms. Two of them, well-known unsupervised soft competitive algorithms, namely the SOM and GHSOM; and the third was my original contribution, the FGHSON. These algorithms were used in order to find analogous environmental zones in Colombia. This application served us to evaluate the performance of those techniques when dealing with spatio-temporal geospatial data, thus allowing to highlight advantages and disadvantages of those algorithms when dealing with geospatial data.

6.2.3 Methodology contributions

- **Clustering spatio-temporal datasets through time**

  Currently, regions with similar agroecological characteristics (e.g. weather and soil) are represented with static and rigid boundaries derived by using data obtained by averaging observations over a period of time. Nevertheless, the boundaries of these similar regions normally change over time.

In this thesis we propose an original "dynamic approach" for clustering spatio-temporal datasets through time. This approach uses time-windows to capture temporal similarities and variations (e.g. regions with similar temperatures during a quarter of a year, but with different annual temperature average) and the FGHSON as a clustering algorithm.

There are a few clustering techniques that are a "dynamic approach" and in addition they present several drawbacks; for instance the number of clusters has to be defined prior to training, and their hierarchical clustering capabilities are quite limited.

In this thesis, the methodology developed is used in a case study where the objective was to find similar agroecozones. With this case study we highlight the advantages of using a "dynamic approach" compared to a "static approach", and how bio-inspired clustering algorithms can tackle the drawbacks when delineating agroecozones.

6.2.4 Agroecological knowledge contribution

- **In sugar cane productivity**

  In section 3.3.1, it was shown how the radiation of the first month after seed and the radiation of the first month before harvest are more correlated with the sugar cane productivity in comparison with the other variables. In addition, a local correlation is observed between a majority of high values of radiation and high productivity.

- **In sugar cane agroecoregionalizacion**

  In section 5.3.1, it was concluded that agroecoregionalization should not be static and that a new approach involving the concept of dynamical delineation of agroecozones must be consider. Taking into account this new paradigm, management crop decisions should be less subject to uncertainty related to the agricultural suitability of a particular area, despite the occurrence of seasonal weather phenomena as "El Niño" or "la Niña".

- **In Andean blackberry production**

  Self-Organizing Maps were used to group Andean blackberry yield from different sites according to similarity of growth conditions and management (as is explained in section 3.3.2). The SOM provided a straightforward manner to visualize the distribution of the variables that affected yield. Component planes generated by SOM illustrated the association of these variables with yield and identified two highly productive geographic areas enabling us to conclude that the optimal conditions for high yields are an average temperature between 16 and 18 °C, an effective soil depth between 60 and 70 cm, and low rainfall during the first month before harvest in poor external drainage locations or moderate to low rainfall in better drained areas.

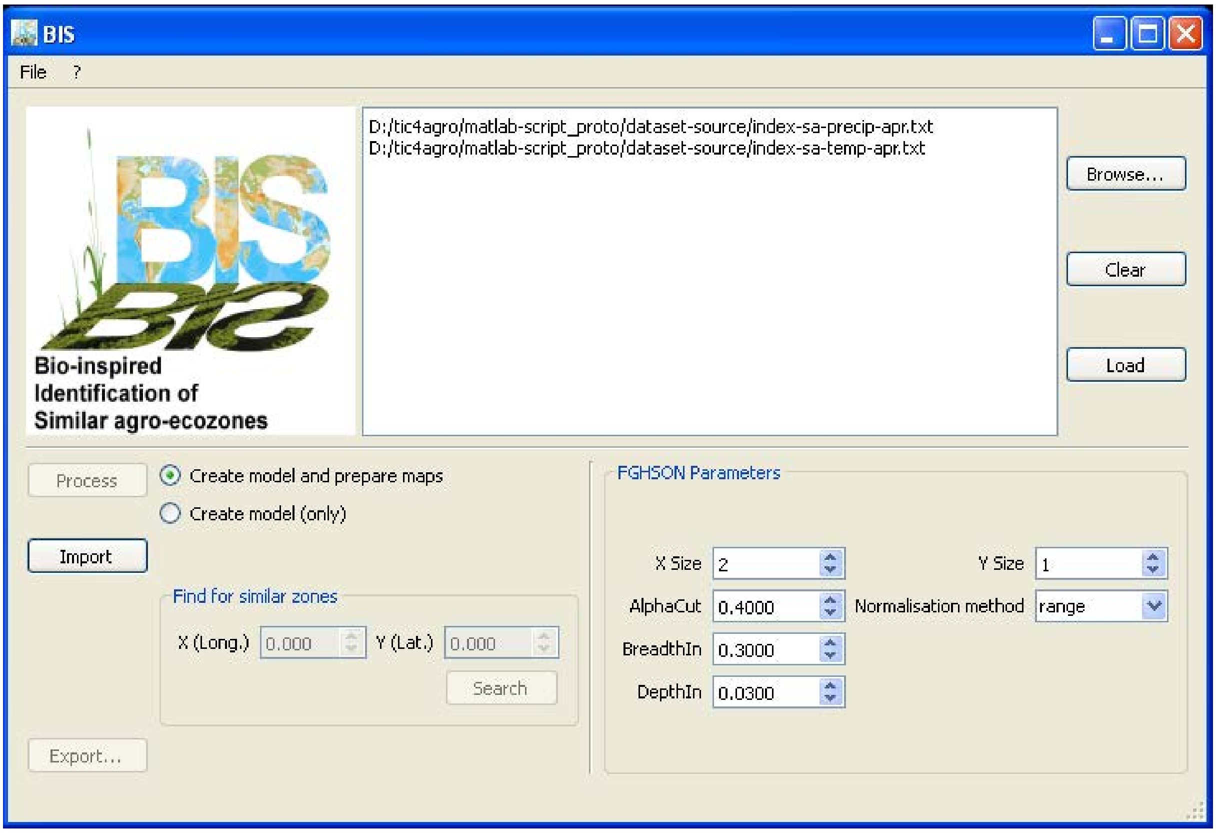
6.3 Contributions for researchers and practitioners

In order to contextualize the contribution of my thesis for researchers and practitioners, it is necessary to understand the new paradigms in agriculture research and the challenges that are imposed. A brief description is presented below.

Every time a farmer plants and harvests a crop represents a unique event or experiment. A premise is that if it were possible to characterize the production system in terms of management and the environmental conditions, and if information on the harvested product were collected from a large number of harvesting events under varied conditions, it should be possible to develop data-driven models that describe the production system. These models can then be used to identify appropriate growing conditions and improved management practices for crops that have received little attention from researchers. The analysis and interpretation of commercial production data in the context of naturally occurring variation in environmental and management, as opposed to controlled experimental data, requires novel approaches. Information is available on both variation in commercial and the associated environmental conditions production for many tropical fruits in Colombia.

The methodologies used once data sets have been established on a large number of well characterized events they must be consolidated. Analysis of a large consolidated data base with a large number of events from a wide range of conditions potentially provides more and better information on the crop response to variation in both the environment and management than the analysis of isolated evens. Furthermore, the analysis of large data sets obtained from field observations is complex and different methodologies are required for different circumstances. It is here where the methodologies described in this thesis play a key role, they allow the knowledge extraction of valuable information nested in those data.

In addition, as part of my thesis and in collaboration with the HEIG-VD (Haute Ecole d'ingénierie et de Gestion du Canton de Vaud) in Switzerland and the CIAT (international center for tropical agriculture) in Colombia it was develop the program BIS (Bio-inspired Identification of Similar agroecozones). BIS integrates the FGHSON algorithm with Google Earth in order to find zones with similar characteristics (see Figure 6.1 and Figure 6.2). This process can be perform with two objectives: (1) to find similar zones static in time; and (2)to find similar zones shifted in time. The program was installed in the CIAT to be used for researchers and practitioners.



D:/tic4agro/matlab-script_proto/dataset-source/index-sa-precip-apr.txt
D:/tic4agro/matlab-script_proto/dataset-source/index-sa-temp-apr.txt

● Create model and prepare maps
○ Create model (only)

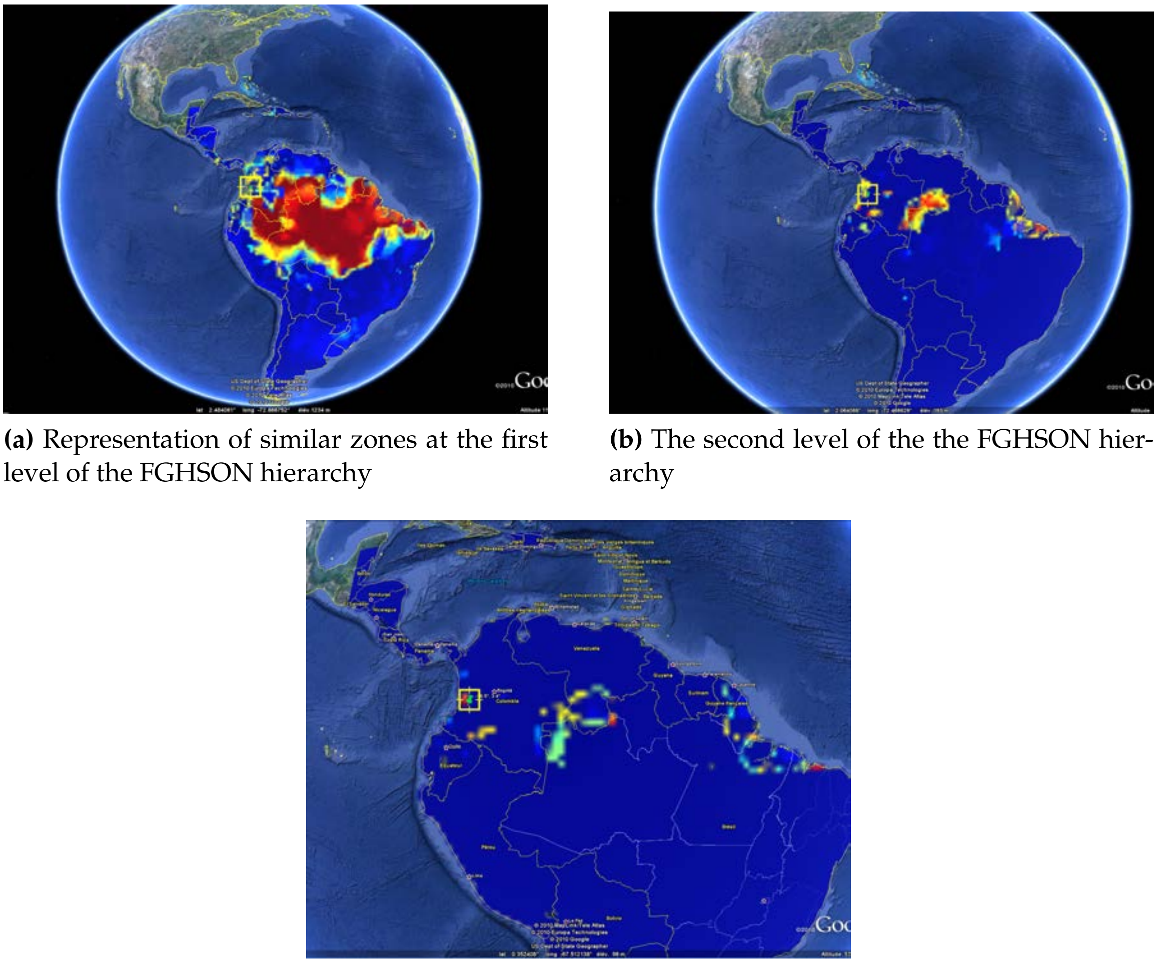
X (Long.) 0.000 Y (Lat.) 0.000

X Size 2 Y Size 1
AlphaCut 0.4000 Normalisation method range
BreadthIn 0.3000
DepthIn 0.0300

Figure 6.1: Bio-inspired Identification of Similar agroecozones.

6.4 Transferability of my work to other areas

Almost all clustering methods assume that each item must be assigned to exactly one cluster and are hence partitional. However, in a variety of important applications, overlapping clustering, wherein some items are allowed to be members of two or more discovered clusters, is more appropriate. For example, in biology, genes have more than one function by coding for proteins that participate in multiple metabolic pathways; therefore, when clustering microarray gene expression data, it is appropriate to assign genes to multiple, overlapping clusters. The same problem is presented in text classification and clustering, where classical algorithms cross-posted observations to multiple groups; the data must be subsequently manipulated to produce disjoint categories. Ideally, a clustering algorithm applied to this data would allow observations to be assigned to multiple groups and would rediscover the original cross-posted observations. The FGHSON algorithm presents interesting characteristics described in this thesis that make it a potential option to tackle the problem of the overlapping clustering in many research and practical areas.



(a) Representation of similar zones at the first level of the FGHSON hierarchy

(b) The second level of the the FGHSON hierarchy

(c) The third level of the FGHSON hierarchy

Figure 6.2: *Screenshots of the BIS program*

6.5 Future work

Deepening and improving the FGHSON algorithm

- As mentioned in section 2.5, two of the parameters that govern the self-organizing process of the FGHSON are based on the quality of the clusters. This quality is measured by using the quantization error. In the literature, we can find several measures of cluster quality for specific problems. In this context, FGHSON could be implemented and tested using other cluster quality measures.

- The parameter φ (the well-known α-cut) represents the minimal membership degree of an observation to be represented by a prototype. In some cases this membership value could increase or decrease obtaining as a result larger or smaller clusters. Future work will be focused on an automatic way to choose the parameter φ based on the clustering problem to tackle.

- Another adaptation to the algorithm could be to use a non-fixed value for the parameter φ. As was described in section 2.5, in the depth growth process of the FGHSON, the samples used for training are a fraction of the whole data. This portion of data is selected according to a minimal membership degree (φ). The value of φ is selected at the beginning of the FGHSON training and is the same during all the self-organizing process. Taking into account that when we move deeper on the hierarchical structure created by the FGHSON the number of observations in each cluster becomes smaller, we could find a way to vary the value of φ (for instance according to the clusters size) in order to obtain a better representation of the clusters.
- The fact that FGHSON creates fuzzy clusters can be used as means to extract fuzzy rules. It can be an interesting starting point for future works. This could enhance the interpretability of the knowledge embedded in clusters.

Exploring the tree-structured SOM component planes algorithm

- Although, the tree-structured SOM component planes is a good method to visually exploratory data analysis, it would also be interesting to have a numeric index of the similarity between clusters in each branch of the tree. Future work will be focused on the study of similarity measurements to improve the tree-structured clusters interpretation.

Clustering of spatio-temporal datasets through time

- To find the right combination of time windows (e.g. days or months) and shifts to find similar agroecologic zones is a hard computing process and it is problem dependent. New strategies must be developed in order to accelerate this process.

Appendix A

First article

Paper published on the *Proceedings of the ICANN 2007 Conference*. Marques de Sa et al. (Eds.), Porto, Portugal. 9 - 13 September, Part I, LNCS 4668, Springer, pp. 379-388.

Improving the correlation hunting in a large quantity of SOM component planes. Classification of agro-ecological variables related with productivity in the sugar cane culture.

Miguel A. Barreto S[1,2,3] and Andres Perez-Uribe[2].

1. Université de Lausanne, Hautes Etudes Commerciales (HEC), Institut des Systèmes d'Information (ISI), Switzerland
2. REDS Institute, University of Applied Sciences of Western Switzerland (HEIG-VD)
3. BIOTEC, Precision Agriculture and the Construction of Field-Crop Models for Tropical Fruit Species, Colombia

Abstract A technique called component planes is commonly used to visualize variables behavior with Self-Organizing Maps (SOMs). Nevertheless, when the component planes are too many the visualization becomes difficult. Thus, a methodology has been developed to enhance the component planes analysis process. This methodology improves the correlation hunting in the component planes with a tree-structured cluster representation based on the SOM distance matrix. The methodology presented here was used in the classification of similar agro-ecological variables and productivity in the sugar cane culture. Analyzing the obtained groups it was possible to extract new knowledge about the variables more related with the highest productivities.

A.1 Introduction

A traditional technique to detect dependencies between variables is the use of scatter plots. In addition, when the variables are more than a pair, it is possible to generate a scatter plot matrix with several sub-plots where each variable is plotted against each other variable. However, in this technique the number of pairwise scatter plots increases quadratically with the number of variables [62]. This type of visualization is thus not practical in applications where the analysis of many variables is necessary.

Another visualization technique consists on using the so-called SOM components planes [75], the number of sub-plots grows linearly with the number of variables. In addition, this technique is able to cluster variables with similar behaviors. Every SOM component plane is formed by the values of the same component in each prototype vector. Therefore, they can be seen as a sliced version of the map [116]. After plotting all component planes, relations between variables can be easily observed. The task of organizing similar components planes in order to find correlating components is called correlation hunting [123]. However, when the number of components is large it is difficult to determine which planes are similar to each other. Different techniques can be used to reorganize the component planes in order to aid the correlation hunting. The main idea is to place correlated components close to each other.

One of the most often used techniques in correlation hunting is the projection of the component planes on another plane. This projection can be done using, e.g. another SOM, as the work of Vesanto et Ahola. [123]. Another interesting approach was introduced by Sultan et al [113] they presented a binary tree-structured vector quantization (BTSVQ) algorithm. The BTSVQ uses SOMs for visualization, and the partitive k-means clustering, to group similar component planes and organizing them into a binary tree structure. This hybrid algorithm is used to improve the process of data analysis and visualization of gene expression profiles.

The approach of Vesanto and Ahola [123] is adequate to organize the component planes, but it is an inefficient tool for visualization when the number of planes is large. It is difficult to clearly observe the relationships between the component planes due to the quantity of planes to show in a same space. Sultan's algorithm is more adequate to organize a large quantity of data. Its binary tree structure allows the analysis of groups of component planes at different levels. Nevertheless, the algorithm proposed to organize the SOM component planes make use of k-means as a clustering algorithm, and the SOM is only employed to show the data.

In this paper we present a methodology to enhance the visualization and analysis process of a large quantity of component planes. This methodology uses a SOM to project the component planes. This SOM is partitioned into in clusters with a technique based on the SOM distance matrix. A tree structure is generated from different clustering levels of the SOM, in order to clearly visualize the groups of component planes. The methodology presented here was used in the classification of similar agroecological variables and productivity in the sugar cane culture. Analyzing the obtained groups it was possible to extract new knowledge about the variables more related with the highest productivity.

This paper presents the following structure. In the next section the methodology is explained. Third section focuses on the application of the methodology to the sugar cane case. Finally, in section four conclusions and future extensions of this work are presented.

A.2 Methods

A.2.1 Self-Organizing Maps

A Self-Organizing Map (SOM) [75] is composed of artificial neurons situated on a but regular low-dimensional grid. This grid can be in one, two or three dimensions, generally two are used. The neurons in the grid have rectangular or hexagonal form. Each neuron i represents an n-dimensional prototype vector $mi = [mi_1, \ldots, mi_s]$, where s is equal to the dimension of the input space. In the beginning of the training process the prototype vectors are initialized with random values. On each step of the training a data vector x from the input data is selected and presented to the SOM. The unit mc closest to x is located into the map, this winner unit is called the best-matching unit (BMU). The BMU and its neighboring prototype vectors on the grid are moved in the direction of the sample vector, $mi = mi + \alpha(t)h_{ci}(t)(x - mi)$ where $\alpha(t)$ is the learning rate and $h_{ci}(t)$ is a neighborhood kernel centered on the winner unit c. The learning rate and neighborhood kernel radius decrease monotonically with time. Through the iterative training, the SOM organizes the neurons so that neurons that represent similar vectors in the input space are located on the map in contiguous zones, trying to conserve the linear or nonlinear relations of the input space.

A.2.2 SOM component planes

SOM allows a straightforward visual inspection because the prototype vectors are organized according to their similarity in a low-dimensional grid. This feature is helpful when it is needed to handle large multidimensional vectors. A way to improve this inspection is by means of the component plane representation. A component plane (CP) is a projection of the same input variable from each vector prototype on a grid. For example, having the prototype vectors $m1, \ldots, mi$. The component plane which represents the first input variable will be formed by $CP1 = [m1_1, \ldots, mi_1]$ in general $CPs = [m1_s, \ldots, mi_s]$ where s is equal to the dimension of the input space. Hence, the number of component planes will be equal to the input space dimension. In addition, the component planes are visualized in an grid identical to that of the SOM. However, the difference between the component plane grid and the SOM grid is that on this new grid each neuron does not plot a prototype vector, instead it represents a component of this vector. Each component in the component plane grid conserves the same place that the prototype vector in the SOM grid. Finally, every component on the component plane is visualized by giving to each neuron a color according to the relative value of the respective component in that neuron. As a result, it is possible to obtain color maps of the component planes in order to compare them and look for relations between variables.

A.2.3 Correlation Hunting

The component planes analysis can be a tool for discovering relations between variables. Comparing the planes, it is possible to observe similar patterns in identical positions indicating correlation between the respective components. Even, local correlations can be found if two parameter planes resemble each other in some regions. The process to find these relationships is called correlation hunting. The expression correlation does not include just linear correlations, but also nonlinear and local or partial correlations between variables [123].

The correlation hunting can be realized manually or automatically. However, in many cases the manual analysis is difficult because usually the component planes are not ordered. In addition, the comparison becomes more difficult when the number of components increases. In order to overcome this drawback, it is possible to apply reorganization of the component planes such that similar component planes could be located close to each other [122]. To do this, the component planes can be projected on a plane. The projection could be done using, e.g., Sammon's mapping [104], CCA [40] or another SOM. In this paper SOM was used as projection technique. The projection process using SOM is the following:

1. Each component plane is transformed into a vector and then normalized to ignore different scales of the components.

2. The vectors are further processed by calculating a measure of distance between them.

3. The measure of distance between component planes i and j can be defined as the value of the correlation of each map position, formally $distCP(i,j) = mc * (CP_i, CP_j)$ where mc is a suitable measure of correlation, in this paper the Pearson correlation coefficient is used.

4. A covariance matrix is generated with the obtained distances.

5. The vectors of the covariance matrix are used as inputs to a new SOM.

6. Each component plane grid from the old SOM is projected to the new SOM.
7. This projection is realized locating in the place of the BMUs of the new SOM, the respective component planes grids from the old SOM. Hence, planes with high correlation are located near each other.

An advantage of using a SOM for component plane projection is that the placements of the component planes can be shown on a regular grid. In addition, an ordered presentation of similar components is automatically generated. A disadvantage is that the choice of grouping variables is left to the user. This task is complicated when the number of component planes is large.

A.2.4 Distance matrix based clustering of the SOM

Having a projection of component planes in a new SOM, it is possible to use a method to cluster the new SOM in order to find component plane groups. For example, partitive (e.g., k-means) or agglomerative clustering algorithms (e.g., agglomerative hierarchical clustering) are used to cluster the prototype vectors [124]. Nevertheless, those approaches do not take into account the SOM neighborhoods. To cope with this drawback, a cluster distance function can be used to consider the neighborhoods into account. The U-matrix [118] had been used as an effective cluster distance function [126]. The U-matrix visualizes distances between each map unit and its neighbors, thus it is possible to visualize the SOM cluster structure. This method is usually applied to select clusters from the map by hand. This selection is normally subjective because it is based on the visual perception of each person. Vellido et al. [121] proposed an algorithm to do distance matrix based clustering automatically. In this algorithm, the U-matrix is used to identify cluster centers from the SOM. The rest of the map units are then assigned to the cluster whose center is closest. The algorithm is the following:

1. Compute the distance matrix local minima. This is done by finding the set of map units i for which:

   $$f(m_i, N_i) \leq f(m_j, N_j), \forall j \in N_i \tag{A.1}$$

   where N_i denoted the set of neighboring map units of the map unit i and $f(m_i, N_i)$ is some function of the set of neighborhood distances $\|m_i - m_j\|$ $j \in N_i$, associated with map unit i. In the experiments, median distance was used. The set of local minima may have units which are neighbors of each other. Only one minimum from each such group is retained.
2. For the initialization, let each local minimum be one cluster: $C_i = m_i$. All other map units j are left unassigned.

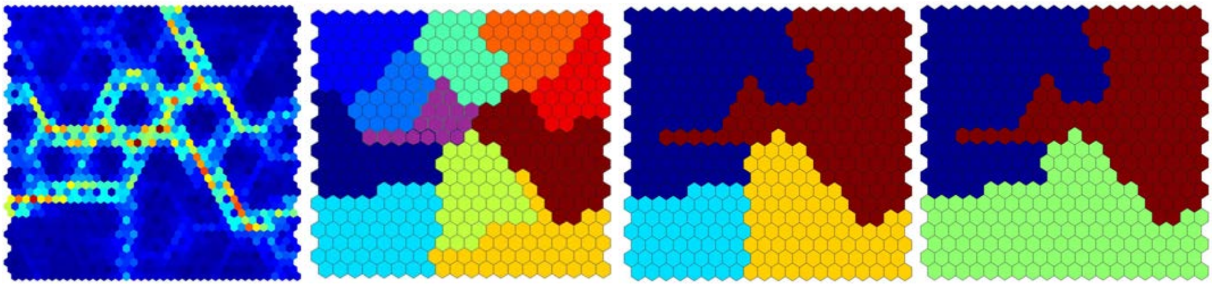
3. Calculate distance $d(C_i, m_j)$ from each cluster Ci to (the cluster formed by) each unassigned map unit j.

4. Find the unassigned map unit with smallest distance and assign it to the corresponding cluster.

This algorithm provides an automatic discrimination of clusters which permits an easier exploration of similar component planes. Although, when the number or component planes is large is desirable an approach that permits to organize the component planes in a structure, and to analyze clusters at several levels of detail. Hence, the idea of considering super-clusters, consisting of several sub-clusters, making easier the analysis of the large quantity of planes.

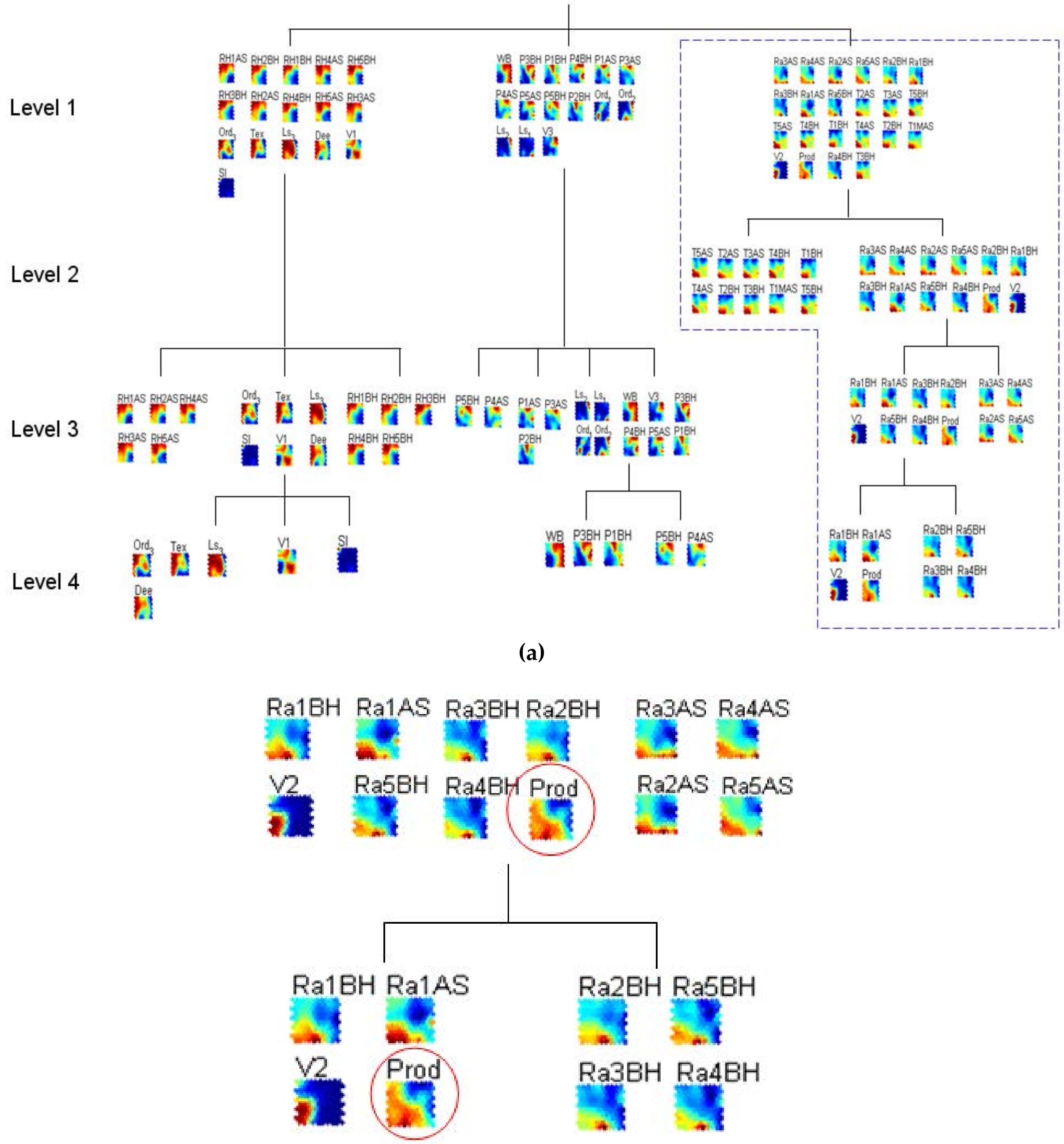
A.2.5 Tree-structured component planes clusters representation

In order to analyze the component planes clusters at several detail levels, it is possible to make a tree-structured representation of them. The Vellido's algorithm is used to obtain different partitioning levels of the clustering of the SOM in an attempt to achieve this goal. The Vellido's algorithm provides a partitioning of the map into a set of base clusters. The number of clusters is equal to the number of local minima on the U-matrix; allowing different levels of clustering. Regarding the equation A.1 it is possible to observe that the local minima depends on the set of neighboring map units (Ni) from the map unit i. Hence, Ni depends on the amount of neighbors chosen to i. As a result, when the neighborhood is large, the number of local minima is small and therefore the number of clusters too. Varying the neighborhood size it is possible to obtain different cluster quantities, see figure A.1. So, it is possible to find different cluster levels, and as a result, build a tree structure that will permit to have several levels of detail, see figure A.2.



(a) U-matrix **(b)** Level 3 **(c)** Level 2 **(d)** Level 1

Figure A.1: *By using the U-matrix shown in (a) the Vellido's algorithm is applied to obtain different cluster granuralities. (b) By using a neighborhood radius set to 3 we obtain a map with 11 clusters. This map form the third level of the tree structure (c) Setting the neighborhood radius to 3 give us a map of 4 clusters. Based on this map is built the second level of the tree (d) With a radius set to 5 we obtain a map form by 3 clusters. This map is the first level of the tree.*


Level 1
Level 2
Level 3
Level 4
RH1AS RH2BH RH1BH RH4AS RH5BH
RH3BH RH2AS RH4BH RH5AS RH3AS
Ord Tex Ls Dee V1
SI
WB P3BH P1BH P4BH P1AS P3AS
P4AS P5AS P5BH P2BH Ord Ord
Ls Ls V3
Ra3AS Ra4AS Ra2AS Ra5AS Ra2BH Ra1BH
Ra3BH Ra1AS Ra5BH T2AS T3AS T5BH
T5AS T4BH T1BH T4AS T2BH T1MAS
V2 Prod Ra4BH T3BH
T5AS T2AS T3AS T4BH T1BH
T4AS T2BH T3BH T1MAS T5BH
Ra3AS Ra4AS Ra2AS Ra5AS Ra2BH Ra1BH
Ra3BH Ra1AS Ra5BH Ra4BH Prod V2
RH1AS RH2AS RH4AS
RH3AS RH5AS
Ord Tex Ls
SI V1 Dee
RH1BH RH2BH RH3BH
RH4BH RH5BH
P5BH P4AS
P1AS P3AS
P2BH
Ls Ls WB V3 P3BH
Ord Ord P4BH P5AS P1BH
Ra1BH Ra1AS Ra3BH Ra2BH Ra3AS Ra4AS
V2 Ra5BH Ra4BH Prod Ra2AS Ra5AS
Ord Tex Ls V1 SI
Dee
WB P3BH P1BH P5BH P4AS
Ra1BH Ra1AS Ra2BH Ra5BH
V2 Prod Ra3BH Ra4BH
Ra1BH Ra1AS Ra3BH Ra2BH Ra3AS Ra4AS
V2 Ra5BH Ra4BH Prod Ra2AS Ra5AS
Ra1BH Ra1AS Ra2BH Ra5BH
V2 Prod Ra3BH Ra4BH

(a)

(b)

Figure A.2: *Different levels of detail of the tree-structured component planes representation. (a) The entire tree structure. (b) A zoom showing the branch where production is located at the levels three and four. Productivity is highlight with a red circle. Note the local correlation between high and medium values of Ra1BH, Ra1AS (hue in red, orange and green) and high productivity (colored in red and orange).*

A.2.6 Problem description

SOM has proved to be effective for the exploratory analysis of agro-ecologic data and has become a very useful technique in ecological modeling [79]. SOMs are recommended in cases when it is essential to extract features out of a complex data set [31]. Moreover, it is useful for generating easily comprehensible low-dimensional maps, improving the visualization and data interpretation [34, 54]. For these reasons, methodologies based in SOM were selected as tools for exploring the data in this study case. The objective of this case study was to determinate which variables are more related to the productivity in the sugar cane culture in a specific region. The methodologies previously shown were used to classify zones with similar productivity, in order to find similar patterns of behavior. Finally, analyzing these patterns it was possible to acquire new knowledge about the relationship between the agro-ecological variables and productivity. A more detailed description of the problem is presented as following:

A plant is affected by diverse variables (e.g., climate, soil) during its life. These variables have different effects on the plant at different moments of its development (e.g., germination, flowering). Moreover, the combination and/or change of these variables in certain moments determines the development states of the plant. This mixture of factors finally determines the crop production. For example, in the sugar cane case, expert knowledge indicates that the most relevant periods are the beginning and the end of plant development. In the first months (after sowing) the vegetative structure is formed (e.g., leafs grows allowing the photosynthesis process), in this moment the water is very important to improve the development of the plant. During the last months (approximately thirteen months after sowing) the plant accumulates the major part of saccharose. In this moment not much water is essential because the plant is totally developed. These periods are the most important in the agricultural productivity. Accordingly, to determine how and when the variables affect the plant development would be very helpful to support decision making (e.g. in what moment to seed and/or to harvest in order to obtain a better productivity).

A.2.7 Classification of agro-ecological variables related with productivity

The database used was provided by a sugar cane research center located in the region under study. The data base contains information collected during seven years (1999 to 2005). The agro-ecological variables used for this experiment are listed as follows. Climate variables are Temperature Average (T), Relative Humidity Average (RH), Radiation (Ra), and Precipitation (P). Soil variables are Order (Ord), Texture (Tex) and Depth (Dee). Topographic variables are Landscape (Ls) and Slope (Sl). Other variables are Water Balance (WB) and Variety (V). Finally, productivity (P) of each cultivated zone. As it was mentioned before, the most relevant periods in the sugar cane are the beginning and the end of plant development. Therefore, it is possible not using all the climate data set and to use only the data from $1, \ldots, x$ Months After Sowing (xAS) and $1, \ldots, x$ Months Before Harvest (xBH). In our case study $x = 5$ was used. Soil variables and Variety were ordered using a presence/absence coding, 0 represents presence and 1 absence. As a result, the vector which defines a cultivated zone (CZ) is compound of 54 variables, 1328 vectors were used representing each one the characteristics of a cultivated zone.

All the variables were scaled [-1,1] in order to allow their comparison in magnitude. Then, it was created a matrix with 1328 vectors CZs ($CZmatrix$) composed by 54 variables each one. Notice that the output of this sugar cane model is the productivity. Nevertheless, in this case the productivity was used as input in order to find the component planes related with. The $CZmatrix$ was used as input for a SOM with 400 neurons (20x20) and it was trained with the batch algorithm. With this SOM, it was possible to generate 54 component planes, one for each agro-ecological variable. These component planes were projected in a new SOM composed of 400 neurons (20x20) and it was trained with the batch algorithm. Finally, this last map was clustered and the clusters were organized. For this aim it was used the algorithm for tree-structured component planes clusters representation showed in the previous section. The results can be observed in figure A.2.

Some interesting aspects can be found here. When $n = 3$ (where n is the neighborhood), it is possible to locate in a same cluster the temperature, radiation and production, each component plane with similar patterns, figure 1.b on dotted line and figure 2.a. In addition, it is possible to view when $n = 1$ that radiation of first month after seed, radiation of first month before harvest and productivity present similar patterns, figure A.2. Thanks to the tree-structured representation, it is easier to group the clusters to facilitate the observation of local correlations. As an example, in the right side and in the top of left side of the component planes, figure A.2, it was possible to see that when the productivity was high most of the values of Ra1BH and Ra1AS were high too (represented for a dark gray). Drawing the BMUs of the component planes productivity, Ra1BH and Ra1AS in a scatter plot, figure A.3, it is possible to detect high values of productivity when there are high values of Ra1BH and Ra1AS.

As a conclusion, the radiation of the first month after seed and the radiation of the first month before harvest are more correlated with the productivity than the other variables. In addition, a local correlation is observed between a majority of high values of radiation and high productivity.

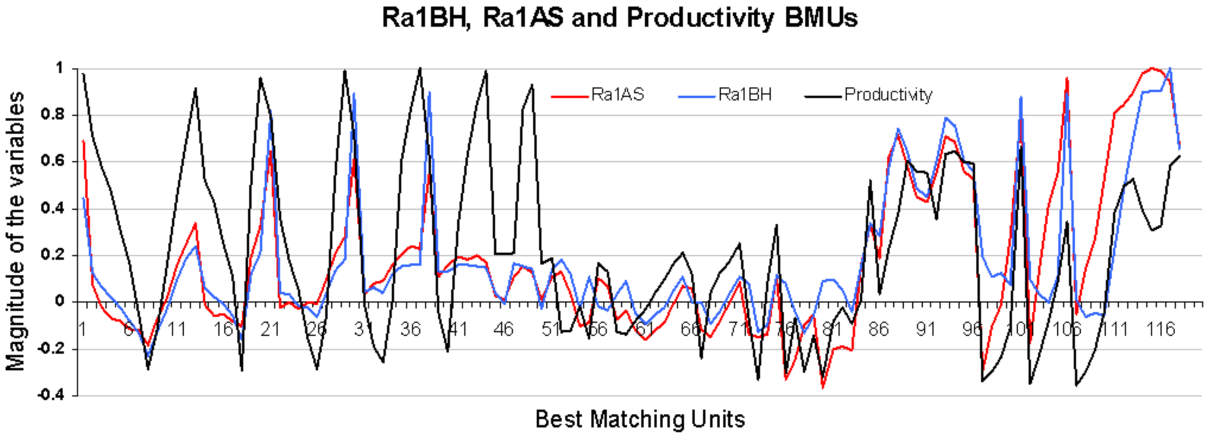


Figure A.3: *BMUs of the component planes: productivity, radiation 1 month before harvest (Ra1BH) and radiation 1 month after seed (Ra1AS).*

A.3 Conclusion

This paper has presented a methodology to enhance the component planes analysis process. This methodology improves the correlation hunting in the component planes with a tree-structured clusters representation based on the SOM distance matrix. This tree-structured representation permits the analysis of component planes clusters at several levels of detail . This methodology can be applied in cases where the number of component planes is very large, witch is quite often in agro-ecological modeling. As an case study, the methodology presented here was used in the classification of zones with similar agro-ecological conditions and productivity in the sugar cane culture. Analyzing the obtained groups of agro-ecological variables and cultivated zones it was possible to find a relationship between the radiation during the first month after seed, the first month before harvest, and high productivity. More analysis can be made in order to improve the decision support in the sugar cane culture based on the aforementioned methodology. This paper shows only a part of this work. Future work will be focus on the analysis of other patterns.

Although, the tree-structured is a good method to show clusters, it would also be desirable to obtain a measurement of similarity between clusters in each branch of the tree. Future work will be focused on the study of a similarity measurements to improve the tree-structured clusters representation.

Acknowledgements This work is part of a cooperation project between BIOTEC, CIAT, CENICAÑA (Colombia) and HEIG-VD (Switzerland) named ŞPrecision agriculture and the construction of field-crop models for tropical fruitsŤ. The economical

support is given by several institutions in Colombia (MADR, COLCIENCIAS, ACCI) and the State Secretariat for Education and Research (SER) in Switzerland.

Appendix B

Second article

Paper published on Studies in Computational Intelligence series, Springer Verlag, Vol 258, 2009, pp. 261-280.

Tuning Parameters in the Fuzzy Growing Hierarchical Self-Organizing Networks

Miguel A. Barreto S[1,2,3], Andres Perez-Uribe[2], Carlos A. Pena-Reyes [2] and Marco Tomassini[1].

1. Université de Lausanne, Hautes Etudes Commerciales (HEC), Institut des Systèmes d'Information (ISI), Switzerland
2. REDS Institute, University of Applied Sciences of Western Switzerland (HEIG-VD)
3. BIOTEC, Precision Agriculture and the Construction of Field-Crop Models for Tropical Fruit Species, Colombia

Abstract Hierarchical Self-Organizing Networks are used to reveal the topology and structure of datasets. Those methodologies create crisp partitions of the dataset producing tree structures compound of prototype vectors, permitting the extraction of a simple and compact representation of a dataset. However, in many cases observations could be represented by several prototypes with certain degree of membership. Nevertheless, crisp partitions are forced to classify observations in just one group losing information about the real dataset structure. To deal with this challenge we propose

the Fuzzy Growing Hierarchical Self-Organizing Networks (FGHSON). FGHSON are adaptive networks which are able to reflect the underlying structure of the dataset, in a hierarchical fuzzy way. These networks grow by using three parameters which govern the membership degree of data observations to the prototype vectors and the quality of the hierarchical representation. However, different combinations of the values of these three parameters can generate diverse networks. This chapter explores how these combinations affect the topology of the network and the quality of the prototypes; in addition it is presented the motivation and the theoretical bases of the algorithm.

B.1 Introduction

We live in a world full of data. Every day we are confronted with the handling of large amounts of information, this information is stored and represented as data, for further analysis and management. One of the essential means in dealing with these data is to classify or group them into a set of categories or clusters. In fact, as one of the most ancient activities of human beings [11], classification plays a very important role in the history of human development. In order to learn a new object or distinguish a new phenomenon, people always try to look for the features that can describe it, and further compare it with other known objects or phenomena, based on the similarity or dissimilarity, generalized as proximity, according to some certain standards or rules.

In many cases, classification must be done without a priori knowledge of the classes in which are divided the dataset (unlabeled pattern). This kind of classification is called clustering (unsupervised classification). On the contrary, the discriminant analysis (supervised classification) is made providing a collection of labeled patters; so the problem is to label a newly encountered, yet unlabeled, pattern. Typically, the given labeled patterns are used to learn the descriptions of classes which in turn are used to label a new pattern. In the case of clustering, the problem is to group a given collection of unlabeled patterns into meaningful clusters. In a sense, labels are associated with clusters also, but these category labels are data driven; that is, they are obtained solely from the data [68, 131].

Even though the unsupervised classification presents many advantages over supervised classification[1] it is a subjective process in nature. As pointed out by Backer and Jain [13], "in cluster analysis a group of objects is split up into a number of more or less homogeneous subgroups on the basis of an often subjectively chosen measure of similarity (i.e., chosen subjectively based on its ability to create "interesting" clusters), such that the similarity between objects within a subgroup is larger than the similarity between objects belonging to different subgroups". Clustering algorithms partition

[1] For instance, no extensive prior knowledge of the dataset is required, and it can be detected "natural" groupings in feature space.

data into a certain number of clusters (groups, subsets, or categories). There is no universally agreed upon definition [46].

Thus, methodologies to evaluate clusters to different levels of abstraction in order to find “interesting" patterns are useful; these methodologies could help to improve the analysis of cluster structure creating representations of them so facilitating the selection of clusters of interest. Methods for tree structure representation and data abstraction have been used for this task, allowing to reveal the topology and organization of clusters.

On the one hand, hierarchical methods are used to help explain the inner organization of datasets, since the hierarchical structure imposed by the data produces a separation of clusters that is mapped onto different branches. Hierarchical clustering algorithms organize data into a hierarchical structure according to the proximity matrix. The results of Hierarchical clustering are usually depicted by a binary tree or dendrogram. The root node of the dendrogram represents the whole data set and each leaf node is regarded as a data object. The intermediate nodes, thus, describe to what extent the objects are proximal among them; and the height of the dendrogram usually expresses the distance between each pair of objects or clusters, or an object and a cluster. The ultimate clustering results can be obtained by cutting the dendrogram at different levels. This representation provides very informative descriptions and visualization for the potential data clustering structures, especially when real hierarchical relations exist in the data, like the data from evolutionary research on different species of organisms. Therefore, this hierarchical organization enables to analyze complicated structures as well as it allows the exploration of the dataset at multiple levels of detail [131].

On the other hand, data abstraction permits the extraction of a simple and compact representation of a data set. Here, simplicity is either from the perspective of automatic processing (so that a machine can perform further processing efficiently) or is human-oriented (so that the representation obtained is easy to comprehend and intuitively appealing). In the clustering context, a typical data abstraction is a compact description of each cluster, usually in terms of cluster prototypes or representative patterns such as the centroid of the cluster [41]. Soft competitive learning methods [52] are employed on data abstraction in a self-organizing way. These algorithms attempt to distribute a certain number of vectors (prototype vectors) in a possibly low-dimensional space. The distribution of these vectors should reflect (in one of several possible ways) the probability distribution of the input signals which in general is not given explicitly but only through sample vectors. Two principal approaches have been used for this purpose. The first one is based on a fixed network dimensionality (e.i. Kohonen maps [74]). In the second approach, not fixed dimensionality is imposed on the network; hence, this networks can automatically find a suitable network structure and size thought a controlled growth process [82].

Different approaches have been introduced in order to combine the capabilities of tree structure of the hierarchical methods and the advantages of soft competitive learning methods used to data abstraction [84, 63, 43, 98, 114, 80]. Hence, obtaining networks being capable of representing the structure of the clusters and their prototypes in a hierarchical self-organizing way. These networks are able to grow and adapt their structure in order to represent the characteristics of the clusters in the most accurate manner. Although these hybrid models provide satisfactory results, they generate crisp partitions of the datasets. The crisp segmentations tend to allocate elements of the dataset in just one branch of the tree in each level of the hierarchy and assign just one prototype as representation of one cluster, so the membership to the other branches or prototypes is zero. However, in many applications crisp partitions in hierarchical structures are not the optimal representation of the clusters, since some elements of the dataset could belong to different clusters or branches with a certain degree of membership.

One example of this situation is presented in Geographic Information Systems (GIS) applications. One of the topics treated by GIS researchers refers to the classification of geographical zones with similar characteristics as climate, soil and terrain (conditions relevant to agricultural production) in order to create the so called agro-ecological zones (AEZ) [49]. AEZ provide the frame for various applications, such as quantification of land productivity, estimation of the land's population supporting capacity, and optimization of land resources use and development. Hence, many institutions, governments and enterprises require to know in advance at what AEZ belongs certain geographical region of interest (that means allocate the region in certain AEZ cluster), in order to apply proper policies to invest, for instance in new cropping systems looking for a economic viability, and sustainability. However, the geographical region of interest can vary in range of resolution depending on the application or the context (countries, states, cities, parcels). In addition the fuzzy and implicit nature of the geographic zones (in which geographical boundaries are not hard, but rather soft boundaries) transform the boundaries of the AEZ in zones of transition rather than sharp boundaries. Thus, the soft boundaries make it possible that regions in the middle of two AEZ have certain membership to both AEZ, in this specific example. Hence, the clustering method to deal with the aforementioned situation has to provide views of AEZ at multiple levels (preferably in a hierarchical way); in addition, it should be capable of discovering fuzzy memberships of the interest geographical regions to the AEZ.

With the purpose of representing degrees of membership, fuzzy logic is a feature that could be added to the characteristics of hierarchical self-organized hybrid models in order to deal with similar problems as the one that has been aforementioned. We propose, thus Fuzzy Growing Hierarchical Self-Organizing Networks (FGHSON), which intends to synergistically combine the advantages of Self-Organizing Networks, hierarchical structures, and fuzzy logic. FGHSON are designed to improve the analysis of datasets where it is desirable to obtain a fuzzy representation of a dataset in a hierarchical way, then discovering its structure and topology. This new model will be able to obtain a growing hierarchical structure of the dataset in a self-organizing fuzzy manner. These kind of networks are based on the Fuzzy Kohonen Clustering Networks (FKCN) [22] and Hierarchical Self-Organizing Structures (HSS) [77, 86, 84, 98].

This book chapter is organized as follows: In the next section the Hierarchical Self-Organizing Structures and the Fuzzy Kohonen Clustering Networks will be explained serving as a base, to subsequently, introduce our model. Section B.3 focuses on the application of the methodology using the Iris benchmark and a toy dataset, in addition a example where are tuning the model parameters is presented. Finally, in Section B.4 are presented some conclusions and future extensions of this work.

B.2 Methods

B.2.1 Hierarchical Self-Organizing Structures

The ability of obtaining hierarchically structured knowledge from a dataset using autonomous learning has been widely used in many areas. This is because of the fact that hierarchical self-organizing structures permit to the unevenly distributed real-world data to be represented in a suitable network structure, during an unsupervised training process. These networks capture the unknown data topology in terms of hierarchical relationships and cluster structures.

Different methodologies have been presented in this area with various approaches. Therefore, it is possible to classify hierarchical self-organizing structures in two classes taking into account the algorithm of self-organization used. The first family of models is based on Kohonen self-organizing maps (SOM), and the second on Growing Cell Structures (GCS) [51].

With respect to approaches based on GCS, it has been proposed for instance: Hierarchical Growing Cell Structures (HiGCS) [27], TreeGCS [63] and the Hierarchical topological clustering (TreeGNG) [43]. The algorithms derived on GCS are based on periodic node deletion based on node activity and on the volume of the input space classified by the node. This approach tends to represent mainly the examples with high occurrence rates, and therefore takes as outliers or noise low frequency examples. As a result, examples with low presence rates are not represented in the model. However, in many cases it is desirable to discover novelties in the dataset, so taking into account the observations with low occurrence rates could permit to find out those exceptional behaviors.

For the aforementioned reason, we focused our research in approaches based on SOM [77, 86, 84], particularly in the Growing Hierarchical Self-Organizing Map (GHSOM)[98] due to its ability to take into account the observations with low presence rates as part of the model. This is possible since the hierarchical structure of the GHSOM is adapted according to the requirements of the input space. Therefore, areas in the input space that require more units for appropriate data representation create deeper branches than others, this process is done without eliminating nodes that represent examples with low occurrence rates.

B.2.2 Fuzzy Kohonen Clustering Networks

FKCN [22] integrate the idea of fuzzy membership from Fuzzy c-Means (FCM) with the updating rules from SOM. Thus, creating a self-organizing algorithm that automatically adjust the size of the updated neighborhood during a learning process, which usually, terminates when the FCM objective function is minimized. The update rule for the FKCN algorithm can be given as:

$$W_{i,t} = W_{i,t-1} + \alpha_{ik,t}(Z_k - W_{i,t-1});\ for\ k = 1, 2, ..., n;\ for\ i = 1, 2, ..., c \tag{B.1}$$

where $W_{i,t}$ represents the centroid[2] of the i^{th} cluster at iteration t , Z_k is the k_{th} vector example from the dataset and α_{ik} is the only parameter of the algorithm and according to [65]:

$$\alpha_{ik,t} = (U_{ik,t})^{m(t)} \tag{B.2}$$

Where $m(t)$ is an exponent like the fuzzification index in FCM and $U_{ik,t}$ is the membership value of the compound Z_k to be part of cluster i. Both of these constants vary at each iteration t according to:

$$U_{ik} = \left(\sum_{j=1}^{c} \left(\frac{\|Z_k - W_i\|}{\|Z_k - W_j\|} \right)^{2/(m-1)} \right)^{-1} \ ;\ 1 \leq k \leq n\ ;\ 1 \leq i \leq c \tag{B.3}$$

$$m(t) = m0 - m\Delta \cdot t \quad ; \quad m\Delta = (m0 - m_f)/iterate\ limit \tag{B.4}$$

Where $m0$ is a constant value greater than the final value (m_f) of the fuzzification parameter m. The final value m_f should not be less than 1.1, in order to avoid the divide by zero error in equation (B.3). The iterative process will stop if $\left\| W_{i,(t)} - W_{(i,t-1)} \right\|^2 < \epsilon$, where ϵ is a termination criterion or after a given number of iterations. At the end of the process, a matrix U is obtained, where U_{ik} is the degree of membership of the Z_k element of the dataset to the cluster i. In addition, the centroid of each cluster will form the matrix W where W_i is the centroid of the i^{th} cluster. The FKCN algorithm is given below:

[2] In the perspective of neural networks it represents a neuron or a prototype vector. So the number of neurons or prototype vectors will be equal to the number of clusters.

1. Fix c, and $\epsilon > 0$ to some small positive constant.
2. Initialize $W_0 = (W_{1,0}, W_{2,0}, \cdots, W_{c,0}) \in \Re^c$.
   Choose $m_0 > 1$ and $t_{max} = max.\ number\ of\ iterations$.
3. For $t = 1, 2, \cdots, t_{max}$
   a. Compute all cn learning rates $\alpha_{ik,t}$ with equations (B.2) and (B.3).
   b. Update all c weight vectors $W_{i,t}$ with
   $W_{i,t} = W_{i,t-1} + [\sum_{k=1}^{n} \alpha_{ik,t}(Z_k - W_{i,t-1})] \ / \sum_{j=1}^{n} \alpha_{ij,t}$
   c. Compute $E_t = \left\| W_{i,(t)} - W_{(i,t-1)} \right\|^2 = \sum_{i=1}^{c} \left\| W_{i,(t)} - W_{(i,t-1)} \right\|^2$
   d. If $E_t < \epsilon$ stop.

B.2.3 Fuzzy Growing Hierarchical Self-Organizing Networks

Fuzzy Growing Hierarchical Self-Organizing Networks (FGHSON) are based on a hierarchical fuzzy structure of multiple layers, where each layer consists of several independent growing FKCNs. This structure can grow by means of an unsupervised self-organizing process in two manners (inspired by [98]):

a. Individually, in order to find the more suitable number of prototypes (which compose a FKCN) that may represent in an accurate manner the input dataset.

b. Or on groups of FKCNs in a hierarchical mode, permitting to the hierarchy to reveal a particular set of characteristics of data.

Both growing processes are modulated by three parameters that regulate the so-called breadth (growth of the layers), depth (hierarchical growth) and membership degree of data to the prototype vectors.

The FGHSON works as follows:

1) Initial Setup and Global Network Control:

The main motivation of the FGHSON algorithm is to properly represent a given dataset. The quality of this representation is measured in terms of the difference among a prototype vector and the example vectors represented by this. The *quantization error qe* is used to reach this aim. The *qe* measures the dissimilarity of all input data mapped onto a particular prototype vector, hence it can be used to guide a growth process with the purpose of achieving an accurate representation of the dataset reducing the *qe*. The *qe* of a prototype vector W_i is calculated according to (B.5) as the mean Euclidean distance between its prototype and the input vectors Z_c that are part of the set of vectors C_i mapped onto this prototype.

$$qe_i = \sum_{Z_c \in Ci} \|W_i - Z_c\| \; ; \; C_i \neq \phi \tag{B.5}$$

The first step of the algorithm is focused on obtaining a global measure of the distribution of the whole dataset. For this purpose the training process begins with the computation of a global measure error qe_0. qe_0 represents the qe of the single prototype vector W_0 that form the layer 0, see Figure B.1(a), calculated as shown in (B.6). Where, Z_k represents the input vectors from the whole data set Z and W_0 is defined as a prototype vector $W_0 = [\mu_{0_1}, \mu_{0_2}, \ldots, \mu_{0_n}]$, where μ_{0_i} for $i = 1, 2, \ldots, n$; is computed as the average of μ_{0_i} in the complete input dataset, in other words W_0 is a vector that corresponds to the mean of the input variables.

$$qe_0 = \sum_{Z_k \in Z} \|W_0 - Z_k\| \tag{B.6}$$

The value of qe_0 will help to measure the minimum quality of data representation of the prototype vectors in the subsequent layers, therefore the next prototypes have the task of reducing the global representation error qe_0.

2) Breadth growth process:

The construction of the first layer starts after the calculation of qe_0. This first layer consists of a FKCN ($FKCN_1$) with two initial prototype vectors. Hence, the growth process of $FKCN_1$ begins by adding a new prototype vector and training it until a suitable representation of the dataset is reached. Each of these prototype vectors is an n-dimensional vector W_i (with the same dimensionality as the input patterns), which is initialized with random values. $FKCN_1$ is trained as shown in section B.2.2, taking as input (in the exceptional case of the first layer) the whole dataset. More precisely, $FKCN_1$ is allowed to grow until the qe present on the prototype of its preceding layer (qe_0 in the case of layer 1) is reduced to at least a fixed percentage τ_1. Continuing with the creation of the first layer, the number of prototypes in the $FKCN_1$ will be adapted. To achieve this, the *mean quantization error of the map* (MQE) is computed according to expression (B.7), where d refers to the number of prototype vectors contained in the FKCN, and qe_i represents the quantization error of the prototype W_i.

$$MQE_m = \frac{1}{d} \cdot \sum_i qe_i \tag{B.7}$$

The MQE is evaluated using (B.8) in an attempt to measure the quality of data representation, and is used also as stopping criterion for the growing process of the

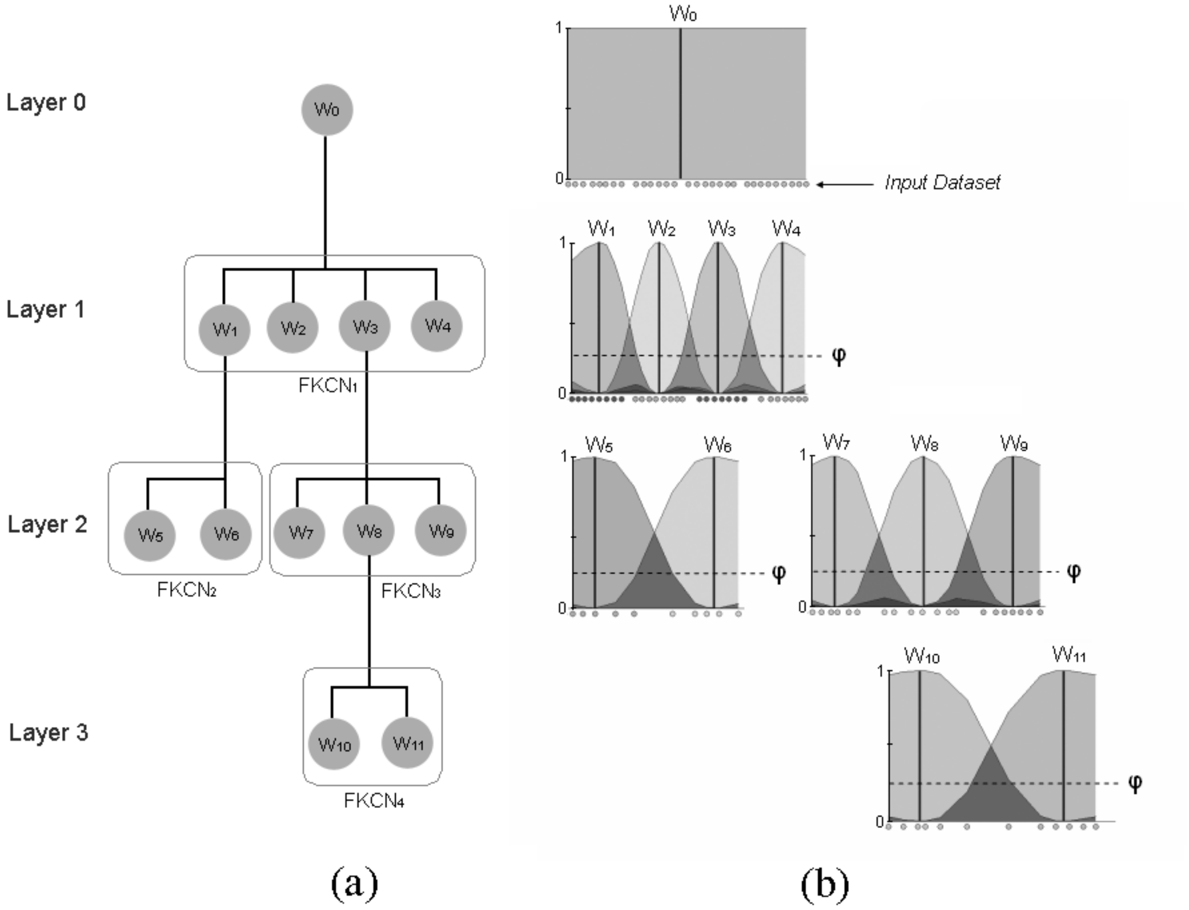


Figure B.1: *(a) Hierarchical structure showing the prototype vectors and FKCNs created in each layer for an supposed case. (b) Membership degrees in each layer, corresponding to the network shown in the diagram on the right side. The parameter φ (the well known $\alpha - cut$) represents the minimal degree membership of an observation to be part of the dataset represented by a prototype vector, the group of data with a desired membership to a prototype will be used for the training of a new FCKN in the next layer (depth process). In this particular diagram the dataset is unidimensional (represented by the little circles below the membership plot) in order to simplify the example*

FKCN. In (B.8) qe_u represents the qe of the corresponding prototype u in the upper layer. In the specific case of the first-layer, the stopping criterion is shown in (B.9).

$$MQE < \tau_1 \cdot qe_u \tag{B.8}$$

$$MQE_{layer1} < \tau_1 \cdot qe_0 \tag{B.9}$$

If the stopping criterion (B.8) is not fulfilled, it is necessary to aggregate more prototypes for a more accurate representation. For this aim, the prototype with the highest qe is selected and is denoted as the error prototype e. A new prototype is inserted in the place where e was computed. After the insertion, all the FKCN parameters are reset to the initial values (except for the values of the prototype vectors) and the training begins according to the standard training process of FKCN. Note that the same value of the parameter τ_1 is used in each layer of the FGHSON. Thus, at the end of the process, a layer 1 is obtained with a $FKCN_1$ formed by a set of prototype vectors W, see Figure B.1(a). In addition, a membership matrix U is obtained. This matrix contains the membership degree of the dataset elements to the prototype vectors, as explained in section B.2.2.

3) Depth growth process:

As soon as the breadth process of the first layer is finished, its prototypes are examined for further growth (depth growth or hierarchical growth). In particular, those prototypes with a large quantization error will indicate us which clusters need a better representation by means of new FKCNs. The new FKCNs thus form a second layer, for instance W_1 and W_3 Figure B.1(a). The selection of these prototypes is regulated by qe_0 (calculated previously in step 1) and a parameter τ_2 which is used to describe the desired level of granularity in the data representation. More precisely, each prototype W_i in the first layer that does not fulfill the criterion given in expression (B.10) will be subject to hierarchical expansion.

$$qe_i < \tau_2 \cdot qe_0 \tag{B.10}$$

After the expansion process and creation of the new FKCNs, the breadth process described in stage 2 begins with the newly established FKCNs, for instance, $FKCN_2$ and $FKCN_3$ Figure B.1(a). The methodology for adding new prototypes, as well as the termination criterion of the breadth process is essentially the same as used in the first layer. The difference among the training processes of the FKCNs in second the layer and the subsequent layers in comparison with the first, is that only a fraction of the whole input data is selected for training. This portion of data will be selected according to a minimal membership degree (φ). This parameter φ (the well known $\alpha - cut$) represents the minimal degree membership of an observation to be part of the dataset represented by a prototype vector. Hence, φ is used as selection parameter, so all the observations represented by W_i have to fulfill expression (B.11), where U_{ik} is the degree of membership of the Z_k element of the dataset to the cluster i. As an example, Figure B.1(b) shows the membership functions of the FKCNs in each layer, and how φ is used as a selection criteria to divide the dataset.

$$\varphi < U_{ik} \tag{B.11}$$

At the end of the creation of layer two, the same procedure described in step 2 is applied in order to build the layer 3 and so forth.
The training process of the FGHSON is terminated when no more prototypes require further expansion. Note that this training process does not necessarily lead to a balanced hierarchy, i.e. a hierarchy with equal depth in each branch. Rather, the specific distribution of the input data is modeled by a hierarchical structure, where some clusters require deeper branching that others.

B.3 Experimental testing

B.3.1 Iris Data Set

In this experiment the Iris dataset [3] is used in order to show the adaptation of the FGHSON to those areas where an absolute membership to a single prototype is not obvious. Therefore, FGHSON must (in an unsupervised manner) look at the representation of the dataset on the areas where observations of the same category share similar zones. For instance in the middle of the data cloud formed by the Virginica and Versicolor observations, see Figure B.2(a).

The parameters of the algorithm were set to τ_1 = 0.2, τ_2 = 0.03, and φ = 0.2. After training, a structure of four layers was obtained. The zero layer is used to measure the whole deviation of dataset as was presented in section B.2.3. The first layer consist of a FKCN with three prototype vectors as shown in Figure B.2(b), this distribution of prototypes attempt to represent three Iris categories. The second layer as shows Figure B.2(c) reach a more fine-grain description of the dataset, placing prototypes in almost all the data distribution, adding prototypes in the zones where more representation was needed. Finally, Figure B.2(d) shows the over population of prototypes in the middle of the cloud of the observations of Virginica and Versicolor . This occurs because this part of the dataset present observations with ambiguous membership in the previous layer, then in this new layer, several prototypes are placed in this zone for a proper representation. Hence, permitting to those observations to obtain a higher

[3] There are three categories in the data set : Iris Setosa, Iris Versicolor and Iris Virginical. Each having 50 observations with four features: sepal length (SL), sepal width (SW), petal length (PL), and petal width (PW).

membership to its new prototypes. Finally, to obtain at the end of the process a more accurate representation of this zone.

B.3.2 Toy set

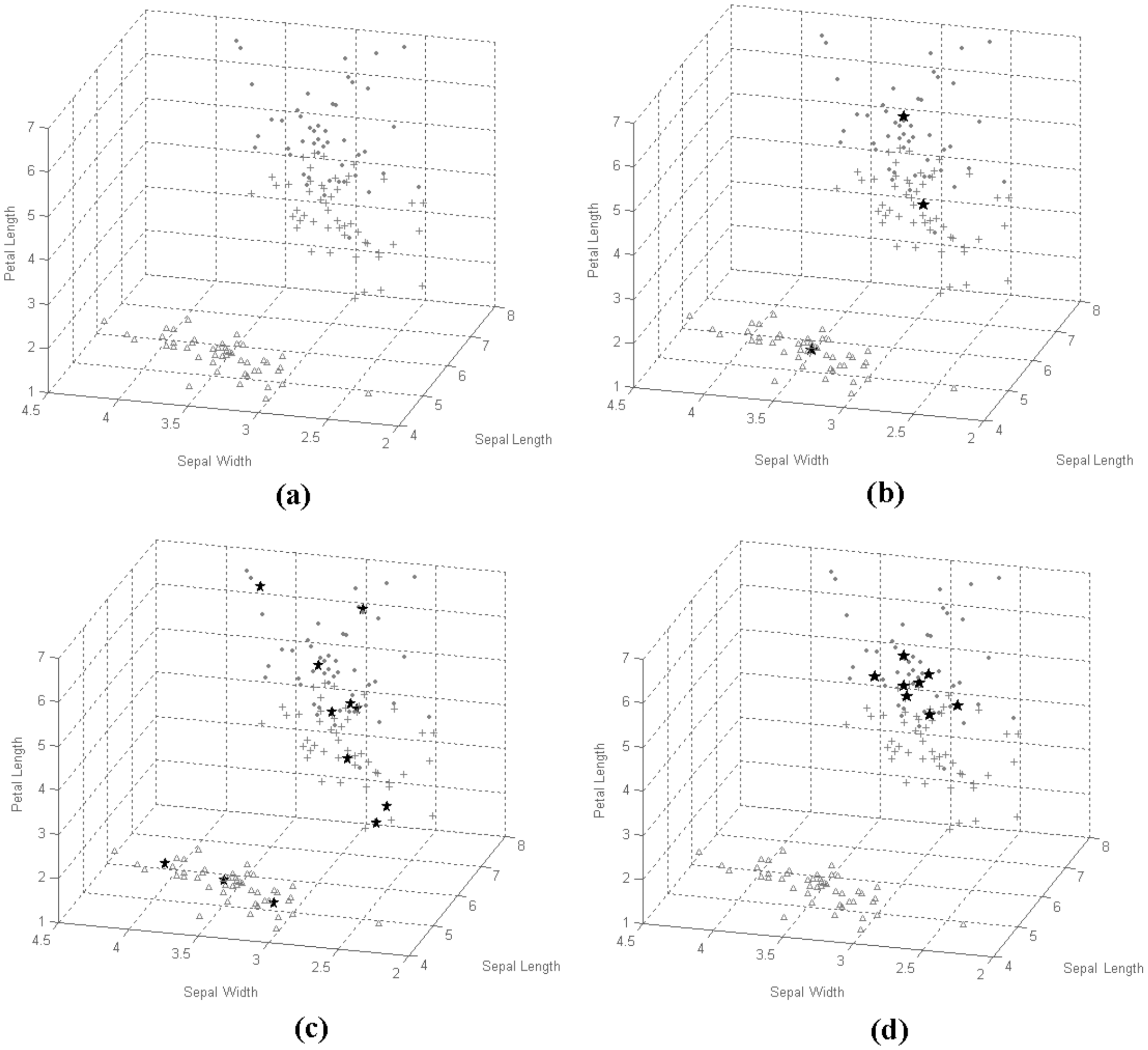
A toy set, as the one presented by Martinez et al [82] is used in order to show the capabilities of the FGHSON to represent a dataset that has multiple dimensionalities. In addition, it is possible to illustrate how the model stops the growing process in those parts where the desired representation is reached and keep on growing where a low membership or poor representation is present. The parameters of the algorithm were set to $\tau_1 = 0.3$, $\tau_2 = 0.065$, and $\varphi = 0.2$ for φ. Four layers were created after training the network. Figure B.3shows: (a) the fist layer. At this level, seven prototypes were necessary to represent the dataset: one for the oval, one for the plane and five for the 3D parallelepiped (note that there are no prototypes clearly associated to the line).

In the second layer shown in the Figure B.3, a more accurate distribution of prototypes is reached, so it is possible to observe prototypes adopting the form of the dataset. Additionally, in regions where the quantization error was large, the new prototypes allow a better representation (e.g., along the line). In layer three (see Figure B.3(c)), no more prototypes are needed to represent the circle, the line and the plane; but a third hierarchical expansion was necessary to represent the parallelepiped. In addition, due to the data density in the parallelepiped, many points share memberships to different prototypes, so a several prototypes at this level were created.

B.3.3 Tuning the model parameters

In order to explore the performance of the algorithm, different values for the parameters φ, τ_1 and τ_2 were tested by using the Iris dataset. The tested were performed by using ten different values of τ_1 (breadth parameter), ten of τ_2 (depth parameter) and eighth of φ ($\alpha - cut$), hence forming 800 triplets. For each triplet (φ, τ_1 and τ_2) a FGHSON was trained using the following fix parameters: $t_{max} = 100$ (maximal number of iterations), $\epsilon = 0.0001$ (termination criterion) and $m_0 = 2$ (fuzzification parameter).

Several variables were obtained in order to measure the quality of the networks created for every FGHSON generated; for instance the number of hierarchical levels of the obtained network, the number of FKCNs created for level, and finally the quantization error by prototype by level. The analysis of these values will allow to find the relationships between the triplet of parameters (φ, τ_1 and τ_2) and the topology of the networks (represented in this experiment for the levels reached for the network and the number of FKCN created). In addition, it will be possible to observe the relationship among the quantization errors of prototype by level and the parameters of the algorithm. Finally, being able to find the values of the parameters that allow to build the most accurate structure, based on the number of prototypes, the quantization error and the number



(a)

(b)

(c)

(d)

Figure B.2: *Distribution of the prototype vectors, represented by stars, in each layer of the hierarchy. (a) Iris data set. There are three Iris categories: Setosa, Versicolor, and Virginica represented respectively by triangles, plus symbols, and dots. Each having 50 samples with 4 features. Here, only three features are used: PL, SW, and SL. (b) First layer (c) Second layer and (d) Third layer of the FGHSON, in this layer prototypes are presented only in the zone where observations of Virginica and Vesicolor share the same area, so the new prototypes represent each category in a more accurate manner.*

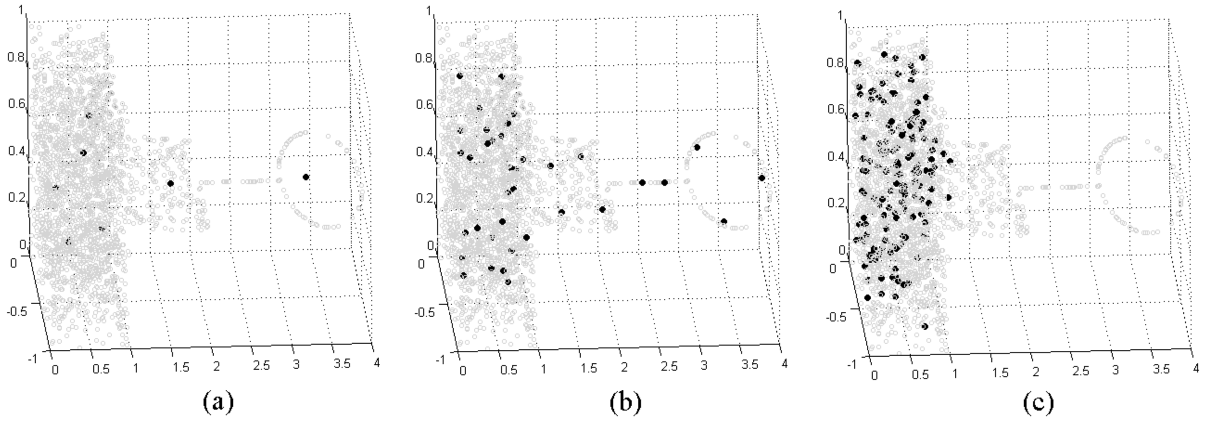


Figure B.3: *Distribution of the prototype vectors (represented by black points) (a) First layer (b) Second layer (c) Third layer*

of levels present in the network.

Due the large amount of information to analyze, it was used a graphical representation of the obtained data in order to facilitate the visualization of the results. For this aim 3D plots were used as follows: the parameter τ_1 (which regulates the breadth of the networks) and the parameter τ_2 (which regulates the depth of the hierarchical architecture) are shown on the x-axis and y-axis respectively. The z-axis is the quantity of levels of the hierarchy (see Figure B.4 and Figure B.5). Each 3D plot corresponds to one fix value of φ. Hence, eight 3D plots represent the eight different values evaluates for φ, then each 3D plot contains the 100 possible combinations of the duple (τ_1,τ_2) for the specific φ. Therefore, 800 networks were generated and plotted making possible analyze τ_1, τ_2, φ, and the levels of the obtained networks.

Furthermore, additional information was added to the 3D plots. On the one hand, the number of FKCNs created for level were represented by a symbol in the 3D plots (see Figure B.4 and Figure B.5 left side). On the other hand, the higher quantization error of the prototypes that were expanded is shown in new group of 3D plots; in others words this prototype is the "father" with the higher quantization error of the prototypes in that level[4]. The round value of the selected quantization error was allocated as mark in the 3D plot for each triplet of values for each level (see Figure B.4 and Figure B.5 right side).

Examining the obtained results, it is possible to find some interesting results related with the quantization error and the topology of the network. For instance Figure B.4

[4] For this reason in level one all the values are 291 (see Figure B.4 and Figure B.5 right side) because the prototype "father" that is expanded have the same quantization error for all the networks; in the case of the first level this error is called qe_0, as is described in section B.2.3.

and Figure B.5 show the different networks created, so it is possible to observe that for values with τ_2 above 0.3 the model generates networks with just one level, so an interesting area to explore lies between the values of τ_2 = 0.1, 0.2 and 0.3. Respect to the quantization error right side) it is possible to observe that in almost all the values of φ, the lower quantization error with the lower number of levels was presented in the points (τ_1 = 0.1, τ_2 = 0.2) and(or) (τ_1 = 0.2, τ_2 = 0.1), see (Figure B.4 and Figure B.5.

In the next step of the analysis, the value of the parameters which generated the best networks so far were selected, based on the premise that the most accurate network has to present the lower number of levels, number of FKCNs, and the lower quantization error. For a selected group of three networks (see table B.1), the distribution of the prototypes on the dataset were plotted; in order to analyze how the prototypes of these selected networks have been adapted to the dataset (see Figure B.6).

Some remarks could be made about the plots obtained. In the first example (φ = 0.1, τ_1 = 0.2, τ_2 = 0.1) (Figure B.6(a)), it is possible to observe four prototypes in the first level of the hierarchy; these prototypes represent four classes[5]. Then, the prototypes of this layer intents to represent the three classes of iris, and in addition they also take the problematic region in between Versicolor and Virginica as a fourth class. Furthermore, new prototypes are created in the second layer in order to obtain a more accurate representation of the dataset, so creating a proliferation of prototypes; this phenomena is due to the low value of φ (0.1). This is because the quantity of elements represented for each prototype is big (because of the membership is low, a lot of data can have membership to one prototype) so it is necessary many prototypes to reach a low quantization error.

In the next example tunning with the values φ = 0.3, τ_1 = 0.2, τ_2 = 0.1, it is possible observe in Figure B.6(b) the three prototypes created in the first level. In this case the number of the prototypes matches with the number of classes of the iris dataset. There is (as in the previous example) an abundance of prototypes in the Virginica-Versicolor group. But in this case the number of prototypes is lower compared with the preceding example, so showing how the φ affect the quantity of prototypes created.

Finally, in the last example tunning with the values φ = 0.6, τ_1 = 0.2, τ_2 = 0.1. The Figure B.6 shows: (c) three prototypes created in the first level of this network matching with the classes of the Iris dataset; additionally, in the second layer one of the previous prototypes is expanded to three prototypes more in order to represent the fuzzy areas of the data set. This last network presents the lower values of vector quantization, levels of hierarchy, and FKCNs; so it is possible to select this as the more accurate topology consider the previously defined premise which said that the most accurate network had to present the lower number of levels, number of FKCNs, and the lower quantization error.

[5] It is knowing that they are three classes (iris Setosa, Virginica and Versicolor) but that exists an area where Versicolor and Virginica present similar characteristics

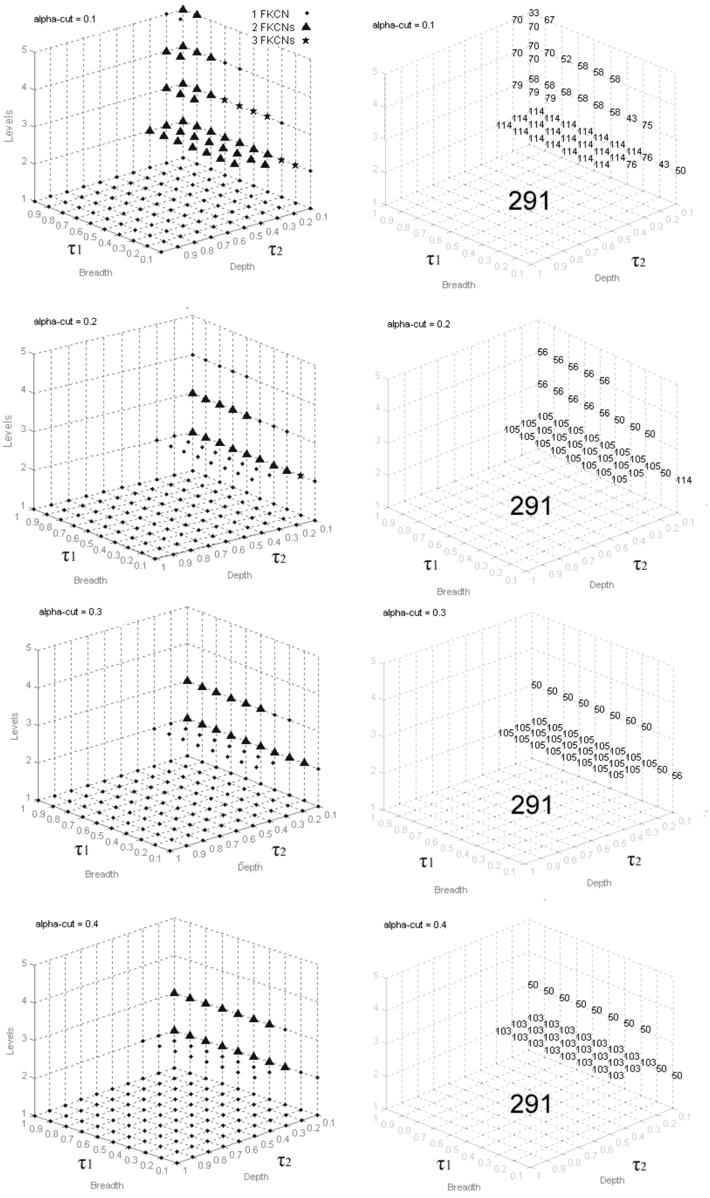


__Figure B.4:__ The figure has 3D plots showing the results obtained using φ = 0.1, 0.2, 0,3 and 0.4. In the left side it is possible observe the levels obtained for each triplet (φ, τ_1, τ_2), in addition the number of FKCN created for level were represented by a symbol. In the right side it is shown the higher quantization error of the prototypes that were expanded ; in others words this prototype is the "father" with the higher quantization error of the prototypes in that level. In the special case of the first level all the values are 291, because the prototype "father" that is expanded have the same quantization error for all the networks; in the case of the first level this error is called qe_0, as is described in section B.2.3.

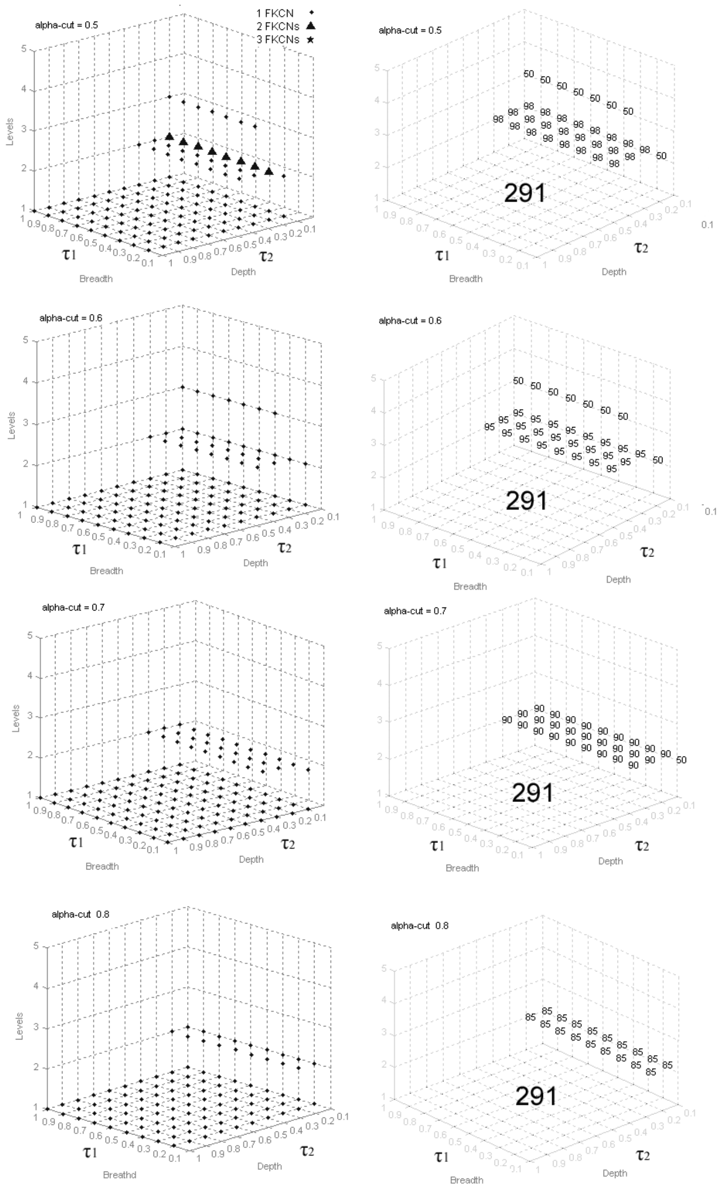
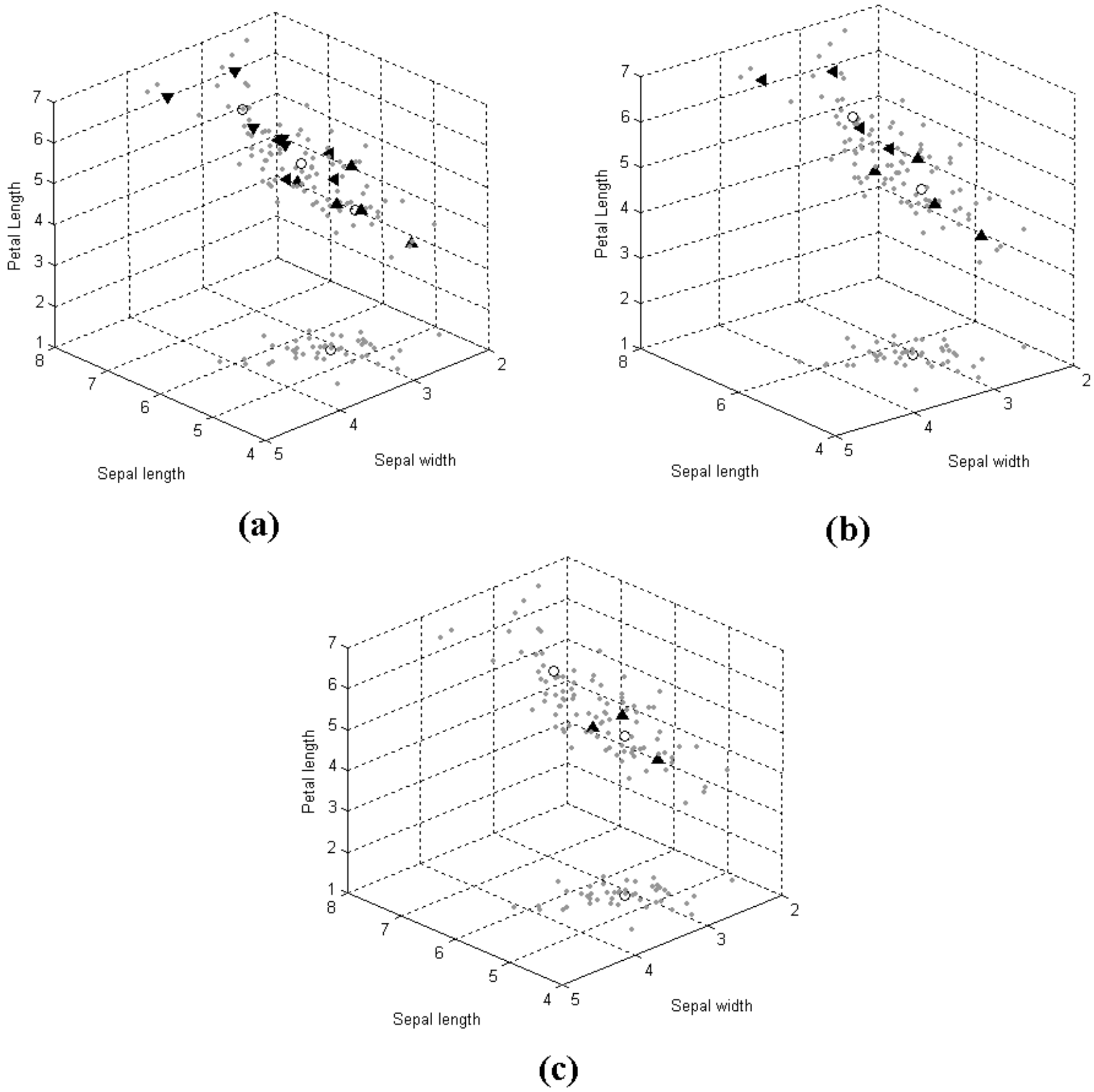


Figure B.5: *The Figure shows the results obtained using the values: φ = 0.5, 0.6, 0.7 and 0.8, in addition the number of FKCN created for level were represented by a symbol. In the right side it is shown the higher quantization error of the prototypes that were expanded; in others words this prototype is the "father" with the higher quantization error of the prototypes in that level.*



(a) (b) (c)

__Figure B.6:__ Structures obtaining tunning the model with the values (a) φ = 0.1, τ_1 = 0.2, τ_2 = 0.1 (b) φ = 0.3, τ_1 = 0.2, τ_2 = 0.1, and (c) φ = 0.3, τ_1 = 0.2, τ_2 = 0.1. In this figure it is possible to observe the distribution of the prototype vectors; the prototypes of the first level are represented by circles and the prototypes of the second level are represented by triangles.

Table B.1: *Parameters and results of the best networks selected*

φ	τ_1	τ_2	Levels	qe[a]	FKCNs
0.1	0.2	0.1	2	43	3
0.3	0.2	0.1	2	50	2
0.6	0.2	0.1	2	50	1

[a] The higher quantization error of the prototype "father" of the prototypes in that level.

B.4 Conclusion

The Fuzzy Growing Hierarchical Self-organizing Networks are fully adaptive networks able to hierarchically represent complex datasets. They allow to clustering data in a a fuzzy way, allocating observations with a similar membership degree into several clusters. This property ais to better describing the structure of the dataset and the inner data relationships.

In this book chapter has been presented the effects of use different values for the parameters of the algorithm, using as example the Iris dataset. It was shown how the different parameters affect the topology and quantization error of the networks created. In addition, some of the better networks created were examined in order to show how different representations of the same dataset can be obtained with similar accuracy.

Acknowledgments This work is part of a cooperation project between BIOTEC, CIAT, CENICAÑA (Colombia) and HEIG-VD (Switzerland) named Precision Agriculture and the Construction of Field-crop Models for Tropical Fruits. The economical support is given by several institutions in Colombia (MADR, COLCIENCIAS, ACCI) and the State Secretariat for Education and Research (SER) in Switzerland.

Appendix C

Third article

Paper published on the journal *Computers and Electronics and Agriculture*, volume 69, issue 2, December 2009.

Analysis of Andean blackberry (Rubus glaucus) production models obtained by means of artificial neural networks exploiting information collected by small-scale growers in Colombia and publicly available meteorological data

Daniel Jiménez[1,3,4], James Cock[4,5], Héctor F. Satizábal[2,3,4], Miguel A. Barreto S[2,3,4], Andres Perez-Uribe[3], Andy Jarvis[5,6], Patrick Van Damme[1]

1. Ghent University, Faculty of BioScience Engineering: Agricultural Science, Laboratory of Tropical and Subtropical Agronomy and Ethnobotany, Coupure links 653-9000, Ghent, Belgium

2. Université de Lausanne, Hautes Etudes Commerciales (HEC), Institut des Systèmes d'Information (ISI), Switzerland

3. REDS Institute, University of Applied Sciences of Western Switzerland (HEIG-VD)

4. BIOTEC, Precision Agriculture and the Construction of Field-Crop Models for Tropical Fruit Species, Colombia

5. International Center for Tropical Agriculture (CIAT), Decision and Policy Analysis (DAPA), Colombia

6. Bioversity International, Colombia

Abstract The Andean blackberry (*Rubus glaucus*), is an important source of income in hillside regions of Colombia. However, growers have little reliable information on the factors that affect the development and yield of the crop, and therefore there is a dearth of information on how to effectively manage the crop. Site specific information recorded by small scale producers of the Andean blackberry on their production systems and soils coupled with publicly available meteorological data was used to develop models of such production systems. Multilayer perceptrons and self-organizing maps were used as computational models in the identification and visualization of the most important variables for modeling the production of Andean blackberry. Artificial neural networks were trained with information from 20 sites in Colombia where the Andean blackberry is cultivated. Multilayer perceptrons predicted with a reasonable degree of accuracy the production response of the crop. Both the soil depth and the average temperature of the first month before harvest were critical determinants of productivity. A proxy variable of location was used to describe overall differences in management between farmers groups. The use of this proxy indicated that, even under essentially similar environmental conditions, large differences in production could be assigned to management effects. The information obtained can be used to determine sites that are suitable for Andean blackberry production, and transfer of management practices from sites of high productivity can be transferred to sites with similar environmental conditions to increase productivity.

Keywords: Andean blackberry, small-scale growers, artificial neural networks, multilayer perceptron, self-organizing maps, input relevance analysis, publicly available meteorological data.

C.1 Introduction

The Andean blackberry (*Rubus glaucus* Benth.), also known as the Andes Berry or Mora de Castilla [4] is a fruit native to an area ranging from the northern Andes to the southern highlands of Mexico [120]. It is grown as a commercial crop in Colombia, Ecuador, Guatemala, Honduras, México and Panamá [50]. It is an important source of income in hillside regions of Colombia [38]. Productivity varies widely between regions and also between farms. Furthermore, the crop is harvested continuously during the year and the productivity varies throughout the year. At the same time growers have little reliable information on the factors that effect the development and yield of the crop, and consequently there is a dearth of readily available information on where to grow the crop and how to effectively manage it.

Research on the Andean blackberry is limited and with the current levels of research intensity it is unlikely that technological packages can be developed for use by growers based on traditional plot based experimentation varying individual factors that affect crop production. The heterogeneous growing conditions and the continuous production throughout the year of many tropical crops mean that a large number of experiments or treatments required to draw firm conclusions concerning the optimum management of the crop under diverse conditions. The situation of a tropical crop such as the Andean blackberry contrasts strongly with that of, let us say, raspberries in a temperate climate. In the case of most temperate crops, there is a relatively short and well defined harvest period and all management is geared to optimal production in that period. In tropical perennial crops that are harvested throughout the year, the number of possible combinations of management practices that need to be tested are enormous. Thus, for example Andean blackberry production during the dry season may require totally different water and pest management practices to those required for the same crop in the wet season. A direct consequence of these multiple management options is continual experimentation by producers of crops like Andean blackberries. Every time a farmer harvests his crop, there is a unique event, an unreplicated experiment [35]. Experience with sugarcane, which is also a perennial tropical crop that may be harvested throughout the year in the low latitude tropics, has shown that by collecting information on crop production produced with the naturally occurring variation in management and the environment, the crops response can be modeled using statistical or best fit models [66]. This approach has later been successfully applied to another perennial tropical crops, like coffee [92]. Given the scarce available information and the limited resources for field work research, and the high degree of heterogeneity in both growth and management, we opted for a data-driven modeling approach to provide information to growers on how to choose apposite sites for and to better manage their crops.

Crop models are basically of two types which can roughly be describe as mechanistic simulation models and best fit or statistical models. The mechanistic models have the great advantage, at least in theory, that they can be extrapolated out of the range of variation for which data exists as they are based on the basic physiological functions of the plant and their response to variation in individual parameters in the environment. Furthermore, variables that affect the observed variation in crop response to changes in the environment can be identified in causal relationships. However, these mechanistic simulation models require detailed knowledge of the functional relationships between the multiple physiological and other processes involved in crop growth and development. This knowledge base simply does not exist, and would take years to develop, for a crop like the Andean blackberry that has received little attention from researchers in the past. Statistical or best fit models are generally simpler and rely upon relationships between variations in observed crop growth and development and variations in the growing conditions. The best fit models, however, have the dual disadvantage that they can neither be used to extrapolate beyond the range of variation encompassed in the initial datasets used to develop the models, and secondly they are not able to determine whether relationships are causal or merely associations. The best fit models do, however, have the advantage that they can be constructed with a limited knowledge of the myriad individual processes and their interaction with variation in the environment that determines how a crop grows, develops and finally produces a useful product. Thus, with insufficient resources to obtain the knowledge required to develop mechanistic models, and the observation that best fit models have successfully been used in other crops, this approach was selected for Andean blackberry.

Many of the best fit models used to predict crop yields are developed using existing information on both crop production and the environment. In the case of small farm crops, such as the Andean blackberry, information on crop production is not readily available and certainly cannot readily be associated with the particular environmental conditions under which a particular crop was harvested. However, as we previously observed, every harvest is effectively an unreplicated experiment. If it were possible to characterize the production system in terms of management and the environmental conditions, and if we were able to collect information on the harvested product of a large number of harvesting events under varied conditions, it should be possible to develop best fit models for the production system. Hence, first step in developing these models was the acquisition of data on Andean blackberry production and the characterization of the production systems.

Agricultural systems are difficult to model due to their complexity and their non-linear dynamic behavior. The evolution of such systems depends on a large number of ill-defined processes that vary in time, that interact with each other, and whose relationships are often highly non-linear and very often unknown [70]. Moreover, the available information describing these systems frequently includes both qualitative and quantitative data, the former often difficult to include in traditional modeling approaches. We surmised that bio-inspired models, such as artificial neural networks, are an appropriate alternative for developing models that can be used to improve production systems.

Artificial neural networks have been successfully used to model agricultural systems [59, 108, 109]. According to Jiménez et al. [70], these techniques are appropriate as an alternative to traditional statistical models and mechanistic models, when the input data is highly variable, noisy, incomplete, imprecise, and of a qualitative nature, as is the case of our Andean blackberry dataset. Artificial neural networks do not require prior assumptions concerning the data distribution or the form of the relationships between inputs and outputs [105, 95, 91]. They are capable of "learning" non-linear models that include both qualitative and quantitative information, and in general, they provide superior pattern recognition capabilities than traditional linear approaches [90, 109, 93]. They have become a powerful technique to extract salient features from complex datasets [32, 54]. Furthermore, when dealing with multiple variables they can be used to produce easily comprehensible low-dimensional maps that improve the visualization of the data, and facilitate data interpretation [16]. Nevertheless, there are a number of disadvantages concerning the use of artificial neural networks, some of them are: its "black box" nature, which makes it difficult to interpret relations between the inputs and outputs, the difficulty of directly including knowledge of a ecological processes, the tendency to overtrain, and the need for enough data to be properly trained [109, 105, 95].

An important first step in developing models that explain variation in yield is the identification of relevant variables that affect yield: identification of these variables guides the data collection required as inputs into the model.

Several studies identify the most relevant variables, and explain given responses in agriculture through the use of multilayer preceptrons. For instance, Miao et al. [85] implemented a neural network for identifying the most important variables for corn yield and quality. Using soil and genetic data, and a sensitivity analysis for each variable, they demonstrated that the hybrid was the most important factor explaining variability of corn quality and yield. In another study, Jain [67] reported that the best frost prediction was obtained from the relative humidity, solar activity and wind speed from 2 to 6 h before the frost event. Paul and Munkvold [95] predicting severity of gray leaf spot of maize (*Cercospora zeae-maydis*) in corn (*Zea mays* L.), concluded that the best variables for predicting severity were hours of daily temperatures, hours of nightly relative humidity, and mean nightly temperature. More recently, Jiménez et al. [71] modeling sugarcane yield, suggested that crop age and water balance were highly relevant for the modeling process.

Self-Organizing Maps (SOM) have also been implemented to improve the visualization of input-onput and input-output dependencies. Thus, for example Moshou et al. [89] found that a waveband centered at 861 nm was the variable which best discriminated healthy from diseased leaves with yellow rust (*Puccinia striiformis* f. sp. *tritici*) in wheat (*Triticum* spp cv. Madrigal). As another example, Boishebert et al. [39] pointed out that growing year was an important factor in the differentiation of yield of strawberry varieties.

Extension agents, expert crop advisers and growers of Andean blackberry have reached a general consensus that optimum conditions for the crop are: soils with high of organic matter content and a loamy texture, altitude between 1800 and 2400 m above sea level, average relative humidity between 70 and 80%, average temperature between 11 and 18 degree Celsius (°C), and 1500 and 2500 mm of rainfall per year [50].

The goal of this research was to demonstrate that collection of data from poor small-scale commercial producers of Andean blackberry and its analysis by means of artificial neural networks can provide growers with useful information to increase their productivity.

C.2 Materials and methods

CorporaciÃşn Biotec together with local Andean blackberry producers developed a simple aid based on a calendar which was used by the farmers to record information on the production of each lot planted to blackberries on their farm. The soil characteristics were determined by the soil and terrain evaluation methodology known as RASTA (Rapid Soil and Terrain Assessment) [10] for 20 different sites in the departments of Nariño and Caldas in Colombia. The information collected by the farmers on the calendars and with RASTA was then transferred to the database of the site-specific agriculture for tropical fruits (AEPS) project. This database includes information on location, landrace varieties, yield, harvest time and data on soil characteristics. A total of 488 records of yield from the database were included in the analysis. These records or "events" provided farmers' estimates of the quantity (in kilograms) of fruit harvested per plant for weekly periods (Fig. C.1).

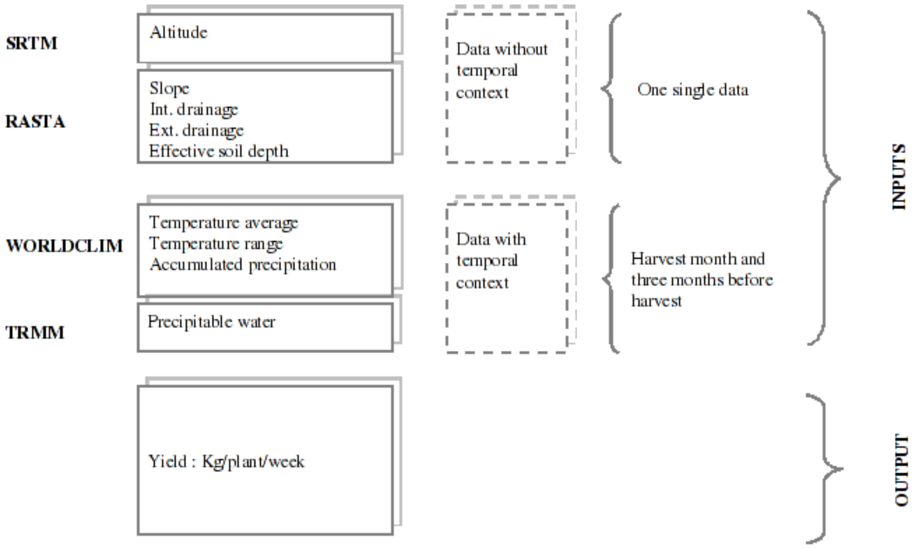


Figure C.1: Variables selected for the construction of the Andean blackberry yield model

Environmental information of landscape and climate was obtained for each site from a range of secondary data sources. Topography and landscape data was extracted from the Shuttle Radar Topography Mission (SRTM) [47], using the Version 3 dataset available from the CSI-CGIAR. Long-term averages for monthly temperature and precipitation were obtained from WORLDCLIM [60], and daily rainfall was extracted from the 3b42 product of the Tropical Rainfall Measuring Mission (TRMM) database [20].

C.2.1 Variable selection

The information compiled in the database for Andean blackberry consisted of 28 variables (Table C.1). This information included categorical variables describing geographical position (large areas for departments, specific areas for particular localities within departments) and variety (thorn blackberry or normal blackberry), and environmental variables based on landscape, soil and climate (Table C.1). Each yield observation was associated with the environmental variables taking into account the date of harvest (Fig. C.1).

Table C.1: *Inputs used for development of Andean blackberry yield model. Cat=Categorical variables. Con=Continuous variables*

Input	Variable	Type	Abbreviation	Source
1	Thorn or Normal blackberry	Cat	AB_Thorn_N	AEPS
2	Nariño-Caldas (Large geographic area)	Cat	Nar-Cal	AEPS
3	Nariño-la union-chical alto (specific geographic area)	Cat	Na_un_chical	AEPS
4	Nariño, la union,cusillo alto (specific geographic area)	Cat	Na_un_cusal	AEPS
5	Nariño, la union, cusillo bajo (specific geographic area)	Cat	Na_un_cusba	AEPS
6	Nariño, la union, la jacoba (specific geographic area)	Cat	Na_un_lajac	AEPS
7	Caldas Riosucio zona rural (specific geographic area)	Cat	Cal_riosu_zr	AEPS
8	Altitude	Con	Srtm	SRTM
9	Slope	Con	Slope	SRTM
10	Internal drainage	Con	IntDrain	AEPS
11	External drainage	Con	ExtDrain	AEPS
12	Effective soil depth	Con	EffDepth	AEPS
13	Precipitable water of the harvest month	Con	Trmm_0	TRMM
14	Precipitable water of the first month before harvest	Con	Trmm_1	TRMM
15	Precipitable water of the second month before harvest	Con	Trmm_2	TRMM
16	Precipitable water of the third month before harvest	Con	Trmm_3	TRMM
17	Average temperature of the harvest month	Con	TempAvg_0	WORLDCLIM
18	Temperature range of the harvest month	Con	TempRang_0	WORLDCLIM
19	Accumulated precipitation of the harvest month	Con	PrecAcc_0	WORLDCLIM
20	Average temperature of the first month before harvest	Con	TempAvg_1	WORLDCLIM
21	Temperature range of the first month before harvest	Con	TempRang_1	WORLDCLIM
22	Accumulated precipitation of the first month before harvest	Con	PrecAcc_1	WORLDCLIM
23	Average temperature of the second month before harvest	Con	TempAvg_2	WORLDCLIM
24	Temperature range of the second month before harvest	Con	TempRang_2	WORLDCLIM
25	Accumulated precipitation of the second month before harvest	Con	PrecAcc_2	WORLDCLIM
26	Average temperature of the third month before harvest	Con	TempAvg_3	WORLDCLIM
27	Temperature range of the third month before harvest	Con	TempRang_3	WORLDCLIM
28	Accumulated precipitation of the third month before harvest	Con	PrecAcc_3	WORLDCLIM

C.2.2 Computational models

C.2.2.1 Multilayer perceptron

A multilayer perceptron [24] was used to model Andean blackberry yield, in such a manner that the output of the neural network, the continuous variable yield, is determined by the 28 variables we used as inputs. The Back-propagation algorithm [24] was employed in order to train the neural networks. This algorithm is a gradient descent based optimizer which minimizes the difference between the desired output of the model (in the training dataset) and the actual output of the network, i.e. the mean square error (MSE).

In order to provide a mechanism for testing the model performance and to compare different models or network topologies, a training and a validation dataset were created by random sampling without replacement from the whole dataset. In this way, each training step was performed using 80% of the whole dataset, and every testing procedure to assess model performance, was performed on the remaining 20%. This method, called “split-sample” or “hold-out” validation, may assess predictive model performance, but is not recommended in its simplest form for small datasets [57]. However, the split sample procedure can be improved for small dataset by repeating the “split-sample” procedure several times, and then calculating the resulting performance as the average of all the tests made over the different validation subsets. Different “flavors”of this method have been used with artificial neural networks [45]. These include “cross-validation”, “leave-one-out validation”, and “bootstrap validation”.

Network topology is an important issue in training a neural network model. The selection of the number of neurons in the hidden layer was made by comparing neural networks having 1,2,3,4,5,6,7,8,9 and 10 hidden units. This comparison was carried out by simple implementation of a bootstrap validation scheme [45]. Thus, each network was tested by performing “split-sample” validations 100 times, and then the different values of the averaged MSE were compared in order to determine the network having the best performance. The topology with the lowest MSE over the validation subset had 5 units in the hidden layer neural network (Fig. C.2) and was selected for further development.

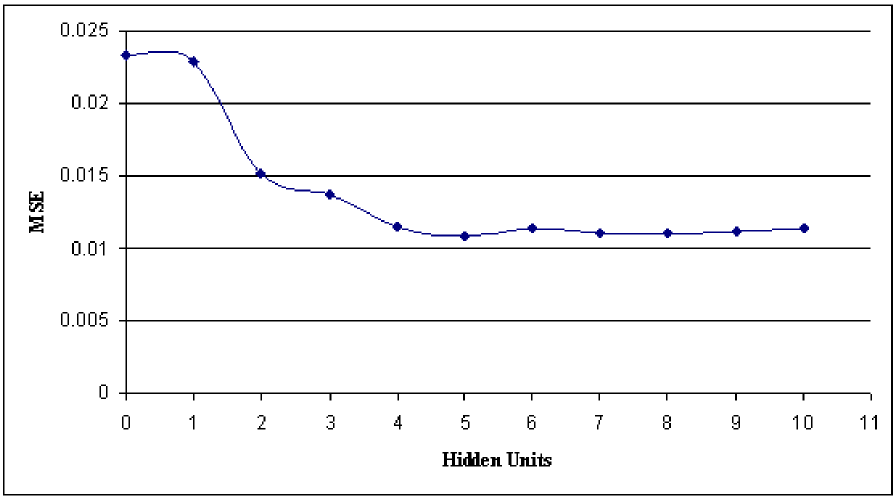


Figure C.2: *MSE of artificial neural networks with different number of neurons in hidden layer*

An ensemble of 100 networks with the selected topology but with different initialization was built and tested in order to improve the generalization capabilities of the model [42, 26]. Neural networks ensembles are less affected by local minima, and have been shown to outperform their single components [133]. In our case, the source of diversity among models was the starting point of the Back-propagation algorithm (random initialization), and the resulting model output was calculated by averaging the outputs of the 100 individual networks.

Finally, to identify the variables which contribute most to yield; an analysis was conducted by means of the relevance metric based on sensitivity described in [107]. This method assesses input relevance by calculating the partial derivative of the output of the neural network ensemble with respect to each one of the inputs. Input sensitivity should reflect input relevance because the Back-propagation algorithm finds higher connection weights to inputs having more relevant information and, in the same way, attenuates connections from noisy inputs.

C.2.2.2 Self-Organizing Maps

The Self-Organizing Map or SOM [75] is a non-supervised algorithm which combines clustering and visualization. The SOM maps high-dimensional datasets can be in a low-dimensional output space (generally a grid of two dimensions) with the SOM technique: observations with similar characteristics appear clustered together in the low-dimensional map produced. Such a map facilitates exploratory visual analysis of the clusters and the relationships between the variables of a complex dataset. However, a SOM does not preserve distance information. In order to address this problem the topology is disregarded, and standard clustering methods are applied to the SOM prototype vectors, and then the clusters are displayed on a lattice [122].

We chose the K-means algorithm to group the observations into a given number of K clusters. One of the limitations of this technique is the a-priori definition of the number of clusters, which is frequently unknown. To tackle this drawback, different K values were tested and then different groups with different number of clusters were calculated. The optimal number of K was then derived using the Davies-Bouldin index [37]. The coordinate axes of the lattice are not clearly interpreted in terms of the original variables. Instead, variables are typically visualized by a "component plane" representation, where several lattices, one for each variable, are shown side by side. A lattice with a variable-specific coloring is called a component plane. The component plane representation is useful in finding dependencies between variables. These dependencies are perceived as similar patterns in identical areas of different component planes (Figure C.7, Figure C.8, Figure C.9, Figure C.10, Figure C.11, Figure C.12 and Figure C.13). The dependency search can be eased by organizing the component planes such that similar planes are positioned near to each other [122]. In the present study, a SOM was used in order to facilitate the visualization of the relations among the productivity and the 28 environmental and geographical variables, and establish the values ranges of these variables associated with high, medium and low yield.

C.3 Results and discussion

C.3.1 Model performance

The neural network model was evaluated to ensure that its performance was acceptable for our purpose of determining relationship between the yield of the Andean blackberry and the characteristics of sites where it was grown. To evaluate the model's performance we computed the coefficient of determination of the real Andean blackberry's yield and the yield predicted by the model using only the data from the "hold-out" validation dataset (Fig. C.3). The coefficient of determination (0.89) indicates that the model explained close to 90% of the total variation, which we considered sufficient to proceed to the next step of determining input relevance.

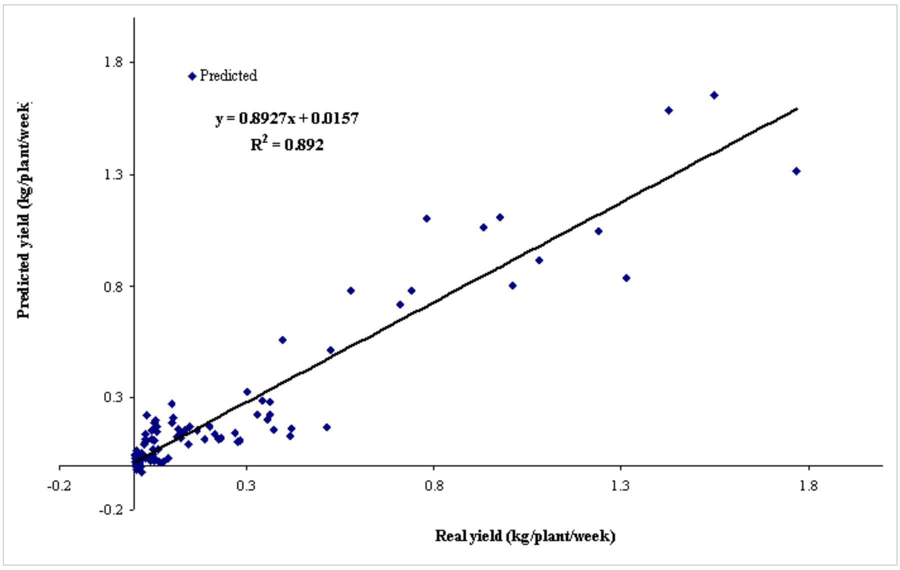


Figure C.3: *Scatterplot displaying multilayer perceptron predicted yield versus real Andean blackberry yield, using only the validation dataset*

The fit between the real yield values and the predicted values taken from the validation data was close at the low levels of production, but was poor over the range of 69-93 (Fig. C.4). At the same time, the model accurately predicted the expected yields at high yield levels. This suggests that the model can be used to determine ex ante the conditions and management associated with high yields and hence to provide farmers with guidelines on how to obtain high yields. In addition, the model can also determine site characteristics that are inevitably associated with poor crop performance and these can be used to indicate to farmers that a particular site and management combination is not a viable option.

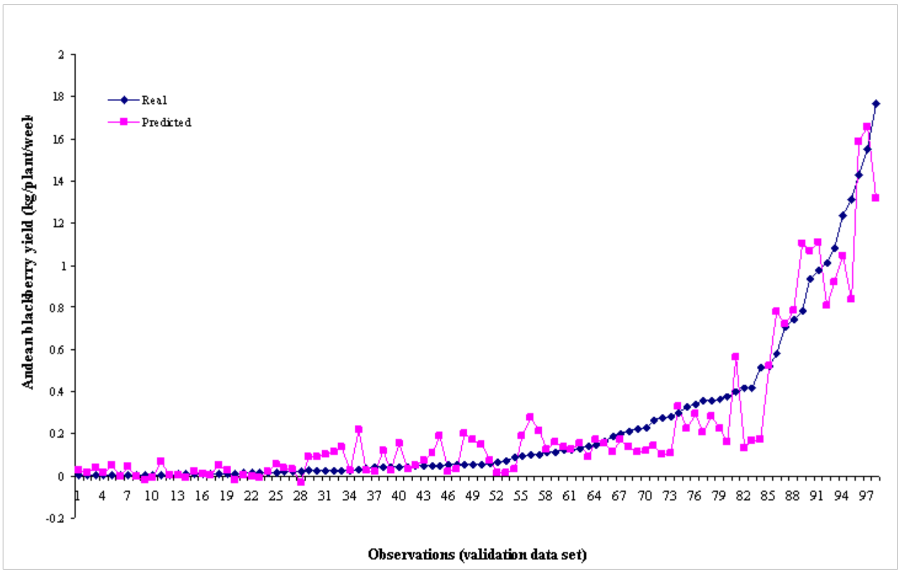


Figure C.4: *Line with markers displaying the fitness of the predicted and real Andean blackberry yield through the observations from the validation dataset (yield values upwardly ordered)*

C.3.2 Analysis of the variables relevance

We assessed the yield response to changes in the 28 variables used in the model by obtaining the sensitivity of the model output with respect to each one of the inputs. We used the relevance metric based on sensitivity described in [107], which expresses the amount of change of the output with the variations of the inputs. The nine most important variables identified by the sensitivity metric were: soil depth, the average temperature of the first month before harvest, the specific geographical areas Nariño-la union-chical alto and Nariño-la union-cusillo bajo, the average temperature of the harvest month, the average temperature of the second month before harvest, the average temperature of the third month before harvest, external drainage and the accumulated precipitation of the first month before harvest (Figure 5). There was a moderately sharp drop off of the sensitivity after the ninth variable (see Figure C.5). A Wilcoxon test at an alpha level of 5% (Table C.2) indicated that the means of this group of nine variables were significantly different ($p = 0.0001$) from the rest of the variables. Hence, the nine most important variables were selected for further analysis.

Table C.2: *Wilcoxon test at an alpha level of 5% comparing means of relevance between the nine most important variables identified by the sensitivity metric and means of the rest of variables*

T	T(expected value)	T(variance)	Z(observed value)	Z(critical value)	Two-tailed p-value
171.000	85.500	527.250	3.724	1.960	0.0001

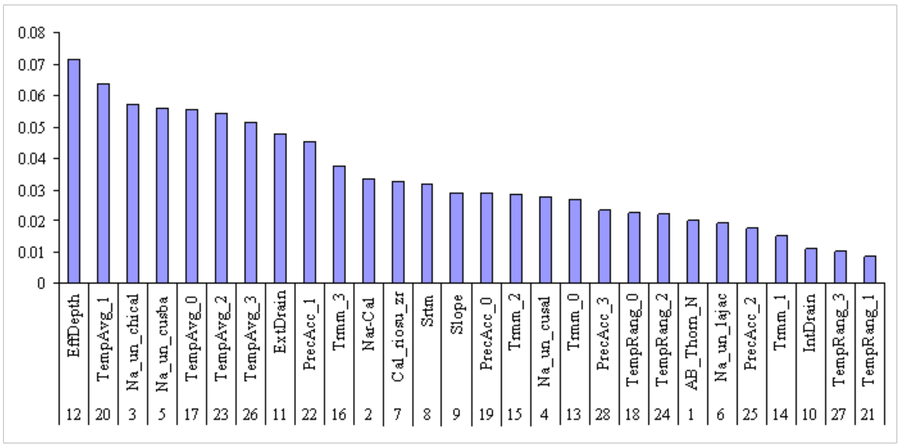


Figure C.5: *Sensitivity distribution of the model with respect to the inputs*

C.3.3 Visualization of the relations between the variables found as relevant by the sensitivity metric and clusters with similar productivity of Andean blackberry

To further analyze the effects of the nine variables, a Kohonen map was trained with the same observations we employed to train the multilayer perceptron. The resulting bidimensional map is composed of vector prototypes which associate topological information of the original 28 variables with Andean blackberry yield (Figure C.6a). These prototypes were clustered by using the K-means algorithm. According to the Davies-Bouldin index, the map was divided into 6 clusters exhibiting similar features that influence Andean blackberry productivity (Figure C.6b).

C.3.4 Component planes and variable dependencies

In order to improve the visualization of the dependencies between the clusters shown in the Kohonen map (Figure C.6b) the "component planes" of Andean blackberry productivity (Figure C.7a), and the variables previously identified as the most relevant for modeling Andean blackberry yield: effective soil depth (Figure C.8), the average temperature of the harvest month, the average temperature of the first, second and third months before harvest (Figure C.9), two specific geographic areas (Figure C.10 and Figure C.11), external drainage (Figure C.12), and the accumulated precipitation of the first month before harvest (Figure C.13), were separated from the Kohonen map and displayed as lattices.

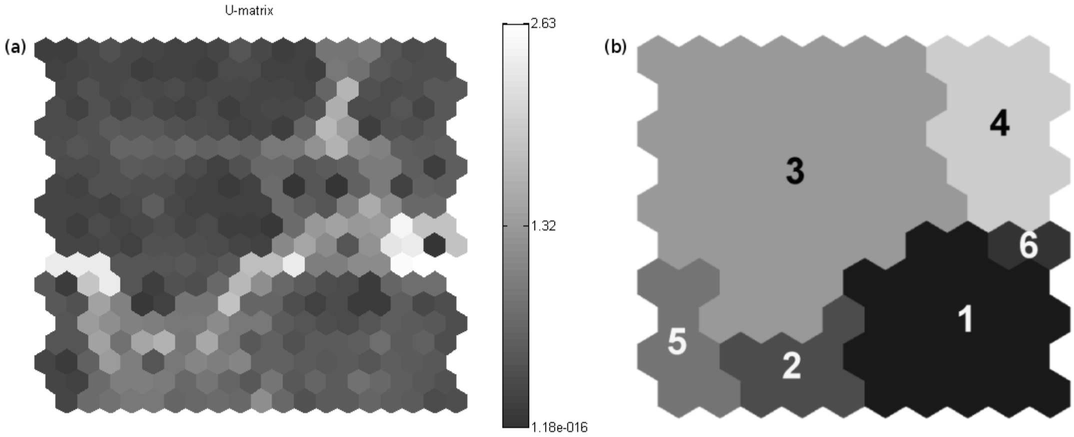


Figure C.6: *Kohonen map showing the resulting clusters. (a) U-matrix displaying the distance among prototypes. The scale bar (right) indicates the values of distance. The upper side exhibits high distances, whilst the lower displays low distances. (b) Kohonen map displaying the 6 clusters obtained after using the K-means algorithm and the Davies-Bouldin index*

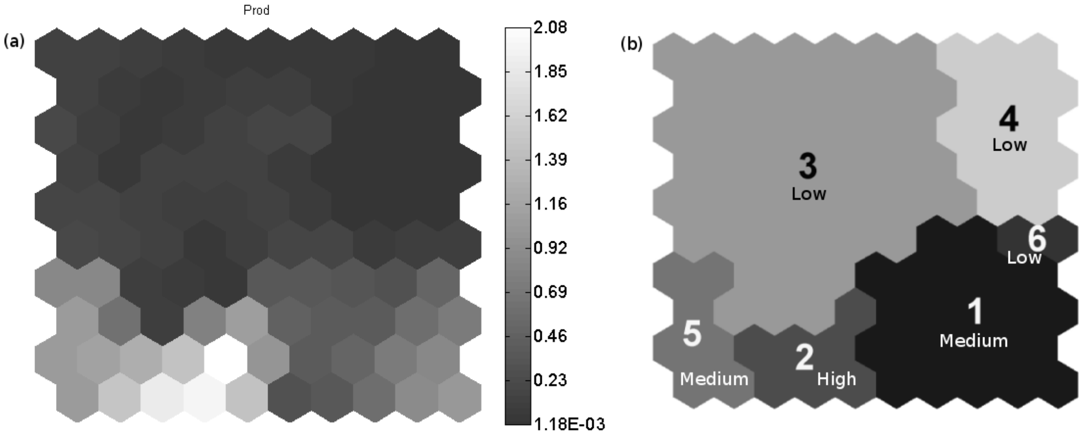


Figure C.7: *(a) Component plane of Andean blackberry yield, the scale bar (right) indicates the range value of productivity in kg/plant/week .The upper side exhibits high values of yield, whereas the lower displays low values. (b) Kohonen map displaying the resultant 6 clusters and their labels according to yield values*

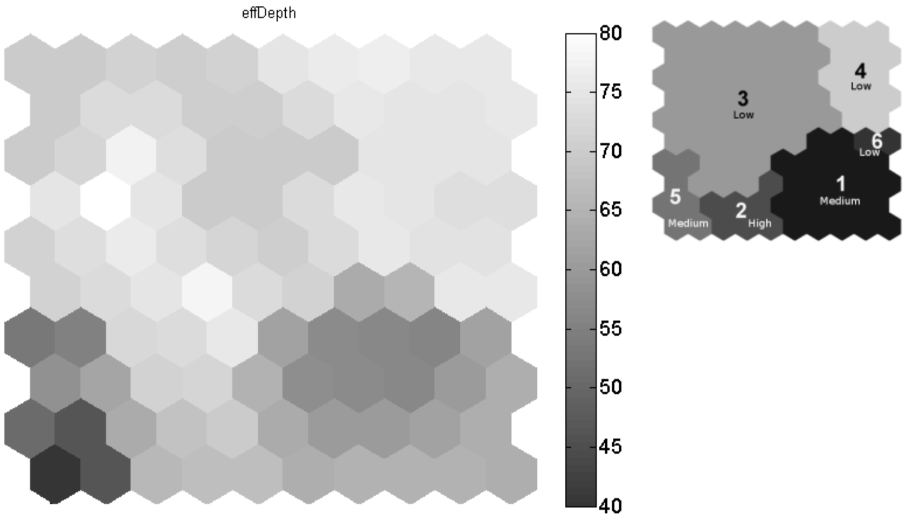


__Figure C.8:__ Component plane of effective soil depth. The scale bar (right) indicates the range value in centimeters of soil depth, the upper side of the scale exhibits high values, whereas the lower displays low values

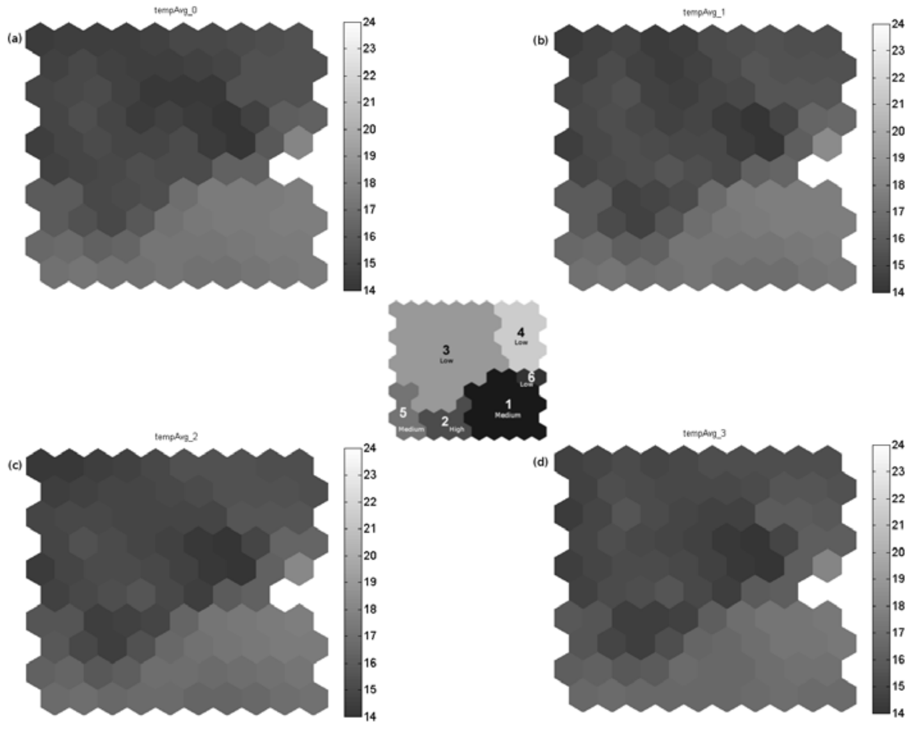


__Figure C.9:__ Components planes of the averages temperature: (a) temperature of the harvest month, (b) average temperature of the first month before harvest, (c) average temperature of the second month before harvest, and (d) average temperature of the third month before harvest. In all figures, the scale bar (right) indicates the range value in °C of temperature. The upper side exhibits high values, whereas the lower displays low values

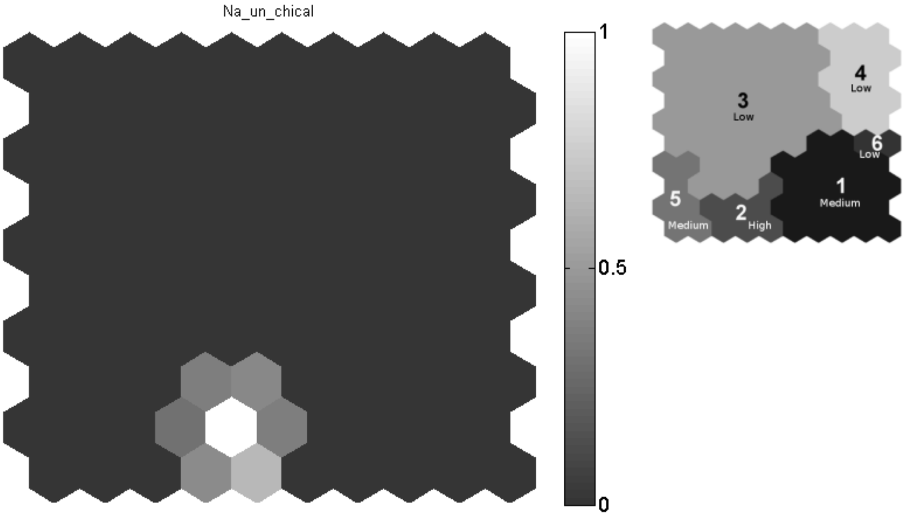


Figure C.10: *Component plane of the specific geographic area Nariño-la union-chical alto. The highest values indicate presence and the lowest absence as they are categorical variables*

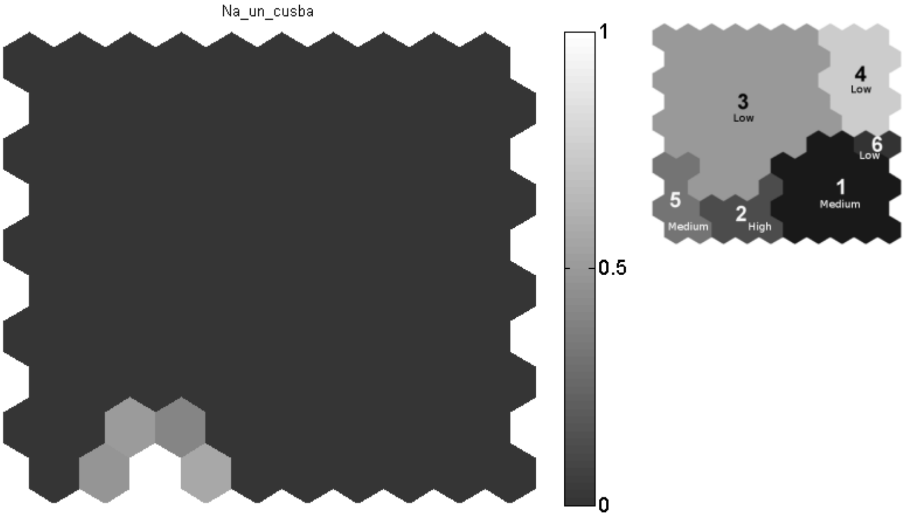


Figure C.11: *Component plane of the specific geographic area Nariño-la union-cusillo bajo. The highest values indicate presence and the lowest absence as they are categorical variables*

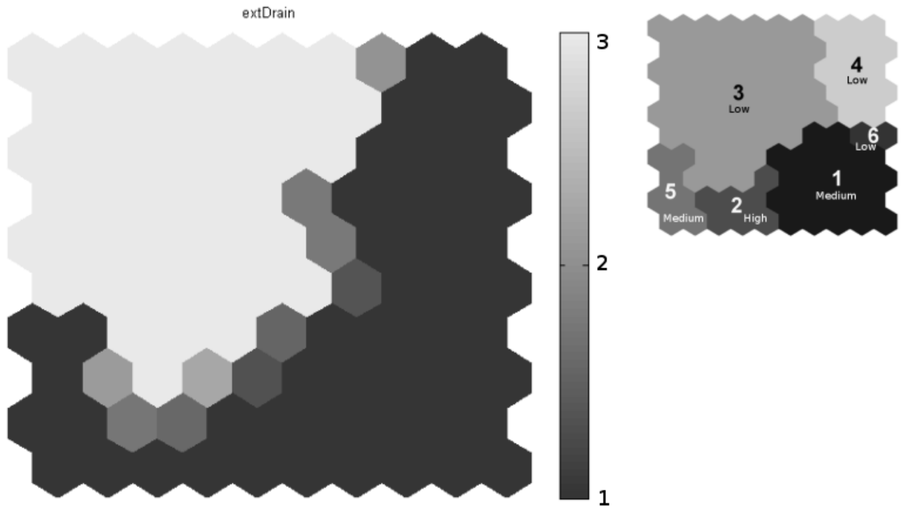


Figure C.12: *Component plane of external drainage. In the scale bar (right), the highest value 3 indicates excellent or fast drainage, 2 moderate drainage, and 1 poor or slow drainage*

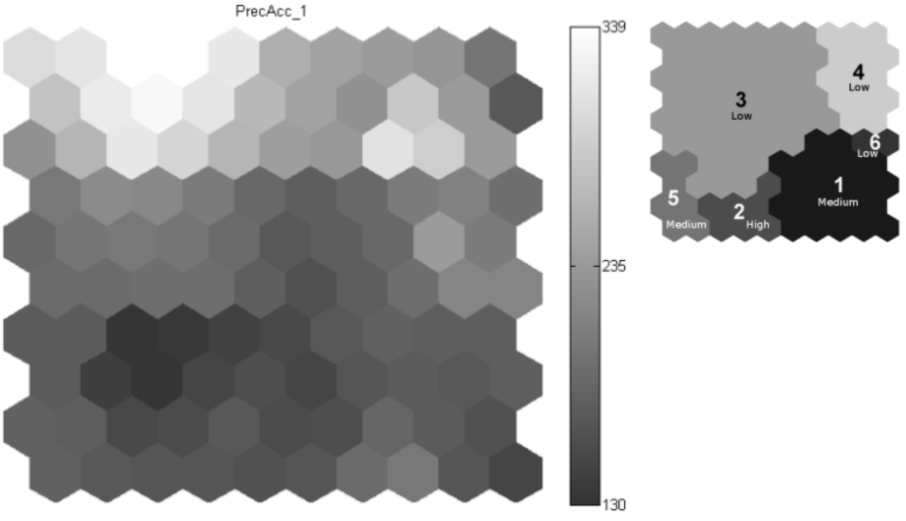


Figure C.13: *Component plane of the accumulated precipitation of the first month before harvest. The scale bar (right) indicates the range value in millimeters of rainfall, the upper side of the scale exhibits high values, whereas the lower displays low values*

C.3.4.1 Productivity plane

Yields greater than 1.16 kg/plant/week were associated with regions in cluster 2 on the Kohonen map (Figure C.7a and b). Yield values between 0.0018 and 1.16 kg/plant/week correspond to clusters 1, 3, 4, 5 and 6 in the Kohonen map. Being 3, 4, 6 the clusters with lowest yields.

C.3.4.2 Effective soil depth

Values of soil depth greater than 70 centimeters (cm) are associated with clusters 3, 4 and 6 (Figure C.8) which are all associated with low yields. In contrast, a soil depth between 40 and 70 cm appears to be related to medium to high yield clusters (1, 2, 5). The cluster with the highest yields had soil depths in the range of 60-70 cm suggesting that this level of soil depth is optimal, and that an effective soil depth greater than 70 cm is not necessary to obtain high yields. Franco and Giraldo [50] stated that for optimal Andean blackberry development, soil depth should be deep enough to allow soil moisture retention without problems of water logging. We suggest that although soil depths above 70 cm were associated with low yields in this study this is probably due to other factors associated with the deeper soils.

C.3.4.3 Average temperature of the harvest month and average temperatures of the first, second and third months before harvest

The Kohonen maps for temperature of the first, second and third months before harvest were similar (Figure C.9). The multilayer perceptron showed that the average temperature of the first month before harvest was more important than the others temperatures (that occurs due to small differences captured to better fit the output). However, the similarity of the components of temperature is probably due to the low monthly variation in temperature under the equatorial conditions of this study. The similarity of the temperature patterns induced us to analyze them as a group rather than separately. It is immediately evident that cluster 6 with temperatures of about 24 °C is not suitable for high yields of blackberries (Figure C.9). Clusters 1, 2 and 5 with medium to high yields are related to temperatures between 16 and 18°C (Figure C.9) and low yields appear to be associated with temperatures in the range of 14-15°C. Andean blackberry experts suggest the optimal temperature for a healthy growth of this crop is between 11 and 18 °C. We suggest a narrower temperature range with 16-18 °C associated with high yields and lower yields as the temperature moves above or below this range.

C.3.4.4 Geographic areas as proxy for crop management

Proxies can be used to estimate the effect of either immeasurable or unobservable variables on a given phenomenon [115, 111, 56, 8, 48, 88]. In our study, geographical areas were integrated into the model with the aim of capturing the effect of variables that were not measured. The geographical proxies were added to the analysis specifically to take into account management and social factors which were not captured by the data collection process and which are likely to be related to the geographic location of a site. For example, farmers from a given locality are likely to use similar management practices that will differ from those used by other communities living in distant localities. The localities Nariño-la union-chical alto (Figure C.10), and Nariño-la union-cusillo bajo (Figure C.11) were associated with cluster 2 which is characterized by the highest yields. Whilst the association with high yields could be a consequence of specific local environmental conditions not accounted for by the environmental variables used in the model, we suggest that is more likely that they are due to particular crop management practices related to local knowledge and socio-economic circumstances. In sugarcane certain groups of farmers consistently obtain higher yields than others even in the same edapho-climatic conditions [66]. The difference is due to better management by certain groups which is related to socio-economic factors including access to knowledge on optimal production practices.

C.3.4.5 External drainage and accumulated precipitation of the first month before harvest

Scrutiny of the external drainage lattice (Figure C.12) gave no obvious clues as to how drainage affects the yield of blackberries. In fact medium yield in cluster 5 is associated with poor external drainage and in cluster 2 with high yields the external drainage is highly variable. However, in all clusters with medium or high yields poor external drainage is associated with low precipitation of the first month before harvest (Figure C.13): not only does this appear to be true from the Kohonen maps, but it also makes agronomic sense. Good external drainage is evidently more important when rainfall is greater. This example clearly indicates how the visual inspection of the Kohonen maps can assist in understanding how various factors effect the growth and development of the crop and the interactions between them. Further inspection of Figure C.12 and Figure C.13 indicate that excellent external drainage is not sufficient to overcome the effects of high or moderate precipitation with moderate external drainage in cluster 3. Overall there was a tendency for low rainfall to be advantageous but there were exceptions. However, when the two variables, precipitation of the first month before harvest and external drainage are taken together it is clear that low rainfall accompanied with varied external drainage conditions can provide good yields, but that heavier precipitation of the first month before harvest with poor drainage is not conducive to high levels of productivity.

C.4 Conclusions

Data collected by small farmers in the Andes couple with information from existing data bases was successfully used to characterize specific production events and to relate production to site and time specific events. The analysis approach focuses first on identifying those variables that explain most of the yield variability by means of artificial neural networks (multilayer perceptron), and then using the Self-Organizing Maps as a tool for dimensionality reduction and visualization of input-input and input-output dependencies.

Artificial neural networks were found to be an effective tool for managing the highly variable, noisy, and qualitative nature of agricultural information collected by farmers and linked to publicly available climate databases. Multilayer perceptrons were used to develop a model based on dataset with 28 variables. This model explained close to 90% of the variation in a validation set. Sensitivity analysis was used to identify the most important variables in determining variation in yield. Self-Organizing Maps were then used to group Andean blackberry yield from different sites according to similarity of growth conditions and management. Data was not available to directly evaluate management practices, so localities were used as a proxy for management. The SOM provided a straightforward manner to visualize the distribution of the variables that affected yield. "Component planes" generated by SOM illustrated the association of these variables with yield and identified two geographic areas as highly productive. The optimal conditions for high yields are an average temperature between 16 and 18 °C, an effective soil depth between 60 and 70 cm, and low rainfall during the first month before harvest in poor external drainage locations or moderate to low rainfall in better drained areas.

The identification of geographic areas with higher yields than those that would be expected solely from the environmental conditions suggests that the farmers in those geographical areas were managing their crops particularly effectively. However, there was not sufficient information to precisely determine which management factors led to the high yields. At the same time the mere identification of areas with farmers that properly manage their crops, offers the chance for these farmers to disseminate their knowledge to other farmers with similar environmental conditions so that they too can improve yields.

Acknowledgements This work is part of a cooperation project between BIOTEC, CIAT, and HEIG-VD (Switzerland) named "Precision agriculture and the construction of field-crop models for tropical fruits". The economical support is given by several institutions in Colombia (MADR, COLCIENCIAS, ACCI) and the State Secretariat for Education and Research (SER) in Switzerland.

Bibliography

[1] Colombian sugarcane research center. http://www.cenicana.org/.

[2] United States Department of Agriculture. http://www.usda.gov/.

[3] Agro-ecological zoning: Guidelines. FAO. Food and Agricultural Organization of the United Nations, Rome., 1996.

[4] Information sheet on rubus glaucus in new world fruits database. http://www.bioversityinternational.org/, July 2008.

[5] Clark Labs, Clark University. IDRISI Taiga, integrated GIS and Image Processing software solution. http://www.clarklabs.org/index.cfm, 2010.

[6] Envi, software solution for processing and analyzing geospatial imagery. http://www.ittvis.com/ProductServices/ENVI.aspx, 2010.

[7] TNT, full-featured GIS and image processing system. http://www.microimages.com/, 2010.

[8] J. Adami, G. Gridley, O. Nyren, M. Dosemeci, M. Linet, B. Glimelius, A. Ekbom, and S.H. Zahm. Sunlight and non-Hodgkin's lymphoma: a population-based cohort study in Sweden. *Int. J. Cancer*, 80:641–645., 1999.

[9] Ethem Alpaydin. *Introduction to Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, October 2004.

[10] D. Alvarez, M. Estrada, and J Cock. Rasta (rapid soil and terrain assessment). Universidad Nacional de Colombia, Palmira, Colombia, 2004.

[11] M. Anderberg. *Cluster analysis for applications*. New York : Academic Press, 1973.

[12] G. Andrienko, D.and M. May Malerba, and M. Teisseire. Mining spatio-temporal data. *J. Intell. Inf. Syst.*, 27(3):187–190, 2006.

[13] E. Backer and A Jain. A clustering performance measure based on fuzzy set decomposition. *IEEE Trans. Pattern Anal. Mach. Intell*, PAMI-3(1):66–75, 1981.

[14] G. Barreto and A. Araújo. Time in self-organizing maps: An overview of models. *International Journal of Computer Research*, 10:139–179, 1990.

[15] M. Barreto-Sanz and A. Pérez-Uribe. Classification of similar productivity zones in the sugar cane culture using clustering of som component planes based on the som distance matrix. In *The 6th International Workshop on Self-Organizing Maps (WSOM)*, 2007.

[16] M. Barreto-Sanz and A. Pérez-Uribe. Improving the correlation hunting in a large quantity of som component planes. In *ICANN 2007: Proceedings of the 17th international conference on Artificial Neural Networks, Part II*, pages 379–388, 2007.

[17] M. Barreto-Sanz and A. Pérez-Uribe. Tree-structured self-organizing map component planes as a visualization tool for data exploration in agro-ecological modeling. In *in Proc. of the 6th European Conf. on Ecological Modelling, Trieste, Italy*, pages 55–56, Nov 2007.

[18] M. Barreto-Sanz, A. Pérez-Uribe, C. Peña-Reyes, and M. Tomassini. Fuzzy growing hierarchical self-organizing networks. In *ICANN 2008: Proceedings of the 18th international conference on Artificial Neural Networks, Part II*, pages 713–722, Berlin, Heidelberg, 2008. Springer-Verlag.

[19] M. Barreto-Sanz, A. Pérez-Uribe, C. Peña-Reyes, and M. Tomassini. Tuning parameters in fuzzy growing hierarchical self-organizing networks. In Leonardo Franco, David Elizondo, and José Jerez, editors, *Constructive Neural Networks*, volume 258 of *Studies in Computational Intelligence*, pages 261–279. Springer Berlin, Heidelberg, 2009.

[20] T. Bell. Space-time stochastic model of rainfall for satellite remote-sensing studies. *J. Geophys. Res.-Atmos.*, 92:9631–9643, 1987.

[21] J. Bezdek. Pattern recognition with fuzzy objective function algorithms. *Plenum Press, New York*, 1981.

[22] J. Bezdek, K. Tsao, and R. Pal. Fuzzy kohonen clustering networks. In *Fuzzy Systems, IEEE Int. Conf. on. pp. 1035–1043.*, 1992.

[23] S. Bimonte, Tchounikine A., and M. Miquel. Towards a spatial multidimensional model. In Song and J. Trujillo, editors, *DOLAP05, ACM Eighth International Workshop on Data Warehousing and OLAP*, pages 39–46. ACM, November 2005.

[24] C. Bishop. *Neural Networks for Pattern Recognition.* Press, Oxford, 1995.

[25] C. Bishop. A hierarchical latent variable model for data visualization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20:281–293, 1997.

[26] G. Brown, J. Wyatt, R. Harris, and X. Yao. Diversity creation methods: a survey and categorisation. *Inform. Fusion*, 6 (1):5–20, 2005.

[27] V. Burzevski and Mohan C:. Hierarchical growing cell structures, tech report: Syracuse university, 1996.

[28] J. Carbonell, A. Amaya, B. Ortiz, J. Torres, R Quintero, and C. Isaacs. Zonificación agroecológica para el cultivo de caña de azúcar en el valle del río Cauca. Tercera aproximación. Technical report, CENICAÑA. Serie Técnica. 29 Cali, Colombia., 2001.

[29] A. Castrignanó, D. De Benedetto, G. Girone, and D. Guastaferro, F.and Sollitto. Characterization, delineation and visualization of agro-ecozones using multivariate geographical clustering. *Italian Journal of Agronomy*, 5(2):121–132, 2010.

[30] Long Chen and C. L. Philip Chen. Pre-shaped fuzzy c-means algorithm (pfcm) for transparent membership function generation. In *SMC*, pages 789–794, 2007.

[31] Park Y. Moon K. Cha Y. Chon, T. Patternizing communities by using an artificial neural network. *Ecological Modelling*, 90:69–78, 1996.

[32] Tae-Soo Chon, Young Seuk Park, Kyong Hi Moon, and Eui Young Cha. Patternizing communities by using an artificial neural network. *Ecological Modelling*, 90(1):69 – 78, 1996.

[33] T. Chow and M. Rahman. Multilayer som with tree-structured data for efficient document retrieval and plagiarism detection. *Trans. Neur. Netw.*, 20(9):1385–1402, 2009.

[34] H. Chung, J. Hsieh, and T. Chang. Prediction of daily maximum ozone concentrations from meteorological conditions using a two-stage neural network. *Journal of Atmospheric Research*, 81(2):124–139, 2006.

[35] James Cock. Sharing commercial information. in: Innovation workshop for the agricultural sector: Site specific agriculture based on sharing farmers experiences. ciat, cali, colombia. http://biotec.univalle.edu.co/Memorias.htm, October 2007.

[36] P Compieta, S. Di Martino, M. Bertolotto, F. Ferrucci, and T. Kechadi. Exploratory spatio-temporal data mining and visualization. *J. Vis. Lang. Comput.*, 18(3):255–279, 2007.

[37] D. Davies and D. Bouldin. A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-1(2):224–227, 1979.

[38] Ángel Dayron Sora, Gerhard Fischer, and Rafael Flórez. Almacenamiento refrigerado de frutos de mora de castilla (rubus glaucus benth.) en empaques con atmósfera modificada. *Agronomía Colombiana*, 24:306–316, 2006.

[39] Virginie de Boishebert, Jean-Luc Giraudel, and Michel Montury. Characterization of strawberry varieties by spme-gc-ms and kohonen self-organizing map. *Chemometrics and Intelligent Laboratory Systems*, 80(1):13 – 23, 2006.

[40] P. Demartines and J. Hérault. Curvilinear component analysis: a self-organizing neural network for nonlinear mapping on data sets. *IEEE Transactions on Neural Network.*, 8:148–154., 1997.

[41] E. Diday and J.C. Simon. Clustering analysis. *Digital Pattern Recognition*, 1:47–94, 1976.

[42] T. Dietterich. Ensemble methods in machine learning. in: Multiple classifier systems. In *First International Workshop (MCS 2000), Cagliari, Italy, pp. 1–15.*, 2000.

[43] J. Doherty, Adams G., and N. Davey. Treegng, hierarchical topological clustering. In *In Proc. Euro. Symp. Artificial Neural Networks*, pages 19–24, 2005.

[44] K. Doherty, R. Adams, R. Doherty, K. and. Adams, and N. Davey. Hierarchical growing neural gas. In *in Proc. Int. Conf. Adaptive and Natural Computing Algorithms*, pages 140–143, 2005.

[45] B. Efron. Estimating the error rate of a prediction rule: improvement on cross-validation. *J. Am. Stat. Assoc*, 78 (382):316–331, 1983.

[46] S. Everitt, B. Landau and M. Leese. *Cluster Analysis*. London: Arnold, 2001.

[47] T. Farr and M. Kobrick. Radar topography mission produces a wealth of data. *American geophysical. Union Eos.*, 81:583–585, 2000.

[48] Pritchett L. Filmer, D. The effect of household wealth on educational attainment: evidence from 35 countries. *Popul. Dev. Rev.*, 25:85–120., 1999.

[49] Günther Fischer, Harrij Van Velthuizen, Freddy O. Nachtergaele, and Arne Jernelöv. Food and agriculture organization of the united nations, 1999.

[50] G. Franco and M.J. Giraldo. El cultivo de la mora. Technical Report 5, Corpoica, Manizales, 2002.

[51] B. Fritzke. Growing cell structures - a self-organizing network for unsupervised and supervised learning. *Neural Networks*, 7:1441–1460, 1994.

[52] B Fritzke. Some competitive learning methods, 1997.

[53] A. Geva. Feature extraction and state identification in biomedical signals using hierarchical fuzzy clustering. *Medical and Biological Engineering and Computing*, 36:608–614, 1998.

[54] J. L. Giraudel and S. Lek. A comparison of self-organizing map algorithm and some conventional statistical methods for ecological community ordination. *Ecological Modelling*, 146(1-3):329 – 339, 2001.

[55] D. Goldin and P. Kanellakis. On similarity queries for time-series data: Constraint specification and implementation. pages 137–153, 1995.

[56] K. Goodman, P. Correa, H. Tengana, H. Ramirez, J. DeLany, O. Pepinosa, M. Quinõnes, and T. Parra. Helicobacter pylori infection in the colombian andes: a population-based study of transmission pathways. *Am. J. Epidemiol.*, 144:290–299, 1996.

[57] C. Goutte. Note on free lunches and cross-validation. *Neural. Comput.*, 9 (6):1245–1249, 1997.

[58] D. Guo, J. Chen, A. MacEachren, and K. Liao. A visual inquiry system for space-time and multivariate patterns (vis-stamp). *IEEE Transactions on Visualization and Computer Graphics*, 12(6):1461–1474, 2006.

[59] Y Hashimoto. Applications of artificial neural networks and genetic algorithms to agricultural systems (special issue). *Computers and Electronics in Agriculture*, 18:71–72, 1997.

[60] R Hijmans, S Cameron, J. Parra, P. Jones, and A. Jarvis. Very high resolution interpolated climate surfaces for global land areas. *Int. J. Clim.*, 25:1965–1978, 2005.

[61] J. Himberg. *Enhancing the SOM-based Data Visualization by Linking Different Data Projections*, pages 427–434. Springer, 1998.

[62] J. Himberg. Enhancing the som-based data visualization by linking different data projections. In *Proceedings of 1st International Symposium IDEAL'98, Intelligent Data Engineering and Learning–Perspectives on Financial Engineering and Data Mining. 427–434.*, 1998.

[63] V. Hodge and j. Austin. Hierarchical growing cell structures: Treegcs. *IEEE Trans. Knowledge and Data Engineering*, 13:2001, 2000.

[64] F. Hoffman, W. Hargrove, R. Mills, S. Mahajan, E. David, and O. Robert. Multivariate spatio-temporal clustering (mstc) as a data mining tool for environmental applications. In *iEMSs Fourth Biennial Meeting: International Congress on Environmental Modelling and Software (3)*, pages 1774–1781. International Environmental Modelling and Software Society (iEMSs), 2008.

[65] T. Huntsberger and P. Ajjimarangsee. Parallel self-organizing feature maps for unsupervised pattern recognition, 1989.

[66] C.H. Isaacs, J.A. Carbonell, A. Amaya, J.S. Torres, J.I. Victoria, R. Quintero, A.E. Palma, and J.H. Cock. Site specific agriculture and productivity in the colombian sugar industry. In *Proc. 26th congress International Society of Sugar Cane Technologists (ISSCT)*, Durban, South Africa, 2007.

[67] Abhishek Jain. Predicting air temperature for frost warning using artificial neural networks. Master's thesis, 2003.

[68] Murty M. Jain, A. and P. Flynn. Data clustering: a review. *ACM Comput. Surv.*, 31, 3:264–323, 1999.

[69] D. Jiménez, J. Cock, H. Satizábal, M. Barreto-Sanz, A. Pérez-Uribe, A. Jarvis, and P. Van Damme. Analysis of andean blackberry (rubus glaucus) production models obtained by means of artificial neural networks exploiting information collected by small-scale growers in colombia and publicly available meteorological data. *Comput. Electron. Agric.*, 69(2):198–208, 2009.

[70] Daniel Jiménez, Andres Perez-Uribe, Héctor Satizábal, Miguel Barreto, Patrick Van Damme, and Marco Tomassini. A survey of artificial neural network-based modeling in agroecology. In *Soft Computing Applications in Industry*. Springer Berlin / Heidelberg, 2008.

[71] D. Jiménez, H. Satizábal, and Pérez-Uribe A. Modelling sugar cane yield using artificial neural networks. In *In: Proceedings of the 6th European Conference on Ecological Modelling (ECEM'07), Trieste, Italy, pp. 244–245.*, 2007.

[72] J. Jones, J. Hansen, F. Royce, and C. Messina. Potential benefits of climate forecasting to agriculture. *Agriculture, Ecosystems and Environment*, 82:169–184, 2000.

[73] Eamonn Keogh and Jessica Lin. Clustering of time series subsequences is meaningless: Implications for past and future research. In *In Proc. of the 3rd IEEE International Conference on Data Mining*, pages 115–122, 2003.

[74] T. Kohonen. Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, 43(1):56–69, 1982.

[75] T. Kohonen. *Self-Organizing Maps*, volume 30 of *Springer Series in Information Sciences*. Springer Verlag, Third edition, 2001. ISBN 3–540–67921–9, ISSN 0720–678X.

[76] E. Koua and M. Kraak. Alternative visualization of large geospatial datasets. *Cartographic Journal*, 41(3):217–228, 2004.

[77] J. Lampinen and E. Oja. Clustering properties of hierarchical self-organizing maps. *J. Math. Imag. Vis.*, 2(2):261–272, 1992.

[78] M. Liu and A. Samal. A fuzzy clustering approach to delineate agroecozones. *Ecological Modelling*, 149(3):215 – 228, 2002.

[79] Y. Liu, H. Weisberg, and R. He. Sea surface temperature patterns on the west florida shelf using growing hierarchical self-organizing maps. *Journal of Atmospheric and Oceanic Technology*, 23(2):325–338, 2006.

[80] S. Luttrell. Hierarchical self-organizing networks. In *In Proceedings of the 1st IEE Conference on Artificial Neural Networks, London, UK, British Neural Network Society.*, pages 2–6, 1989.

[81] J. MacQueen. Some methods for classification and analysis of multivariate observations. In L. M. Le Cam and J. Neyman, editors, *Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297. University of California Press, 1967.

[82] T. Martinez and J. Schulten. Topology representing networks. *Neural Networks*, 7(3):507–522, 2002.

[83] M. Mendes and L. Sacks. A scalable hierarchical fuzzy clustering algorithm for text mining. In *In Proceedings of the 5th International Conference on Recent Advances in Soft Computing*, 2004.

[84] D. Merkl, H. He, M. Dittenbach, and A. Rauber. Adaptive hierarchical incremental grid growing: An architecture for high-dimensional data visualization. In *In Proc. Workshop on SOM, Advances in SOM*, pages 293–298, 2003.

[85] Yuxin Miao, David Mulla, and Pierre Robert. Identifying important factors influencing corn yield and grain quality variability using artificial neural networks. *Precision Agriculture*, 7:117–135, May 2006.

[86] R. Miikkulainen. Script recognition with hierarchical feature maps. *Connection Science*, 2:83–101, 1990.

[87] H. Miller and J. Han. *Geographic Data Mining and Knowledge Discovery*. CRC Press, 2 edition, Bristol, PA, USA, 2009.

[88] Gragnolati M. Burke K.A. Paredes E. Montgomery, M.R. Measuring living standards with proxy variables. *Demography*, 155–174.:37, 1999.

[89] D. Moshou, C. Bravo, J. West, S. Wahlen, Alastair McCartney, and Herman Ramon. Automatic detection of yellow rust in wheat using reflectance measurements and neural networks. *Computers and Electronics in Agriculture*, 44(3):173 – 188, 2004.

[90] H. Murase. Artificial intelligence in agriculture. *Computers and Electronics in Agriculture*, 29(1-2):1 – 2, 2000.

[91] S.M. Shiva Nagendra and Mukesh Khare. Artificial neural network approach for modelling nitrogen dioxide dispersion from vehicular exhaust emissions. *Ecological Modelling*, 190(1-2):99 – 115, 2006.

[92] Norbert Niederhauser, Thomas Oberthür, Sibylle Kattnig, and James Cock. Information and its management for differentiation of agricultural products: The example of specialty coffee. *Comput. Electron. Agric.*, 61(2):241–253, 2008.

[93] Peter A. Noble and Erik H. Tribou. Neuroet: An easy-to-use artificial neural network for ecological and biological modeling. *Ecological Modelling*, 203(1-2):87 – 98, 2007. Special Issue on Ecological Informatics: Biologically-Inspired Machine Learning, 4th Conference of the International Society for Ecological Informatics.

[94] N. Pal and J. Bezdek. On cluster validity for the fuzzy c-means model. *Fuzzy Systems, IEEE Transactions on*, 3(3):370–379, 1995.

[95] P. A. Paul and G. P. Munkvold. Regression and artificial neural network modeling for the prediction of gray leaf spot of maize. *Phytopathology*, 95(4):388–396, 2005.

[96] X. Qiang, G. Cheng, and Z. Li. A survey of some classic self-organizing maps with incremental learning. In *Signal Processing Systems (ICSPS), 2010 2nd International Conference on*, 2010.

[97] X. Qiang, G. Cheng, and Z. Wang. An overview of some classical growing neural networks and new developments. In *Education Technology and Computer (ICETC), 2010 2nd International Conference on*, 2010.

[98] A. Rauber, D. Merkl, and M. Dittenbach. The growing hierarchical self-organizing map: Exploratory analysis of high-dimensional data. *IEEE Transactions on Neural Networks*, 13(6):1331–1341, 2002.

[99] J. Roddick, E. Hoel, M. Egenhofer, D. Papadias, and B. Salzberg. Spatial, temporal and spatio-temporal databases - hot issues and directions for phd research. *SIGMOD Rec.*, 33(2):126–131, 2004.

[100] J. Roddick, K Hornsby, and M. Spiliopoulou. An updated bibliography of temporal, spatial, and spatio-temporal data mining research. In *TSDM '00: Proceedings of the First International Workshop on Temporal, Spatial, and Spatio-Temporal Data Mining-Revised Papers*, pages 147–164, London, UK, 2001. Springer-Verlag.

[101] J. Roddick and B. Lees. Paradigms for spatial and spatio-temporal data mining. In *In Geographic Data Mining and Knowledge*. Taylor and Francis, 2001.

[102] J. Roddick and B. Lees. *Spatiotemporal Data Mining Paradigms and Methodologies*. Taylor and Francis, New York, 2009.

[103] T. Samad and S. Harp. Self-organization with partial data. *Network: Computation in Neural Systems*, 3:205–212, 1992.

[104] J. Sammon. A nonlinear mapping for data structure analysis. *IEEE Transactions on Computers.*, 18:401–409., 1969.

[105] Daniel J. Sargent. Comparison of artificial neural networks with other statistical approaches. *Cancer*, 91(S8):1636–1642, 2001.

[106] H. Satizábal, M. Barreto-Sanz, D. Jiménez, A. Perez-Uribe, and J. Cock. Enhancing decision-making processes of small farmers in tropical crops by means of machine learning models. EPFL / UNESCO chair international scientific conference on technologies for development, 2010.

[107] H. Satizábal and A. Pérez-Uribe. Relevance metrics to reduce input dimensions in artificial neural networks. In Joaquim de Sá, Luís Alexandre, Wlodzislaw Duch, and Danilo Mandic, editors, *Artificial Neural Networks ICANN 2007*, volume 4668 of *Lecture Notes in Computer Science*, pages 39–48. Springer Berlin / Heidelberg, 2007.

[108] A. Schultz and R. Wieland. The use of neural networks in agroecological modelling. *Computers and Electronics in Agriculture*, 18(2-3):73 – 90, 1997. Applications of Artificial Neural Networks and Genetic Algorithms to Agricultural Systems.

[109] A. Schultz, R. Wieland, and G. Lutze. Neural networks in agroecological modelling – stylish application or helpful tool? *Computers and Electronics in Agriculture*, 29(1-2):73 – 97, 2000.

[110] M. Smith, M. Goodchild, and P. Longley. *Geospatial Analysis: A Comprehensive Guide to Principles, Techniques and Software Tools*. Matador; 3rd Revised edition edition, 2009.

[111] R.H. Steckel. Stature and standard of living. *J. Econ. Lit.*, 33:1903–1940, 1995.

[112] A. Strehl and J. Ghosh. Relationship-based clustering and visualization for high-dimensional data mining. *INFORMS Journal on Computing*, 15:208–230, 2003.

[113] M. Sultan, D. Wigle, C Cumbaa, M. Maziarz, J. Glasgow , M. Tsao, and I. Jurisica. Binary tree-structured vector quantization approach to clustering and visualizing microarray data. *Bioinformatics*, 18:111–119, 2002.

[114] H. Taniichi, N. Kamiura, T. Isokawa, and N Matsui. On hierarchical self-organizing networks visualizing data classification processes. In *In Proc. International Conference on Instrumentation, Control and Information Technology (SICE), 2007 Annual Conference*, pages 1958–1963, 2007.

[115] D. Thomas, J. Strauss, and M. Henriques. Child survival, height for age and household characteristics in Brazil. *J. Dev. Econ.*, 33:197–234., 1990.

[116] V. Tryba and K. Goser. Self-organizing feature maps for process control in chemistry. In *Proc. ICANN, Helsinki pp. 847-852*, 1991.

[117] E. Tsao, J. Bezdek, and Pal N. Fuzzy kohonen clustering networks. *Pattern Recognition*, 27(5):757 – 764, 1994.

[118] A. Ultsch and H.P. Siemon. Kohonen's self organizing feature maps for exploratory data analysis. In *Proc. Int'l Neural Network Conf. (INNC '90)*, pages 305–308, 1990.

[119] D. Urska. Data mining of geospatial data: combining visual and automatic methods. PhD thesis, royal institute of technology, Stockhol, Sweden., 2006.

[120] National Research Council (U.S.). *Lost crops of the Incas : little-known plants of the Andes with promise for worldwide cultivation / report of an ad hoc panel of the Advisory Committee on Technology Innovation, Board on Science and Technology for International Development, National Research Council*. National Academy Press, Washington, D.C. :, 1989.

[121] A. Vellido, Lisboa P., and Meehan K. Segmentation of the on-line shopping market using neural networks. *Expert Systems with Applications*, 17:303–314, 1999.

[122] J. Vesanto. Som-based data visualization methods. *Intelligent Data Analysis*, 3:111–126, 1999.

[123] J. Vesanto and J. Ahola. Hunting for correlations in data using the self-organizing map. In *Proceeding of the International ICSC Congress on Computational Intelligence Methods and Applications pp. 279–285.*, 1999.

[124] J. Vesanto and E. Alhoniemi. Clustering of the self-organizing map. *IEEE Transactions on Neural Networks.*, 11:586–600., 2000.

[125] J. Vesanto, J. Himberg, M. Siponen, and O. Simula. Enhancing som based data visualization. In *Proceedings of the 5th International Conference on Soft Computing and Information/Intelligent Systems*, pages 64–67. World Scientific, 1998.

[126] J. Vesanto and M. Sulkava. Distance matrix based clustering of the self-organizing map. In José Dorronsoro, editor, *Artificial Neural Networks ICANN 2002*, volume 2415 of *Lecture Notes in Computer Science*, pages 134–134. Springer Berlin Heidelberg, 2002.

[127] D. Vicente and A. Vellido. *Review of Hierarchical Models for Data Clustering and Visualization*. In R.Giráldez, J.C. Riquelme, J.S. Aguilar-Ruiz, 2004.

[128] C. Vörösmarty, C. Fernandez-Jauregui, and M. Donoso. A regional, electronic hydrometeorological data network for south america, central america, and the caribbean. http://www.r-hydronet.sr.unh.edu/english/, November 1995.

[129] C. Williams, W. Hargrove, M. Liebman, and D. James. Agro-ecoregionalization of iowa using multivariate geographical clustering. *Agriculture, Ecosystems and Environment*, 123(1-3):161 – 174, 2008.

[130] S. Wu, M. Rahman, and T. Chow. Content-based image retrieval using growing hierarchical self-organizing quadtree map. *Pattern Recognition*, 38(5):707–722, 2005.

[131] R. Xu and D. Wunsch. Survey of clustering algorithms. *Neural Networks, IEEE Transactions on*, 16(3):645–678, 2005.

[132] X. Yao. Research issues in spatio-temporal data mining. *IEEE Transactions on Knowledge and data*, 14:750–767, 2002.

[133] X. Yao and Y. Liu. Making use of population information in evolutionary artificial neural networks. *IEEE Trans. Syst. Man Cybern . B*, 28 (3):417–425, 1998.