NOVA IMS
Information Management School

MGI

Mestrado em Gestão de Informação
Master Program in Information Management

Contribution Towards Smart Cities: Exploring Block Level Census Data for the Characterization of Change in Lisbon

Jorge Manuel Alves Antunes

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre em Gestão de Informação

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

NOVA Information Management School

Instituto Superior de Estatística e Gestão de Informação

Universidade Nova de Lisboa

CONTRIBUTION TOWARDS SMART CITIES: EXPLORING BLOCK LEVEL CENSUS DATA FOR THE CHARACTERIZATION OF CHANGE IN LISBON

por

Jorge Manuel Alves Antunes

Dissertação apresentada como requisito parcial para a obtenção do grau de Mestre em Gestão de Informação, Especialização em Gestão dos Sistemas e Tecnologias de Informação

Orientador/Coorientador: Fernando José Ferreira Lucas Bação

Coorientador: Roberto André Pereira Henriques

Novembro 2015

DEDICATÓRIA

Dedico à minha família e amigos, em particular Ana e João

AGRADECIMENTOS

A realização deste trabalho só foi possível graças à paciência e compreensão dos meus orientadores, Professor Doutor Fernando Bação e Professor Doutor Roberto Henriques. Agradeço-lhes a oportunidade; e é, foi e continuará a ser uma honra poder aprender e evoluir com a sua instrução.

Agradeço ao Luís Garcia (INE) o fornecimento da base de dados de 2001.

Agradeço à NOVA IMS a disponibilização do ArcGIS que me permitiu evoluir mais rapidamente e também a possibilidade de ser aluno nesta instituição.

A todos o meu muito obrigado.

ABSTRACT

The interest in using information to improve the quality of living in large urban areas and its governance efficiency has been around for decades. Nevertheless, the improvements in Information and Communications Technology has sparked a new dynamic in academic research, usually under the umbrella term of Smart Cities. This concept of Smart City can probably be translated, in a simplified version, into cities that are lived, managed and developed in an information-saturated environment. While it makes perfect sense and we can easily foresee the benefits of such a concept, presently there are still several significant challenges that need to be tackled before we can materialize this vision. In this work we aim at providing a small contribution in this direction, which maximizes the relevancy of the available information resources. One of the most detailed and geographically relevant information resource available, for the study of cities, is the census, more specifically the data available at block level (Subsecção Estatística). In this work, we use Self-Organizing Maps (SOM) and the variant Geo-SOM to explore the block level data from the Portuguese census of Lisbon city, for the years of 2001 and 2011. We focus on gauging change, proposing ways that allow the comparison of the two time periods, which have two different underlying geographical bases. We proceed with the analysis of the data using different SOM variants, aiming at producing a two-fold portrait: one, of the evolution of Lisbon during the first decade of the XXI century, another, of how the census dataset and SOM's can be used to produce an informational framework for the study of cities.

KEYWORDS

Data mining; Self-Organizing Map; Smart City; Geo-SOM; Clustering; Geographic Information Science; Census

INDEX

FIGURES INDEX

TABLES INDEX

LIST OF ACRONYMS AND ABBREVIATIONS

ED	Enumeration District
ICT	Information and Communication Technologies
BGRI	Base de Geográfica de Referenciação de Informação
SOM	Self-Organizing Map
HC	Hierarchical Clustering
PCA	Principal Component Analysis
COS	Carta de Ocupação do Solo

1. INTRODUCTION

The city challenges and urban trends exploration have received a lot of attention concerning the sustainable development (Braulio-Gonzalo, Bovea, & Ruá, 2015; Huang, Yan, & Wu, 2016) considering the four pillars for achieving sustainability of cities, namely Social development, Economic development, Environmental management and Urban governance(United Nations, 2013) , it is necessary to face three major challenges: the increasing economic division between rich and poor, climate change and public goods efficiency (Birch & Wachter, 2011) The pressure put over the cities explicitly the population increasing, in which by 2050 67,2% of the world population will live in urban areas, requires a huge demanding of resources. It is imperative to find solutions to face these challenges, specifically social, economic and environmental (Kondepudi, 2014).

In this demanding environment, it appeared the recent concept of Smart Cities, which is referring to the opportunity to develop strategies to face the challenges that the urban society stand up every day (Roche, 2014). Although the complexity presented by the cities has lead to a non-agreement in the exact definition of the smart city concept (Lombardi, Giordano, Farouh, & Yousef, 2012; Nam & Pardo, 2011) . Nevertheless, it is possible to identify a major set of activities focused on the implementation of technologies and strategies aimed to improve city itself. The current researchers present essentially two main drivers, the solutions based in the Information technologies which fully rely on IoT (Internet of Things) and IoS (Internet of Services) as smart cities enablers through an unified ICT (Information and Communication Technologies) platform (Hernández-Muñoz et al., 2011) and the ones that are more systemic sustained in management and organization, technology, governance, policy, people and communities, economy, built infrastructure and the natural environment. The previous driver views the city as organic complex with multiple and diverse stakeholders, high levels of interdependence, competing objectives and values, and social and political complexity (Chourabi et al., 2012).

An urban fabric could be hard to understand, the inclusion of major sets of attributes like population, infrastructure, social environment, economic environment among others arriving in a constant stream flow revealed the inherit complexity of this task (Lee & Rinner, 2014). The increasing sophistication of the Geographic Information Systems (GIS) contribute to surpass the previous described complications.

Context visualization aims at including the non-spatial high-dimensionality data into a geographical framework. The result is a cartographic representation of multivariable groups depending in what knowledge is desired to obtain. The nongeographic information is processed by computational tools and the results are expressed in maps where it is easier to acquire and interpret the output (Etien L Koua & Kraak, 2004; Penn, 2005; Skupin & Hagelman, 2005; Skupin, 2002).

Although the visualization of high dimension data by humans at a single moment of time represents a problem once it is hard to apprehend several dimensions in an interpretable way (Bação, Lobo, & Painho, 2004). It is necessary to implement techniques that are drivers to the increasing data perception. The dimension reduction task aims to reduce the data parameters to a minimum needed to explain the data properties, known as intrinsic dimensionality (Fukunaga, 1990). As a result, dimensionality reduction facilitates, among others, classification, visualization, and compression of high-dimensional data (van der Maaten, Postma, & van den Herik, 2009).

Lisbon will be the study object due to the intense modifications suffered during the last decades and particularly throughout last recent years (Silva & Syrett, 2006; Veiga, 2014) representing an excellent case study.

The performed analysis enable the population and residential infrastructure characterization through clustering techniques. In order to answer the Smart cities initiative, the city portray enables the identification of the areas where it is possible to emerge the most recent IT tools based on the "intelligent resident users" and at the same time identify the areas where the inhabitants and related infrastructure are unprepared to deal with the Future Internet, excluding all the possibilities allied to the innovative and efficiency processes. The methods applied presented in this work are Self-Organizing Maps and Geo-Self Organizing Maps. Both aim to mitigate the curse of dimensionality to produce a reasonable picture of the city.

2. CONTEXT OF RESEARCH

The concentration of people, investment and resources make the cities potential spots for economic development, innovation and social interaction (Longo, Gerometta, & Haussermann, 2005). Defined by Polèse (2010) the key factors for the grow and prosper of a city are:

- home to a highly skilled and educated population;
- centrally located, at the heart of a rich market, and/or well positioned for trade with expanding markets;
- diversified economy with a significant proportion of high-order services, largely untainted by a legacy of Rustbelt-type industries;
- climate and/or natural setting superior to most of the other cities in the nation

Whilst previous features and characteristics reduce the number of potential successful cities, the emergent concept "Smart Cities" brought a new perspective. The term "smart" means having or showing a quick-witted intelligence, that's why in some literature it is possible to find the term intelligent city, or digital city due to the required speed.

There are several definitions of "Smart Cities" (Nam & Pardo, 2011), although it is possible to assume that a smart city refers to the opportunity to develop strategies to face the challenges the urban society encounters every day (Roche, 2014). The main components of a Smart City were concatenated by Nam & Pardo (2011) as presented in the following picture:

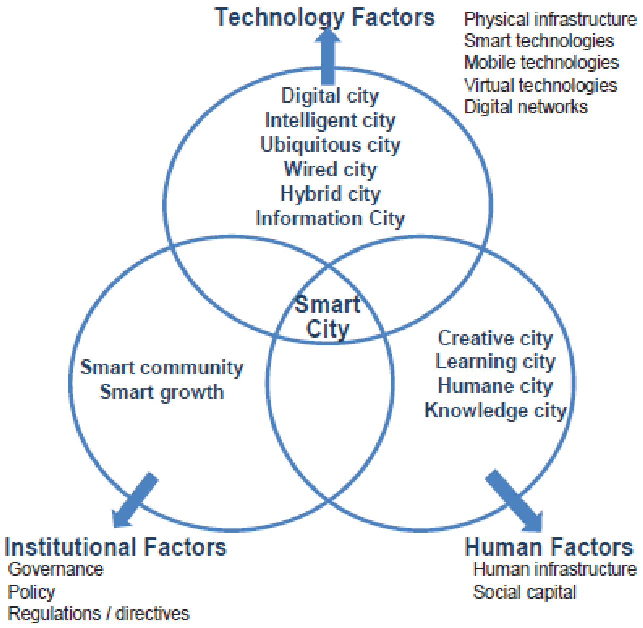


Figure 2-1 Fundamental Components of Smart City (Nam & Pardo, 2011)

It is easily recognizable there are a large set of potential attributes to be used in the transformation of a city in a Smart Sustainable City.

In order to recommend or provide any strategy, integrated or not to achieve the desired sustainable development it is necessary to assess what a city is. Identify the patterns resulted by the dynamic processes such as population mobility, natural growth, socioeconomic development, environmental changes and local and national policies is the first step to be taken.

The process to characterize a city requires the input of several forms of data.

The use of spatial features attached to non-location attributes, like economic, social, demographic data brings new opportunities to develop methods to understand environment phenomena and socio-economic behaviors (Bação, Lobo, & Painho, 2005b).

Spatiotemporal data pose serious challenges to analysts. The number of distinct places can be too large, the time period under analysis be too long, and/or the attributes depending on space and time might be too numerous. Therefore, human analysts require a proper support from computational methods capable to deal with large and multidimensional data (G. Andrienko et al., 2010), as we present further below.

Comprehensive analysis of spatiotemporal data requires consideration of the data in a dual way (N. V. Andrienko & Andrienko, 2006):

- As a temporally ordered sequence of spatial situations. A spatial situation is a particular spatial distribution of objects and/or values of attributes in some time unit (i.e. moment or interval);
- As a set of spatially arranged places where each place is characterized by its particular temporal variation of attribute values and/or presence of objects. We shall call it local temporal variation.

The subsequent analysis based on the previous propositions will bring the high level sub tasks (G. Andrienko et al., 2010; G. Andrienko, Andrienko, Dykes, Fabrikant, & Wachowicz, 2008):

- Analyze the change of the spatial situation over time, i.e. temporal evolution of the situation.
- Analyze the distribution of the local temporal variations over space.

In order to perform an analysis with high standards, the raw data must follow the information quality criteria such as valid, reliable, timely, fit for purpose, accessible, cost effective. Only with good resources it is possible to achieve new knowledge as research success goal (Howard, Lubbe, & Klopper, 2011) . Although the geographic subjects we try to answer are too complex for experimentation, so it necessary to have an observational data source available to address the aimed research (Mennis & Guo, 2009) .

The censuses are one of the biggest data providers. They follow the high quality standards enabling several studies about environment phenomena and socio-economic behaviors.

The amount of data and attributes presented by them (infrastructure, demography, morphology and economic), as well the granularity obtained (Skupin & Agarwal, 2008), which fill the needs from national until block level, enables a sort of analysis which generates all kind of interests (Skupin & Hagelman, 2005).

Looking for the best ways to explore data, the clustering techniques showed several advantages. First, the ability to group similar objects in one cluster and dissimilar ones in other clusters enables the labeling and acting accordingly to their features and characteristics. The objects are automatically assigned to a cluster, meaning that every single item can be described by its representative. These techniques are often scalable and ease to handle, even with different attribute types.

Several methodologies have been applied to the censuses datasets (and others) in order to extract knowledge (Ceylan & Özbay, 2007; Lee & Rinner, 2014). The combination of PCA and Neural Networks performed by Kambhatla & Leen(1997) revealed that the non-linear techniques were more accurate, reducing the resulting mean square error.

The application of the previous techniques require computational tools to process all sort of abundant data. The use of Neural Networks has been exploited, particularly to gather knowledge that is stealthy in big datasets, such as the census database.

The taken path leads us to a method, used in intensive computation, applied to large data sets, the Kohonen Map or Self-Organizing Map (SOM) (Kohonen, 2013). This method performs vector quantization and projection (Kohonen, 2013). The output is a two-dimensional grid, easy to visualize and understand. Using a two-dimensional surface avoids the human inability to realize the visualization of multidimensional data (Bação et al., 2005b; E.L. Koua, 2003) .

Using this technique it's possible to perform a clustering operation by aggregation the items or objects to nearest neuron or unit, as their representative (Jain, Murty, & Flynn, 1999; Mennis & Guo, 2009)

2.1. SELF-ORGANIZING MAP

SOM is an unsupervised Artificial Neural Network, a clustering technique based on the classical vector quantization. The idea is to simulate the brain maps where it is possible to find that certain single neural cells in the brain respond selectively to some specific sensory stimuli (Kohonen, 2013). SOM maps high-dimensional data onto, generally, two dimensions preserving the relations already existents in the input source, meaning that the components that are more similar will appear near to each other and far away from the ones where it is identified bigger dissimilitude.

The following figure shows how the output spaces (a two-dimensional SOM grid (3x3)) adapt to the input space (a three-dimensional dataset):

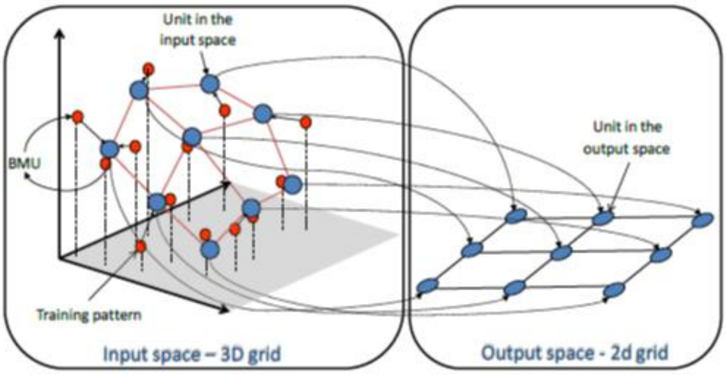


Figure 2-2 Self Organizing Map's output space (two-dimensional) and input space (three-dimensional). Blue circles represent the units of the SOM while the red circles represent the input patterns (R. A. P. Henriques, 2010)

The SOM algorithm performs a number of iterations in order to represent as better as possible the input patterns by the reference vectors. The movements can be visualized as exemplified in the following figure:

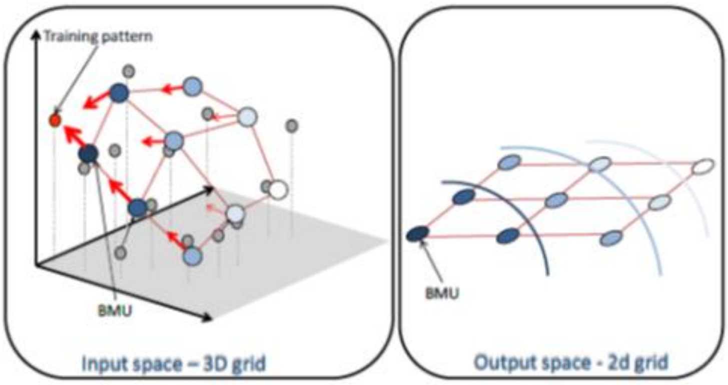


Figure 2-3 SOM training phase. A training pattern (red dot) is presented to the network and the closest unit is selected (BMU). Depending on the leaning rate, this unit moves towards the input pattern (represented by the red arrow). Based on the BMU and on the neighbourhood function, neighbours are selected on the output space (blue lightness represents the degree of neighbourhood). Neighbours are also updated towards the input pattern (R. A. P. Henriques, 2010)

The Best Matching Unit is found, as well the closest neighbour's.

The SOM algorithm could be described as follows:

Consider:

$$\mathrm{X} = \{x_j : j = 1,2 \ldots, m\} \subset \Re^n$$

Where the χ refers to m data patterns with n variables

$$x_j = [x_{j1}, x_{j2}, \ldots x_{jn}]^T \in \Re^n$$

The nodes i are defined in the input space $\Re^n$ by the reference vector mi and the location ri.

$$m_i = [m_{i1}, m_{i2}, \dots m_{in}]^T \in \Re^n$$

$$r_i = [r_{i1}, r_{i2}, \dots r_{in}]^T \in \Re^n$$

The reference vectors mi are defined in the input space and the patterns xj are present to the network. The Euclidean distance is calculated

$$\|x - m_c\| = \min_i\{\|x - m_i\|\}$$

Where *c* is the Best Match Unit.

After this finding, starts the learning where the BMU and its neighbours are modified to get closer to the data pattern in order to represent it better.

$$m_i = m_i + \alpha(t)h_{ci}(t)(x - m_i)$$

α(t) is the learning rate;

h_{ci} is the neighbourhood function centered in *c*.

The training ends after an explicit number of iterations achieved or another stopping criteria being reached.

2.2. Visualization

The SOM's visualization is possible through a set of methods like component planes, distortion patterns, clustering and U-matrix (Skupin & Agarwal, 2008). Although a major set of representations can be found and used (Vesanto, 1999).

The perspectives obtained about the output and input space are directly connected once the output space tries to preserve the topology of the input space (Gorricha & Lobo, 2012; Kohonen, 2013).

The input space interpretation is possible by the use of several kinds of projections as scatter plots, Principal Component Analysis maps and Sammon Maps. Because of the high-dimensionality the application of 2D or 3D plots generally isn't effective that's why PCA and Sammon maps are an useful alternative. Sammon Map is a nonlinear mapping solution where the inherent structure of the data is preserved by maintaining approximately the distances of the projected points (Sammon, 1969). The PCA provides a linear projection by reducing the amount of variables creating new ones (Pearson, 1901).

As referred previously there are other methods to visualize the SOM. The U-Matrix is a solution that shows the clusters and it is the most used method to perform the SOM visualization (Ultsch & Siemon, 1990). The U-matrix uses colors in order to identify the neighbour's distance. Closest units use lighter colors and the distance is performed in dark ones (Kohonen, 1995).

The other option is to use the Component Planes to view each variable individually. Using the same locations units as in the U-matrix, it is possible to analyze the distribution of each variable based on the clustering result (Hajek, Henriques, & Hajkova, 2014).

2.3. GEO-SOM

The data used in our research is the census information, that has the particular feature of includes spatial data or geographic location. Due to this specific attribute, a new approach was developed in order to adapt the traditional SOM to a new one Geo-SOM, where the spatial variables have a different handling. The calculation of the BMU is performed in two phases, a first one where the spatial data is processed to find the nearest neighbour. This phase is carried out based on the k value. Briefly, if k=0 only the nearest unit counts, otherwise the nearest neighbours will expand until k assumes the size of the SOM. The second phase researches for the BMUs based on the remaining attributes (non-spatial data) (Bação et al., 2004).

The algorithm presented in Bação, Lobo, & Painho (2004) is the following:

Let

X be the set of n training patterns **x1, x2,..xn**, each of these having a set of components $\mathbf{geo}_i$ and another set $\mathbf{ngf}_i$.

W be a *pxq* grid of units $\mathbf{w}_{ij}$ where i and j are their coordinates on that grid, and each of these units having a set of components $\mathbf{wgeo}_{ij}$ and another set $\mathbf{wngf}_{ij}$.

α be the learning rate, assuming values in]0,1[, initialized to a given initial learning rate

r be the radius of the neighbourhood function ***h(wij,wmn,r)***, initialized to a given initial radius

k be the radius of the geographical BMU that is to be searched

f be a logical variable that is true if the units are at fixed geographical locations.

```
Repeat
    For m=1 to n
      For all $\mathbf{w}_{ij} \in W$,
        Calculate $d_{ij} = ||\mathbf{wgeo}_k - \mathbf{wgeo}_{ij}||$
        Select the unit that minimizes dij as the geo-winner $w_{winnergeo}$
        Select a set $W_{winner}$ of $\mathbf{w}_{ij}$ such that the distance in the grid between $w_{winnergeo}$ and $w_{ij}$ is smaller or equal to k.
        For all $\mathbf{w}_{ij} \in W_{winner}$,
          calculate $dij = ||\mathbf{x}_k - \mathbf{w}_{ij}||$
          Select the unit that minimizes dij as the winner $W_{winner}$
        If *f* is true, then
          Update each unit $\mathbf{w}_{ij} \in W$: $\mathbf{wngf}_{ij} = \mathbf{wngf}_{ij} + \alpha\, h(\mathbf{wngf}_{winner}, \mathbf{wngf}_{ij}, r)\ ||\mathbf{x}_k - \mathbf{w}_{ij}||$
       Else
          Update each unit $\mathbf{w}_{ij} \in W$: $wij = wij + \alpha\, h(w_{winner}, w_{ij}, r)\ ||x_k - w_{ij}||$
    Decrease the value of α and r
```

15. Until α reaches 0

The advantages of the use of the Geo-SOM algorithm are the flexibility and the exploratory nature (Bação et al., 2004; R. Henriques, Bacao, & Lobo, 2012). The concept of spatial dependency presented by Tobler First Law “everything is related to everything else but near things are more related than distant things” (Tobler, 1970) and its incorporation to the expansion of the SOM abilities revealed an excellent opportunity to manage the geographic and non-geographic data in a way that the relationship between these variables is completely within the skills of an analyst desires.

2.4. GIS VISUALIZATION

The SOM output visualization gives us the first insight of the clustering performed, although, the census deal with data that is spatial dependent which has special characteristics (Anselin, 1989). The geographical visualization merged with the Geo-SOM output brings us to a new opportunity to understand what the results distribution is.

The links between SOM and geographic visualization face successfully the challenge of the input high dimensionality data set. The most common method to present the results is the cartographic representation as maps. There are several types of maps like choropleth maps, dot maps, proportional symbol maps and isopleths maps (R. A. P. Henriques, 2010; Robinson, Morrison, Muehrcke, Kimerling, & Guptill, 1995).

In order to get what really matters, cartographic generalization has been applied to the cartographic elements (Buttenfield & McMaster, 1991; R. A. P. Henriques, 2010; Robinson et al., 1995). Generally the application of SOM could be over the location or in a more widespread circumstances, a context based visualization. Both use SOM to mitigate the high dimensionality of the data, but location visualization focus on the degree of abstraction in order to fit the desired map scale (R. A. P. Henriques, 2010), the typification work is an example (Allouche & Moulin, 2005; Sester, 2005). The Context visualization aims to include the non-spatial high-dimensionality data. The result is a cartographic representation of multivariable groups depending in what knowledge is desired to obtain (Etien L Koua & Kraak, 2004; Penn, 2005; Skupin & Hagelman, 2005; Skupin, 2002).

Based on the previous works the link between the Geo-SOM outputs with the GIS maps, using for example colors, dots or similar allows an easy way to identify the obtained clusters based on the segment size output (Countries, municipal parish, etc.). These methodology has been used in several projects and researches. Bação et al.(2004) used Geo-SOM with GIS maps to represent the small census data. R. Henriques et al.(2012) applied similar technique to the Lisbon Metropolitan Area and Squareville dataset where it possible to identify the patterns, component planes, neurons distribution among other features. Gorricha & Lobo(2012) used georeferenced data to produce a 3D SOM where the borders width were used to explore the distances between elements and Best Matching Units. Lee & Rinner (2014) identify spatio-temporal patterns over Toronto area using some of the previous features.

3. DATA AND SOFTWARE

"The population and housing census represents one of the pillars for the data collection on the number and characteristics of the population of a country" (United Nations Economic Commission for Europe, 2006). The Portuguese Population Census was performed by the first time according to the international standards in 1864, and the Portuguese Housing Census was performed by the first time in 1970 (Statistics Portugal, 2014). The population and housing census must be in conformity with the following features: Individual Enumeration, Simultaneity, Universality, Small-area data and Defined periodicity (United Nations Economic Commission for Europe, 2006). Based on the quality of this data source, the study focused on the information extracted from the 2001 and 2011 censuses. The data was collected from the institutional Statistics Portugal web site, 2001 and 2011 files.

As referred previously, to georeference the inquiries datasets the cartography is essential to determinate the global position. In order to fulfill this requirement it will be used the Portuguese "Based Geographical Referencing of Information" (BGRI).

The BGRI is a geographic referencing system which divides the parish areas in smaller statistical territorial units like statistical section, sub statistical section and place/hamlet (Geirinhas, 2001; Statistics Portugal, 2013).

- Section Statistics - territorial unit corresponding to a continuous area of the parish, with about 300 apartments, for housing.
- Subsection Stats - Territorial unit that identifies the smallest homogenous construction area or not, existing within the statistical section. It corresponds to the block in urban areas, the place or of the place in rural areas or to residual areas that may or may not contain statistical units (isolated).
- Place - population cluster with 10 or more dwellings for housing people and with its own name, regardless of belonging to one or more parishes.

Once the Place could belong to different parishes, it will not be used in the present work. The subsection Stats will be used over Section Statistics because it represents the smallest area (block level), increasing the available study resolution.

The principal attributes used in this work are Building, Family Accommodation, Classic Family and Resident Individual. From these, censuses provided us an enormous subset of variables, obtained directly from the census or by processing.

The dataset refers to the Lisbon County and includes 3623 Enumeration Districts and 122 original variables in 2011 that are able to characterize social, demographic and economical the applied areas. Related to 2001, there are 4390 Enumeration Districts and 99 original variables. As can be perceived, the census basis has changed as result of spatio-temporal mutations over the selected areas. In order to accomplish consistent results the original dataset suffered an exhaustive pre-process. The goal is to reduce the influence of different population and housing sizes, as well include a sort of ratios that increment the ability of analysis and ease the clustering description. The establishment of relations between variables is part of the scientific practices (Fink, 2009) as well the ability to avoid the high dimensionality curse (Donoho, 2000), the vector dimension was reduced to the attributes that could be used to explain the results and to decrease the occurrence of spurious relations.

The variables used, after the processing phase, are described in the next table:

Variable Category	Index	Variable Formula	Variable Explanation
Buildings	**A**	$\frac{\text{Number of buildings above 4 floors}}{\text{Total number of buildings}}$	Percentage of buildings with more than 4 floors
	B	$\frac{\text{Number of buildings built before 1980}}{\text{Total number of buildings}}$	Percentage of buildings built before 1980
	C	$\frac{\text{Number of buildings built between 1981 \& 2000}}{\text{Total number of buildings}}$	Percentage of buildings built between 1981 and 2000
	D	$\frac{\text{Total number of Accomodations}}{\text{Area}}$	Number of Accommodations per area
Family Accommodation	**E**	$\frac{\text{Total Free accomodations}}{\text{Total Accomodations}}$	Percentage of free accommodations
	F	$\frac{\text{Number of Rented accomodations}}{\text{Total Accomodations}}$	Percentage of rented residences
Classic Family	**G**	$\frac{\text{Number of families above 4 members}}{\text{Total number of families}}$	Percentage of families with more than 4 members
Resident Individual	**H**	$\frac{\text{Number of Resident individuals aged below 20}}{\text{Total number of resident individuals}}$	Percentage of residents aged between 0 and 19
	I	$\frac{\text{Number of Resident individuals aged between 20 64}}{\text{Total number of resident individuals}}$	Percentage of residents aged between 20 and 64
	J	$\frac{\text{Number of Resident individuals aged above 64}}{\text{Total number of resident individuals}}$	Percentage of residents aged above 64
	K	$\frac{\text{Number of Resident individuals with primary education}}{\text{Total number of resident individuals}}$	Percentage of residents with Elementary Education
	L	$\frac{\text{Number of Resident individuals with secundary education}}{\text{Total number of resident individuals}}$	Percentage of residents with Secondary Education
	M	$\frac{\text{Number of Resident individuals with tertiary education}}{\text{Total number of resident individuals}}$	Percentage of residents with Tertiary

			Education
	N	$\frac{Number\ of\ unemployed\ Resident\ individuals}{Total\ number\ of\ resident\ individuals}$	Percentage of unemployed residents
	O	$\frac{Number\ of\ Resident\ individuals}{Area}$	Number of residents per area

Table 3-1 Pre-Process Variables

The actual variables were obtained after a deep iterative and analysis process. The majority of them represent a subset of an attribute in relation with the total amount (Classical Percentage). Others represent the population and buildings density.

In order to better understand the city, we must be able to map properly the data extracted from the census. The information of land cover it's extremely useful in these cases. There are two main available datasets, the CORINE Land Cover and the Portuguese COS2007. The selected source to be used in the present study was the COS2007 level 2, once it has a smaller mapping unit (1ha vs 25ha) and the subsequent levels fit the purpose.

In order to allocate the population and the buildings, the included areas are the ones which represent urban tissue. Based on the recent years, the Lisbon urban tissue didn't suffer significant changes from 2007 to 2011, when the census were performed. Although, the present research processes always depict the relation between the BGRI2011 and COS2007.

We used the GeoSOM suite tool which allowed a set of operations like training Self-organizing maps using the standard SOM or the GeoSOM algorithm and produce several representations of the input and output data. This tool is implemented in Matlab and uses the public domain SOM toolbox (R. Henriques et al., 2012).

During the pre and pro processing operations, some data had to be manipulated, in order to do it in a faster way, we used R which is an open-source project language widely used to perform data analysis tasks (Peng, 2015).

After the constant iterations and adjustments needed to obtain generous quality SOM and GeoSOM, the visualization of the clustering outcome is a critical step once it will be the responsible by the human knowledge interpretation. The fulfil of these activity was done using QGIS Desktop 2.8.1 and ArcMap 10.3.

4. METHODOLOGY

As referred previously the study area is the Lisbon municipality extracted from the governmental institutional websites.

As explained, the census subsections have changed from 2001 to 2011, in order to improve the information quality obtained from the inquiry. One of the study goals is to address the mutations occurred in the presented decennium timeframe in the Lisbon County (which haven't changed its own borders).

First we must merge the COS2007 with the BGRI to obtain a real distribution of the population and buildings (Urban fabric) excluding all the other areas, namely:

- Industry, business and transport
- Areas of inert extraction, waste disposal and construction sites
- Urban green spaces, sports, cultural and leisure equipment, and historic areas
- Temporary crops
- Permanent crops
- Permanent pasture
- Heterogeneous Agricultural areas
- Forests
- Open forests, shrub and herbaceous vegetation areas
- Open areas with little vegetation
- Interior Wetlands
- Coastal Wetlands
- Inland waters
- Marine and coastal waters

Keeping the desired areas enables a better portray of the assessed county.

4.1. SHAPEFILE

The process of getting a final dataset to be used as input to the SOM and Geo-SOM algorithms is expressed in the following picture flow.

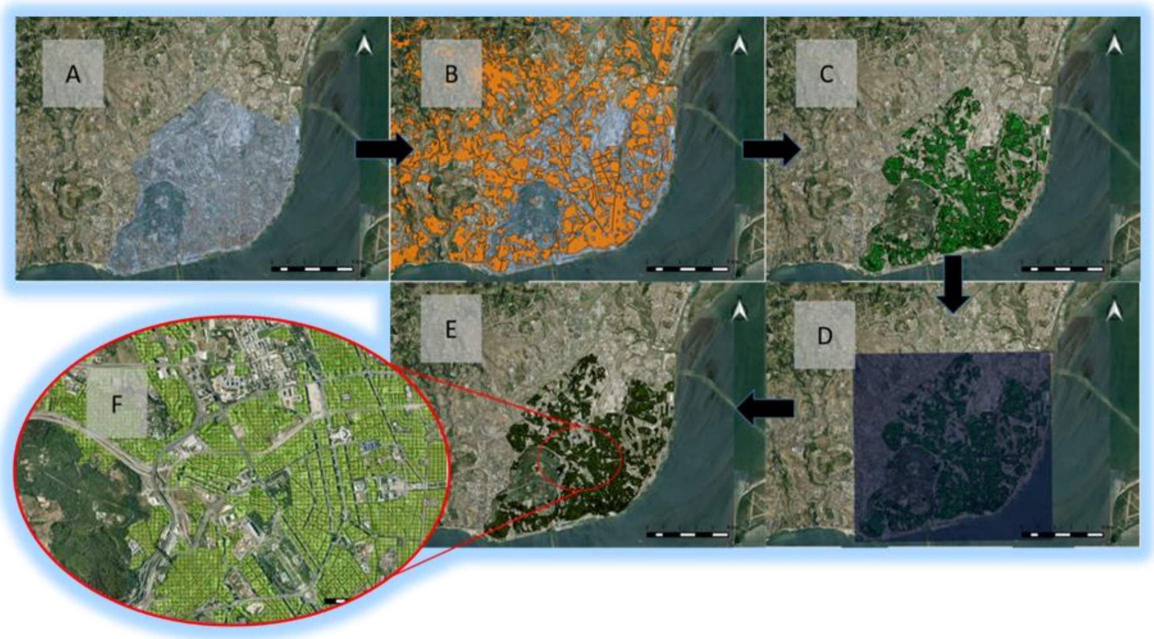


Figure 4-1 Intersection Methodology Choropleth

1. Figure 4.A – The Lisbon BGRI shapefile is loaded (light blue)
2. Figure 4.B – The Portuguese COS2007 N2 is loaded with only the Urban Fabric polygons (orange)
3. Figure 4.C – we performed an intersection between Lisbon BGRI and COS 2007 N2 resulting in a shapefile constituted only with Lisbon County Urban Areas (dark green).
4. Figure 4.D – In order to perform comparison with 2001 and 2011 databases, we create a grid to allocate properly the population and buildings in the same locations
5. Figure 4.E – Each specific urban area is now represented by a “pixel” that doesn’t change location or boundaries over time
6. Figure 4.F – Zooming a specific Lisbon County area

It is possible to see in the Figure 4.F, the Monsanto Forest Area (in left) isn’t included in the urban fabric, as well other areas, like stadiums, green parks, university campus among others. The obtained subset fits our needs, once it enables a more accurate distribution of the used features.

The attribute values distribution among the generated pixel's was performed recurring to the proportional areas. We present a dummy example to demonstrate this process.

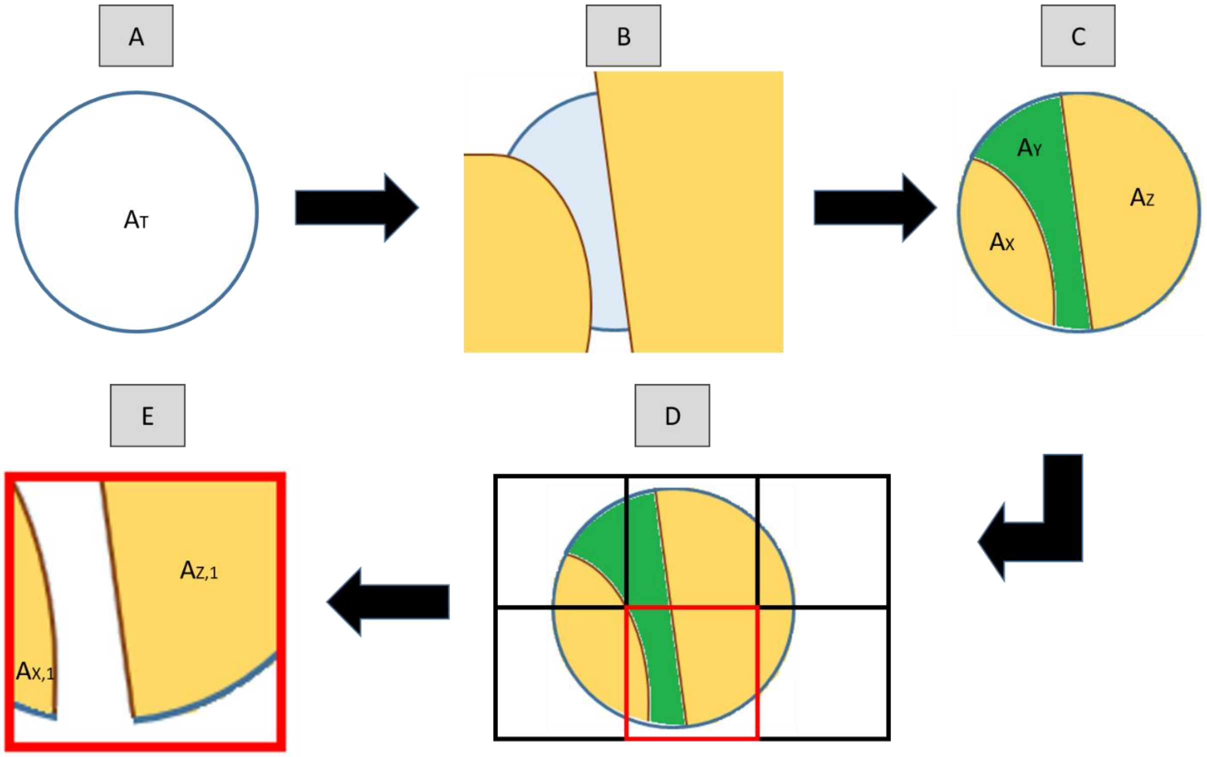


Figure 4-2 Intersection Methodology Process

The previous picture shows the flow to obtain the BGRI relative areas inside a pixel.

For example, imagine we use the individual residents to create a new distribution on the final grid.

The Figure 5.A shows a subsection BGRI polygon (SBP) with the Area A_T and number of Individuals I_T.

Performing the intersection (Figure 5.B) between the SBP and the Urban Fabric (from COS2007N2), the resulting is presented in Figure 5.C. A green area (A_Y) is created representing a green park (no population allocated) is created along with two new areas with population (A_x and A_z)

The percentages areas are the following:

- A_X = 25 %
- A_Y = 25 %
- A_Z = 50 %

This means that the new considered area is:

$$Area_{new} = A_X + A_Z = A_T \text{ x } 0.25 + A_T \text{ x } 0.50$$

Since the EDs from 2001 and 2011 don't have the same spatial delimitation, we created a grid over the urban fabric to map the changes occurred in that period (Figure 5.D)

Now, we've a new Area that will be distributed by the pixels. Each pixel will include the areas of one, or more, SBPs.

In this case, the red Pixel includes two Areas ($A_{X,1}$ and $A_{Z,1}$) that represent the proportion inside the pixel from the new SBP.

Let's assume that the percentages to the new SBP area are the following:

- $A_{X,1}$ = 5% =>0.05 x $Area_{new}$
- $A_{Z,1}$ = 20% => 0.2 x $Area_{new}$

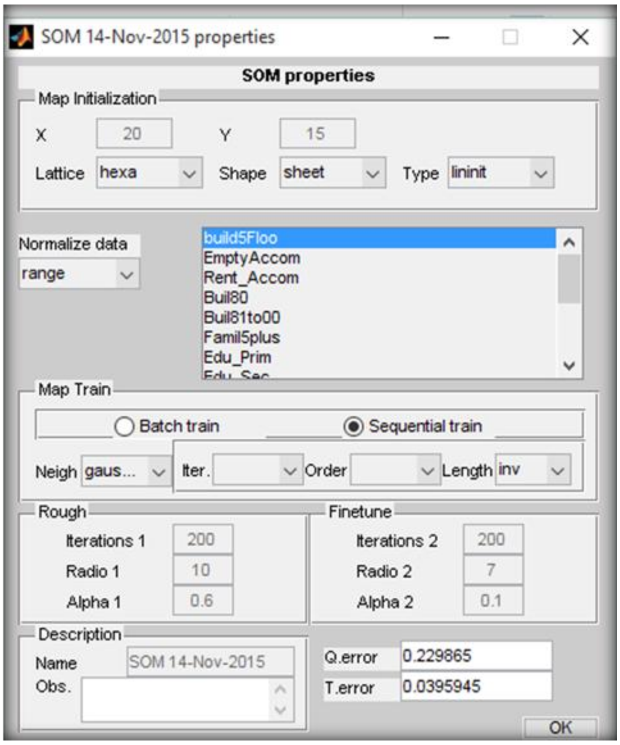
The population allocated to this red Pixel is the following:

$$Ind_{pixel} = \sum_{BGRI=j}^{m} \sum_{Area=i}^{n} I_T \times \frac{Area_i}{Area_{newSBP}}$$

The issues that raise from this data transformation are already known as MAUP (modifiable areal unit problem). Using again the example provided, the population were better allocated in 75% of the total area than in 100%, although the final distribution by each pixel assumes that the attribute values spreading is uniform.

4.2. Neural Network

The resulting shapefile, with the attributes previously described in the Data and Software section, was imported to the Geo-SOM Suite were the configuration process of the algorithm takes place.



SOM 14-Nov-2015 properties

SOM properties

Map Initialization

X 20 Y 15

Lattice hexa Shape sheet Type lininit

Normalize data

range

build5Floo
EmptyAccom
Rent_Accom
Buil80
Buil81to00
Famil5plus
Edu_Prim
Edu_Sec

Map Train

○ Batch train ● Sequential train

Neigh gaus... Iter. Order Length inv

Rough

Iterations 1 200

Radio 1 10

Alpha 1 0.6

Finetune

Iterations 2 200

Radio 2 7

Alpha 2 0.1

Description

Name SOM 14-Nov-2015

Obs.

Q.error 0.229865

T.error 0.0395945

OK

Figure 4-3 SOM Suite Parameters Display

The chosen parameters aimed to use the best characteristics and features of the Kohonen's map that fit our study object.

4.2.1. Map

The decision of the map size was based on the results obtained in the work performed by Bação, Lobo, & Painho (2005a). Selecting a wider map allows a better distribution of units per neurons as well increases the interpretability of the grid distances in order to identify the present clusters. Although, an extreme large map will bring a conundrum, once what we gain in reliability, it is lost in interpretability.

Based on the previous premises, a grid of 20 x 15 was used, representing around 10% of the total analyzed population (of subsections).

There are some varieties of Network architectures to run on SOM. Regular or Cyclic arrays and growing networks are the common available choices (Kohonen, 2013) The recommended architecture taking in consideration simplicity, illustrative and accuracy is the hexagonal array. The hexagonal neurons can have six neighbors with the same Euclidean distance which is an advantaged (Kohonen, 1995).

4.2.2. Train

The obtained data attributes (present in the shapefile) have different ranges of values, without scaling, it is possible that some variables weighing more leading to an increase of their relative importance. . So a standardization was performed too, through a score-range transformation where the variables will assume values between [0,1]. This operation allows the equalization of the influence among all attributes (Lee & Rinner, 2014). This operation of normalization is also known as Min-Max and is calculated as (Vesanto, 1999):

$$Normalized(x_i) = \frac{x_i - X_{min}}{X_{max} - X_{min}}$$

The learning process performed by the algorithm, also depends on the neighbourhood function. In order to have a network learning process where all/part of the neurons are influenced by the units, we selected the Gaussian function:

$$\exp\left(-\frac{||r_c - r_i||^2}{2\sigma^2(t)}\right)$$

The iterations selected were obtained iteratively until the moment where adding more iterations or changes in the applied parameters didn’t add value or decreased the obtained errors.

The resulting U-Matrix has a large number of neurons and implicitly a major difficult to understand which are the general characteristics applied to each cluster.

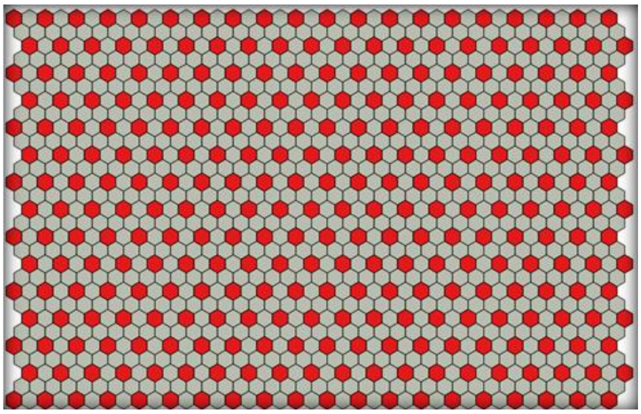


Figure 4-4 U-Matrix 20x15

The interpretability issue presented by selected map size was solved introducing in the process flow a known clustering technique, hierarchical clustering.

The hierarchical clustering is depicted in trees or dendrograms, nesting partitions but not deterministic, once no random initial conditions are given, except the method itself.

The hierarchical clustering method selected was Ward. This method measures the distance between two clusters (or units), looking for the increasing sum of squares or also known as the merging cost.

Assuming two clusters, A and B,

$$\Delta(A,B) = \sum_{i \in A \cup B} ||\vec{x}_i - \vec{m}_{A \cup B}||^2 - \sum_{i \in A} ||\vec{x}_i - \vec{m}_A||^2 - \sum_{i \in B} ||\vec{x}_i - \vec{m}_B||^2 = \frac{n_A n_B}{n_A + n_B} ||\vec{m}_A - \vec{m}_B||^2$$

Where:

- $\vec{m}_j$ is the center of cluster *j*
- n_j is the number of points in *j*

and

- Δ is the merging cost

With the previous method, the sum of squares starts at zero (the cluster is the unit itself), and then grows maintaining the merging cost Δ as small as possible.

The applied clustering technique had been studied by the academia for a while (Jain et al., 1999; Steinbach, Karypis, & Kumar, 2000). The demonstrated advantages fitted our purpose, resulting in a manner way to aggregate the obtained units from the neural network.

Finally, the resulting cluster is described by the taken average from all the input grid elements to be mapped and used to spatial-temporal comparisons.

In order to perform an accurate comparison between the two datasets (2001 and 2011), we’ve done a classification of the 2001 elements based on the resulting hierarchical clusters obtained from the 2011 study process.

The SOM classification was done excluding the geo-location attributes, while the Geo-SOM classification was done using the geo-location attributes, although they were considered as another variable.

The classification was done using the Euclidean distance among the 2001 elements and the obtained units (neurons) from the 2011 dataset. In a certain point we can say the 2001 element looks for the Best Matching Unit without change the network.

As explained previously, the Geo-SOM algorithm includes a vicinity property (*k* neighbourhood). The selection of the representative Geo-SOM with the appropriate *k* was done through a square error quantization. The error calculation only took into account the non-location attributes, once they are the ones which depict the core urban area.

The remaining parameters like location update and learning rate are part of the initial conditions settled by the analyst that were tuned during the iterative process.

5. RESULTS

The study aims to explore and depict the Lisbon County based on the input variables and the applied data mining and exploratory techniques.

5.1. INPUT VARIABLES

The input variables are a great and first source of information to be collected and explored. We will focus our study in two main areas, the City Historic center and the "new Lisbon" located at the east part of the county where was located the EXPO'98, giving place to Parque das Nações. The remaining areas are presented but in the attachment's section, as well other input not used to explain the major changes in these two interest areas.

Figure 5-1 City Historic Center View

Figure 5-2 Parque das Nações View

5.1.1. City Historic Center

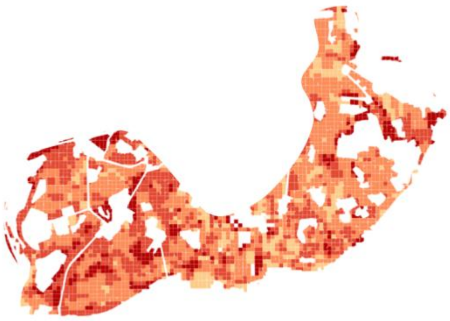
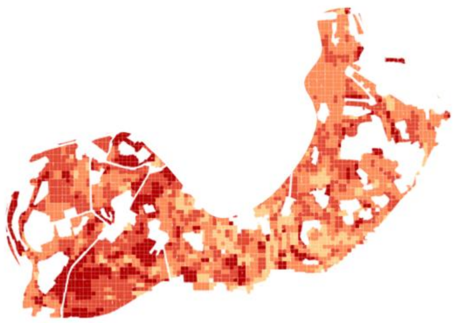
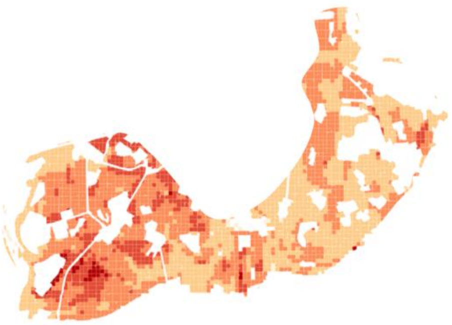
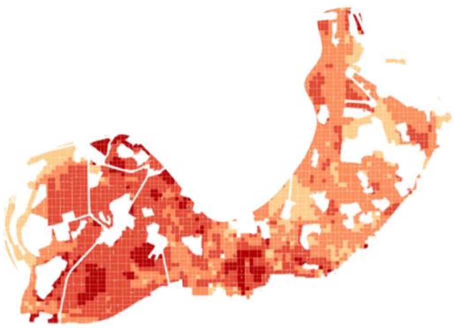
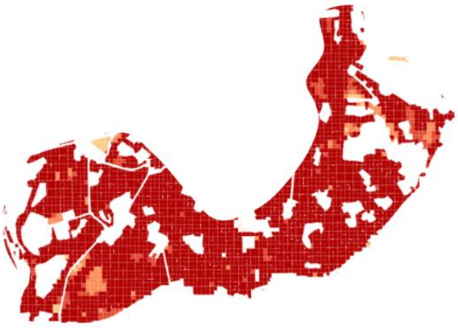
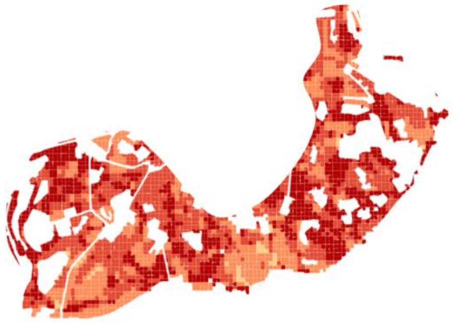
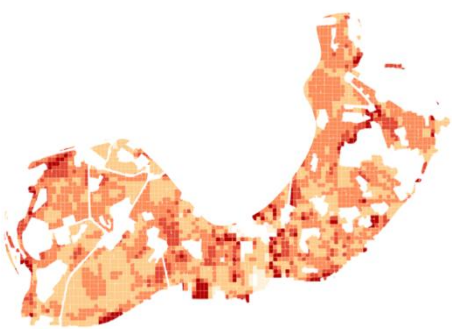
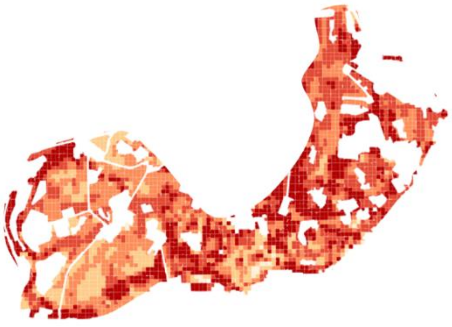
Variable	2001	2011	Scale
Percentage of residents aged between 0 and 19			0,00 0,00 – 0,10 0,10 – 0,15 0,15 – 0,20 0,20 – 1,00
Percentage of residents with Tertiary Education			0,00 0,00 – 0,15 0,15 – 0,30 0,30 – 0,50 0,50 – 1,00
Percentage of rented residences			0,00 0,00 – 0,10 0,10 – 0,30 0,30 – 0,40 0,40 – 1,00
Percentage of unemployed residents			0,00 0,00 – 0,03 0,03– 0,05 0,05 – 0,07 0,07 – 1,00

Table 5-1 City Historic Center Variables Comparison

The previous table doesn't show all the variables, but provides a first insight about the mutations suffered in the city center. It was selected only one "age" variable, but in general it wasn't identifiable major changes related with residents age feature. Although, other variables like education, property ownership and unemployment rate revealed a significant mutation in the last decade. Curiously the tertiary education is an excellent reverse indicator on the other two. The identifiable pattern shows that, areas where tertiary educated residents rate is higher tend to have lower rent accommodation rates and unemployment rate also lower. A variable that shows the impact of the economic trends during the past recent years, it's the owner / rented accommodation. The city center is getting people how aim to live there as their own house representing a paradigm change, once Lisbon has shown in the recent years a higher tourism demand, which requires free beds to receive the visitors.

5.1.2. Parque das Nações (East)

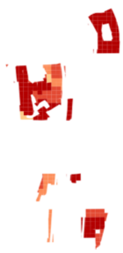
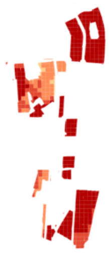
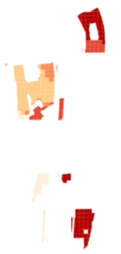
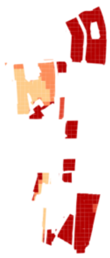
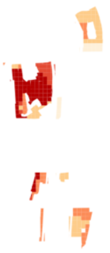
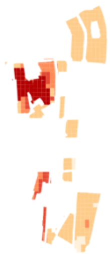
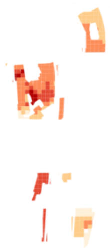
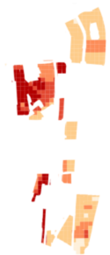
Variable	2001	2011	Scale
Percentage of residents aged between 0 and 19			0,00 0,00 – 0,10 0,10 – 0,15 0,15 – 0,20 0,20 – 1,00
Percentage of residents with Tertiary Education			0,00 0,00 – 0,15 0,15 – 0,30 0,30 – 0,50 0,50 – 1,00
Percentage of rented residences			0,00 0,00 – 0,10 0,10 – 0,30 0,30 – 0,40 0,40 – 1,00
Percentage of unemployed residents			0,00 0,00 – 0,03 0,03– 0,05 0,05 – 0,07 0,07 – 1,00

Table 5-2 Parque das Nações Variables Comparison

The new Parque das Nações located at the East part of the County showed a different behavior over the last decade. Besides an increment in buildings and population, the general characterization (upper class area) remain the same. Only the unemployment rate increases showing the impact of the crisis.

The presented results based on four variables aimed to show the importance of this exploratory technique, although the inclusion at the same time all the used variables could lead to a surplus of information. In order to overcome this limitation the results of the clustering and dimension reduction techniques are presented further.

5.2. Clustering

The applied techniques to the study object were SOM, Geo-SOM and Hierarchical clustering, the last one used in the resulting network units.

5.2.1. SOM

The SOM method was applied to the non-location attributes.

The obtained results were the following:

5.2.1.1. Component Planes

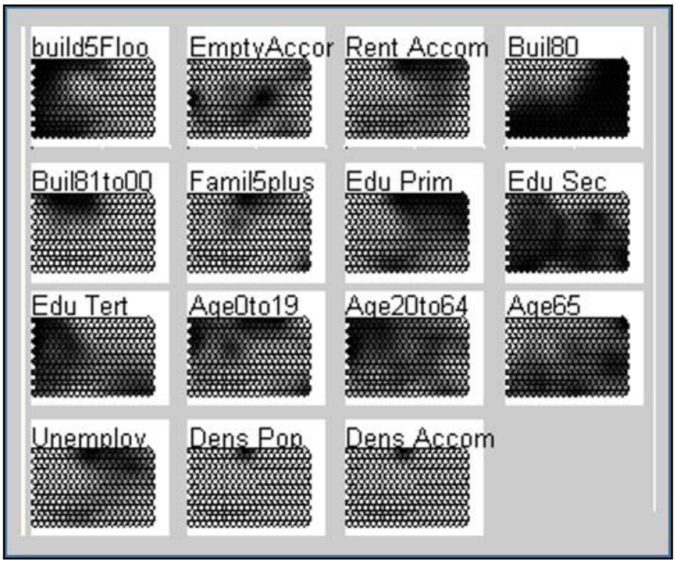


Figure 5-3 SOM Component Planes

The component planes provide a first look of what will be the final results. The values distribution for each component will be translated in a U-Matrix were it will be easier to identify the patterns in a 2D visualization.

5.2.1.2. U-Matrix

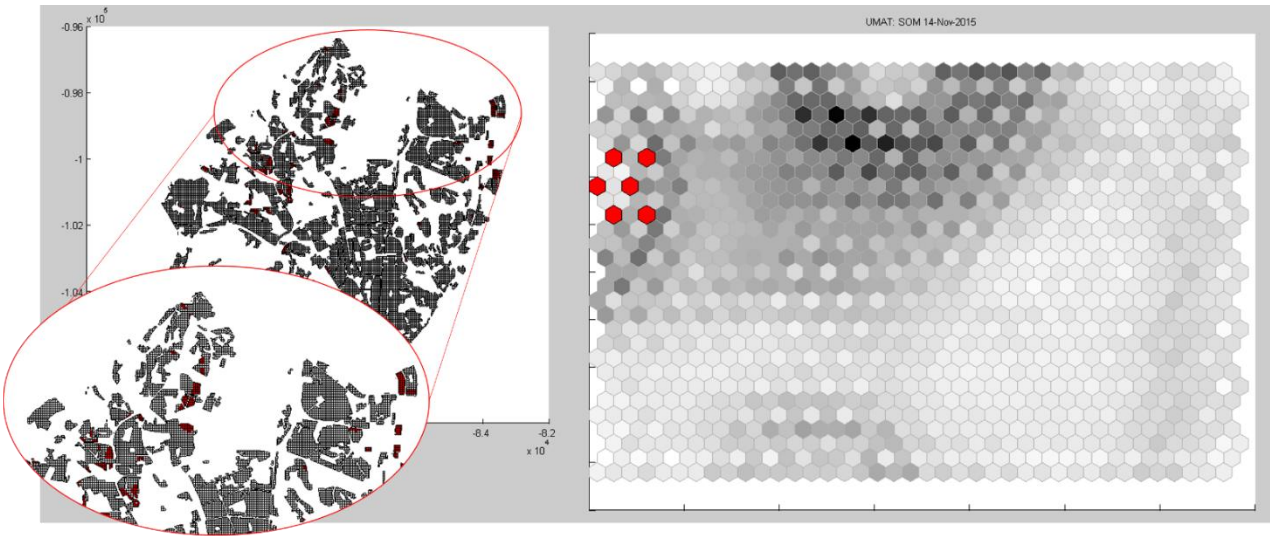


Figure 5-4 SOM U-Matrix and Map View

The obtained U-Matrix (20x15) reveals the existence of some clusters with different sizes. Also it was possible to locate the Best Matching Units with the elements in the Lisbon County urban selection map. For example, after the selection of 6 units (red hexagons in the U-Matrix at right) and they appear selected in red also in the map (left).

This feature enables a fast evaluation of the accuracy of the results, once the analyst can infer easily if the together units correspond to an expected output.

Although the 300 units obtained with the SOM represent a wide variety of clusters. It was used the hierarchical clustering as merging technique to obtain the higher level of clusters.

5.2.1.3. Hierarchical Clustering

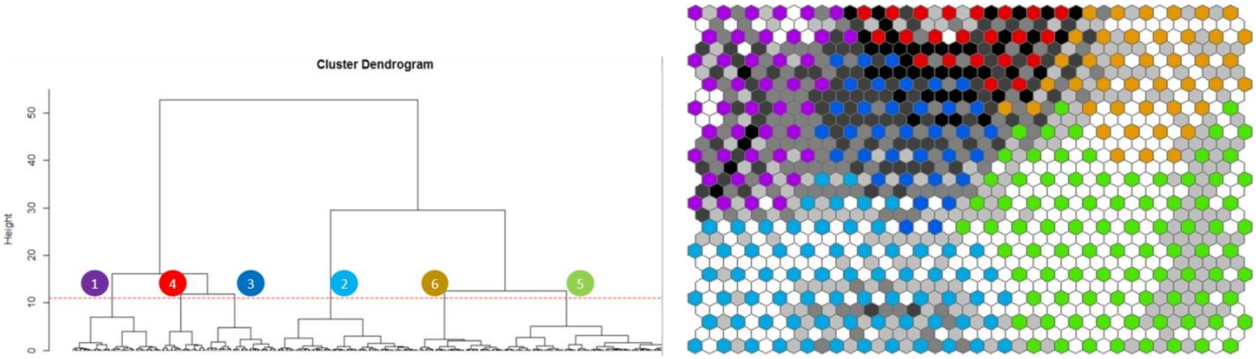


Figure 5-5 Hierarchical clustering and U-Matrix

The decision to cut at the "height = 11" was based on the ability to cluster enough to get interpretability without remove the "special" characteristics that each cluster should have to differentiate them. The resulting U-Matrix with the new six clusters match is presented with the matching colors/numbers between the Hierarchical clusters (HC) and the units.

For each HC was calculated the centroid in order to understand and describe the cluster for which element belongs.

Cluster (Name/Number)	Description
Upper Class BT - 1	Newer and Taller Buildings Highly occupied accommodations by the owners Population Highly Educated with a lower unemployment rate Population tend to be young and active
Upper Class BS - 3	Older Buildings, mainly occupied by the owners Families tend to bigger (more than 4 members) Population Highly Educated
Middle Class BT - 2	Older and Taller Buildings Families tend to smaller (few members) Population tend to be Highly Educated, but old
Middle Class BS - 5	Older and short Buildings Families tend to be smaller (few members) Population tend to have average Education Old Population (low density)
Middle Lower Class BS - 6	Older and short Buildings, with a high rent rate Population tend to have lower Education level with high unemployment rate Population age fits the average (low density)
Lower Class - 4	High density of population and accommodations in newer buildings, with the highest rent rate Large families Lower education level with the highest unemployment rate Population is the youngest

Table 5-3 SOM Hierarchical cluster description[1]

The cluster depict enables us to create an idea about what kind of population are we speaking about. Map the clusters is one of the features earlier presented by the GIS tools. We've mapped both datasets 2001 and 2011.

[1] BS – Buildings short (less than four floors); BT – Buildings Tall (five or more floors)

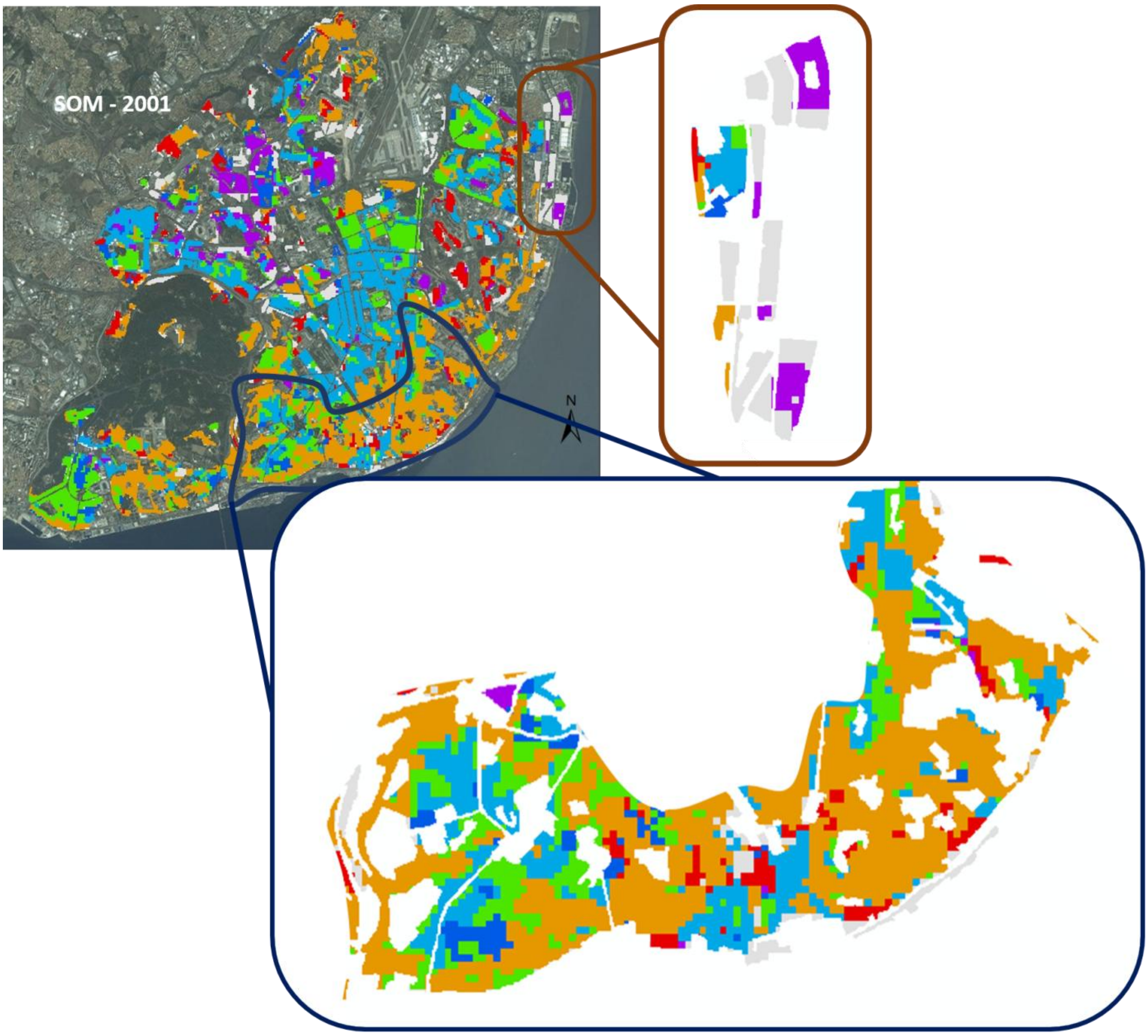


Figure 5-6 SOM 2001 Choropleth

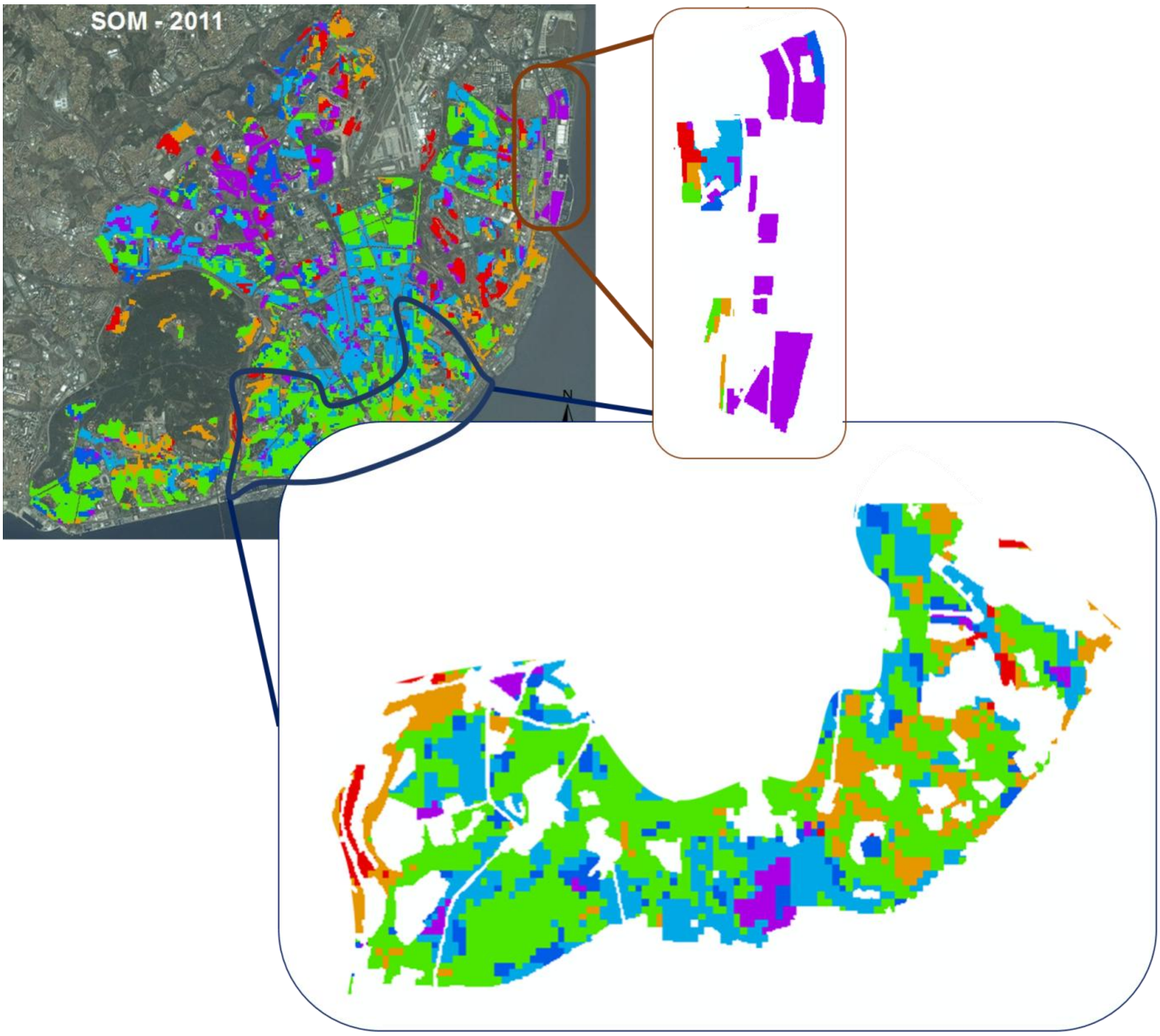


Figure 5-7 SOM 2011 Choropleth

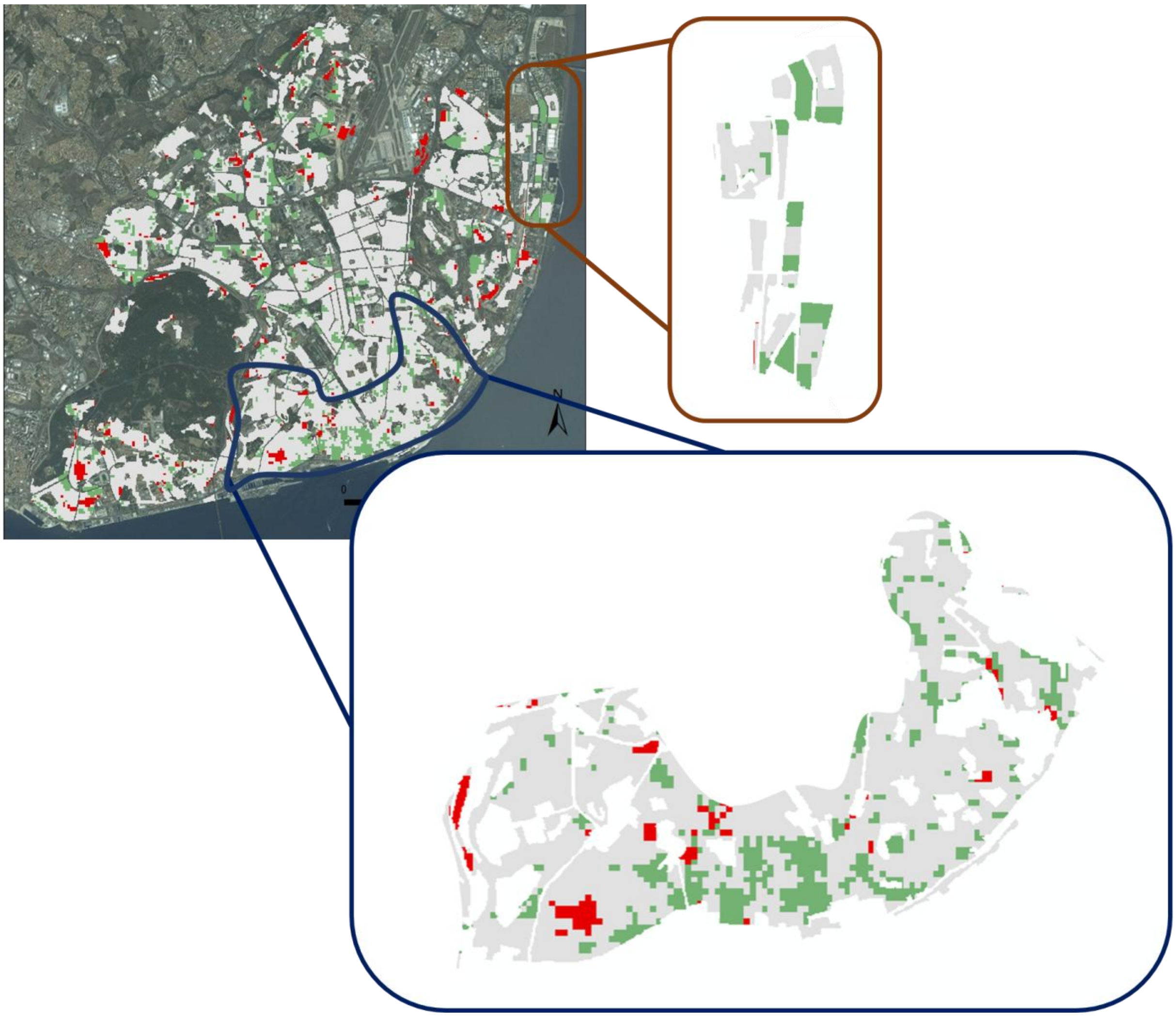


Figure 5-8 SOM Comparison 2001 versus 2011 Choropleth

The choropleth maps showing the HCs distribution, 2001 and 2011 showed how the population is organized and structured in the Lisbon County. The reason why we apply the Hierarchical Clustering over the BMUs instead of the items themselves was Hierarchical clustering doesn't produce good results in big datasets and the final dendrogram isn't easy to understand due to the huge amount of height links (Guha, Rastogi, & Shim, 1998).

The geolocation of the clusters is heterogeneous, meaning that urban fabric didn't follow a precise pattern when growing, specifically related with the non-geographic features.

Looking to the clusters present in the zoomed areas, we identify particularly that the most active clusters in the city center reveal an old population living in older buildings as it was supposed. In the east part of the county, specifically near shore (Parque das Nações) the defined cluster as Upper Class takes care of the all area.

The last choropleth aims to show how the city mutate over the decennium (2001 to 2011) and shows in fact that the city center is getting clear spots of improvement, represented by the green color.

5.2.2. Geo-SOM

The previous method, SOM didn't include the spatial dependency defined by the TFL. In order to solve it, we used the Geo-SOM algorithm.

As explained earlier in the paper, the Geo-SOM has neighborhood feature which defines the threshold of searching for the Best Matching Unit.

In order to perform an accurate and reliable research with the Geo-SOM we run the algorithm for vicinities between 0 and 4, and process the square errors to ascertain the quality of the produced final outputs.

The error table is the following:

Cluster	K=0	K=1	K=2	K=3	K=4	SOM
1	425	22245	917	1542	853	700
2	1175	1104	597	520	1343	808
3	895	634	1443	745	548	578
4	1212	1235	22650	22283	22111	22143
5	1632	1047	872	587	1024	777
6	23632	833	802	1181	627	289
Total	28970	27097	27281	26859	26508	25295
Error %	1,00	0,94	0,94	0,93	0,92	0,87

Table 5-4 Geo-SOM Square Error

The error was performed based only in the non-geolocation attributes. As expected as we increase "freedom" to the network the error decreased, although the relations that are only explained by the proximity are lost. Following this stream of thought we chose the k=3 (G3). The selection of this specific neighborhood attribute value as algorithm representative to this research was based on the premise that the smaller error (bigger k), like SOM will bring a too heterogeneous map, and the looking for the closer neighbor could lead to misinformation.

5.2.2.1. Geo-Units

The Geo-Units were obtained from the U-Matrix. The Geo-Unit is the lower level cluster obtained in this study task.

In order to better understand their distribution, we present the resulting U-Matrix.

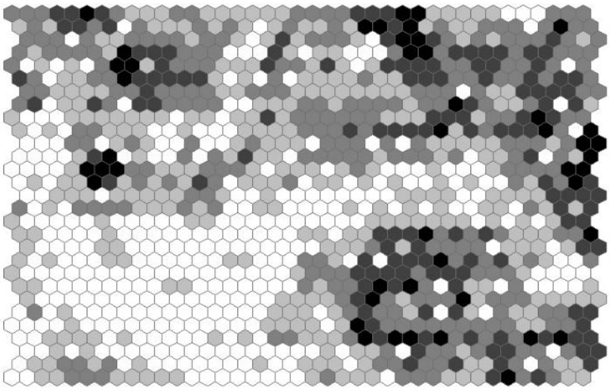


Figure 5-9 Geo-SOM k=3 U-Matrix

The neurons distribution is quite complex, instead of the component planes, we map a sort of variables in order to clarify how the Geo-Units are distributed.

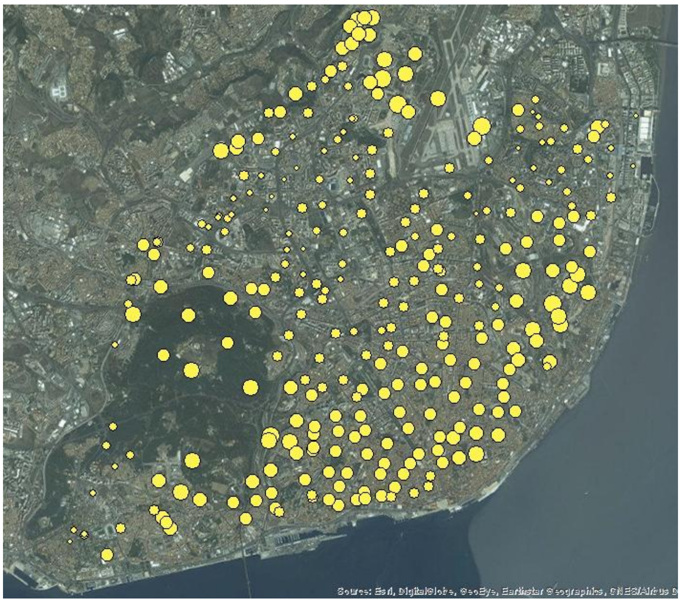

Source: Esri, DigitalGlobe, GeoEye, Earthstar Geographics, CNES/Airbus DS

Figure 5-10 Geo-Units Rent Rate Distribution

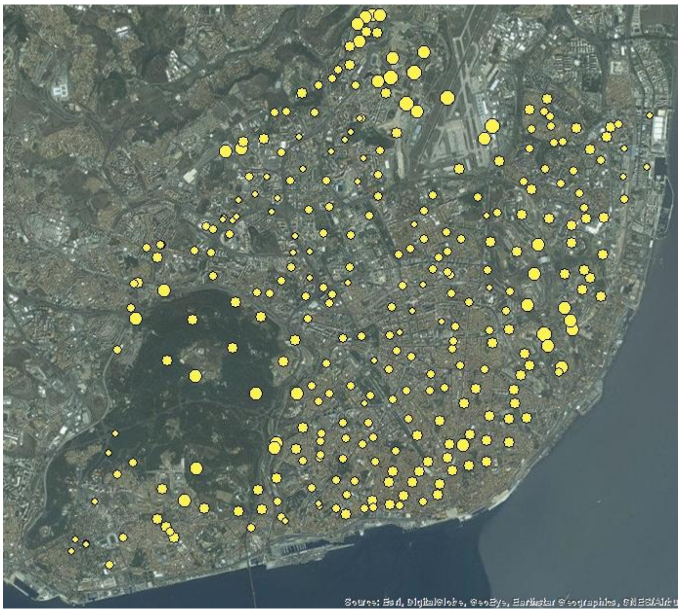

Sources: Esri, DigitalGlobe, GeoEye, Earthstar Geographics, CNES/Airbu

Figure 5-11 Geo-Units Unemployment Rate Distribution

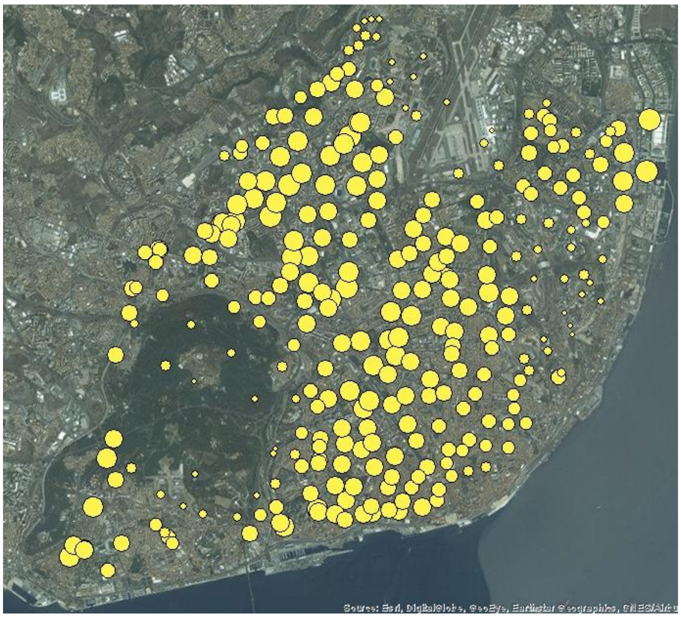
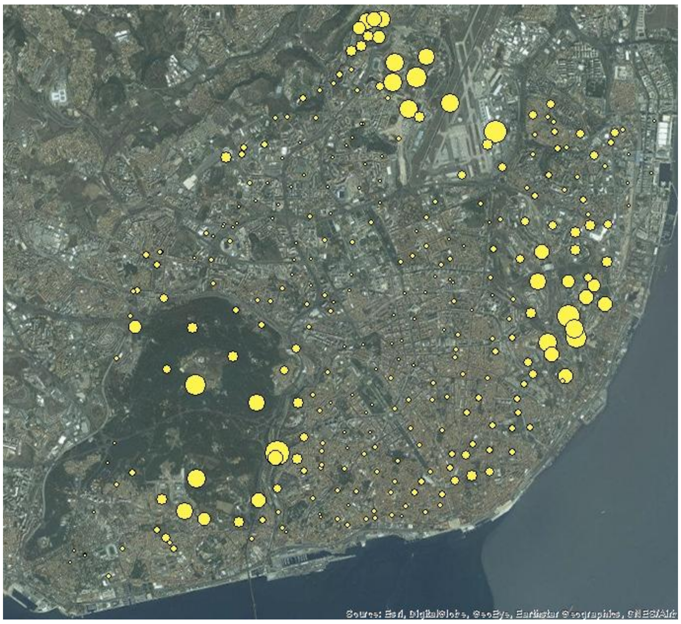

Sources: Esri, DigitalGlobe, GeoEye, Earthstar Geographics, CNES/Airbu

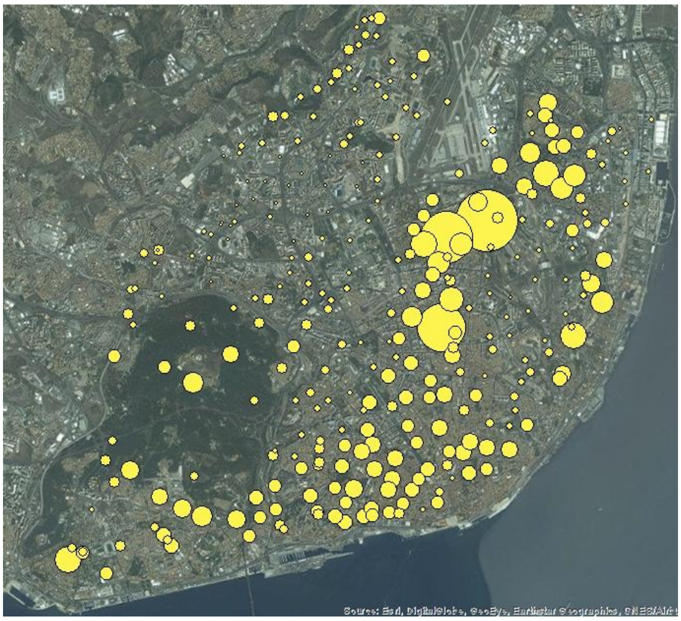
Figure 5-12 Geo-Units Education Tertiary Rate Distribution



Source: Esri, DigitalGlobe, GeoEye, Earthstar Geographics, CNES/Air

Figure 5-13 Geo-Units Unemployment Rate per Tertiary Education Rate Ratio

Figure 5-14 Geo-Units Age 0 to 19 per Builds before 80 Ratio



Source: Esri, DigitalGlobe, GeoEye, Earthstar Geographics, CNES/Air

Figure 5-15 Geo-Units Families above 4 members per Builds 1981 to 2000 Ratio

The presented variables and ratios among the Geo-Units showed some patterns like higher the education level (Tertiary) lower will be the Rent rate as well the unemployment. This is particularly obvious when we look closer to the Parque das Nações. This area also shows bigger numeric ratios with the other relations we presented (Age 0 to 19 / Buildings before 80; Families bigger than 4 members / Buildings from 81 to 2000).

Once again, the exploratory process of the variables alone provides an excellent insight but increases the difficulty to generalize to the all area. In that way we process in the same way and hierarchical cluster the resulting Geo-Units.

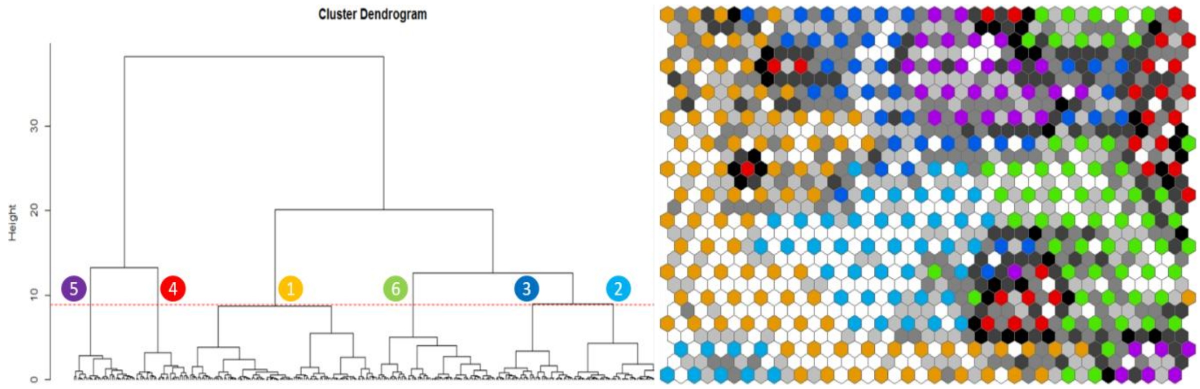


Figure 5-16 Geo-SOM k=3 Hierarchical clustering and U-Matrix

In the resulting U-Matrix is possible to see final clusters disconnected. This evidence shows that the inherit characteristics of the cluster could be geographically separated, meaning that the non-location attributes are similar but apart, although there is a clear effort from the algorithm to keep together the near areas.

The choropleth representation of the Hierarchical clusters for both datasets (2001 and 2011 is present in the following pictures:

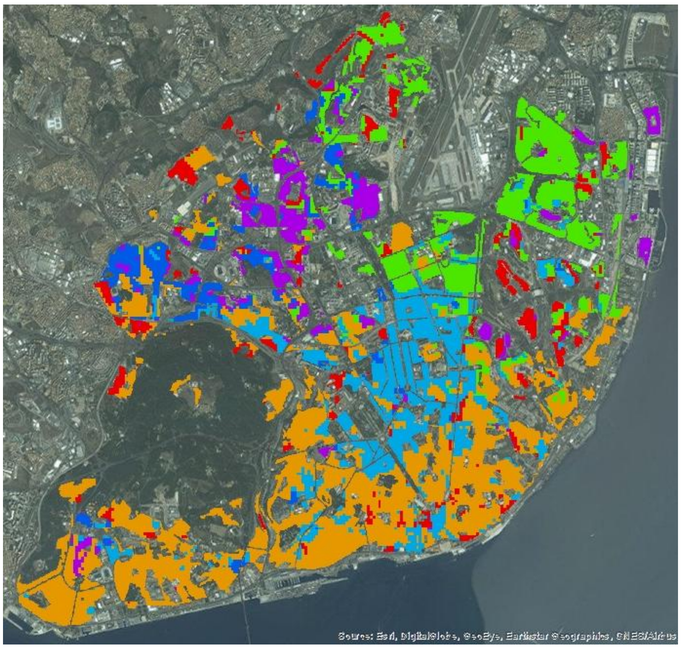
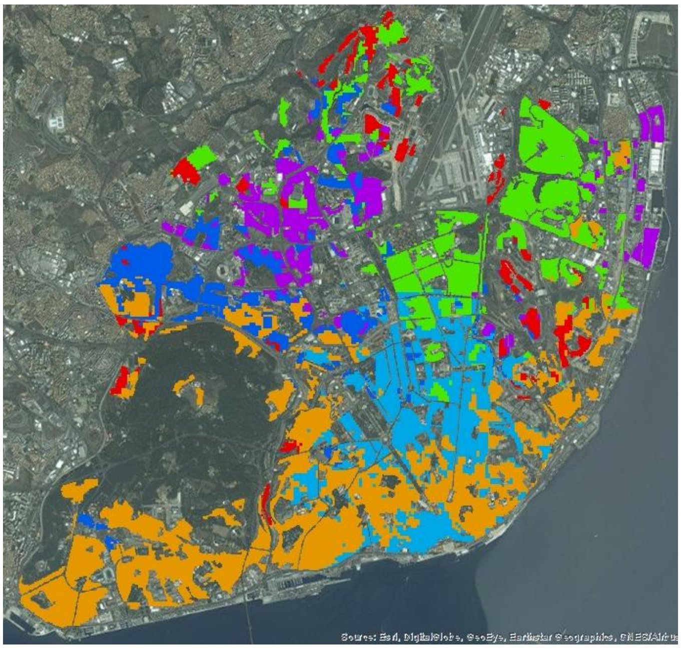

Source: Esri, DigitalGlobe, GeoEye, Earthstar Geographics, CNES/Airbus

Figure 5-17 Geo-SOM 2001 Choropleth



Source: Esri, DigitalGlobe, GeoEye, Earthstar Geographics, CNES/Airbu

Figure 5-18 Geo-SOM 2011 Choropleth

The final clusters have the following characteristics recurring to the centroids:

Cluster (Name/Number)	Description
Upper Class BT - 5	Newer and Taller Buildings Highly occupied accommodations by the owners Population Highly Educated with a lower unemployment rate Population tend to be young and active
Upper Class - 3	Older Buildings, mainly occupied by the owners Population Highly Educated, but tend to be old
Middle Class center - 2	Older and Taller Buildings with free accommodations and high rent rate Families tend to smaller (few members) Population tend to be Highly Educated, but old
Middle Class North County - 6	Older and short Buildings Population tend to have average Education Old Population
Middle Class River view - 6	Older and short Buildings, with a high rent rate and free accommodations Population tend to have lower Education level Population age tend to be old
Lower Class - 4	High density of population and accommodations in newer buildings, with the highest rent rate Large families Lower education level with the highest unemployment rate Population is the youngest

Table 5-5 Geo-SOM Hierarchical Clustering Description

Looking to the characteristics that represent each cluster, a compare was performed.

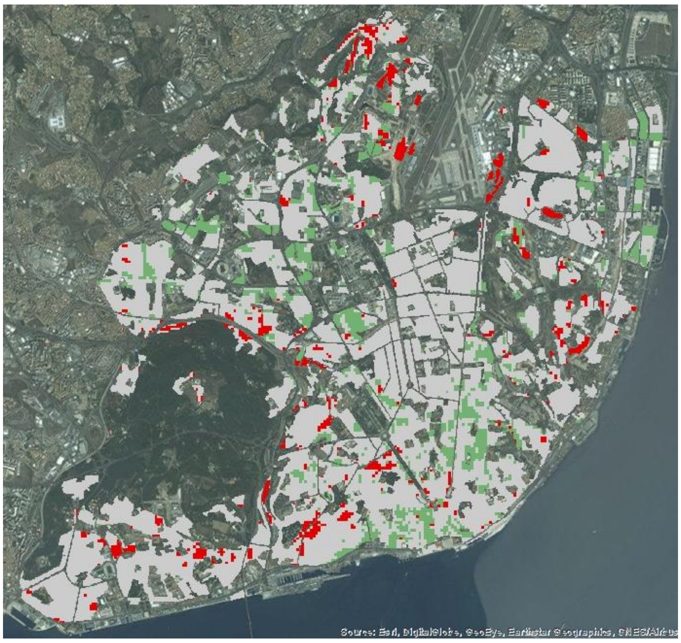

Source: Esri, DigitalGlobe, GeoEye, Earthstar Geographics, CNES/Airbus

Figure 5-19 Geo-SOM Comparison 2001 versus 2011 choropleth

It is possible to realize that some areas, particularly near the center and pathing to the river trough the principal avenues, have shown the green color, meaning they are improving, moving from a lower level cluster to a higher one.

5.2.3. Geo-SOM and SOM comparison

The clustering methods, SOM and Geo-SOM, results were different. The following picture presents this statement. It was performed an Euclidean distance between the centroids in order to attain which ones were more similar. If they matched, a blue color was addressed, red in the opposite.

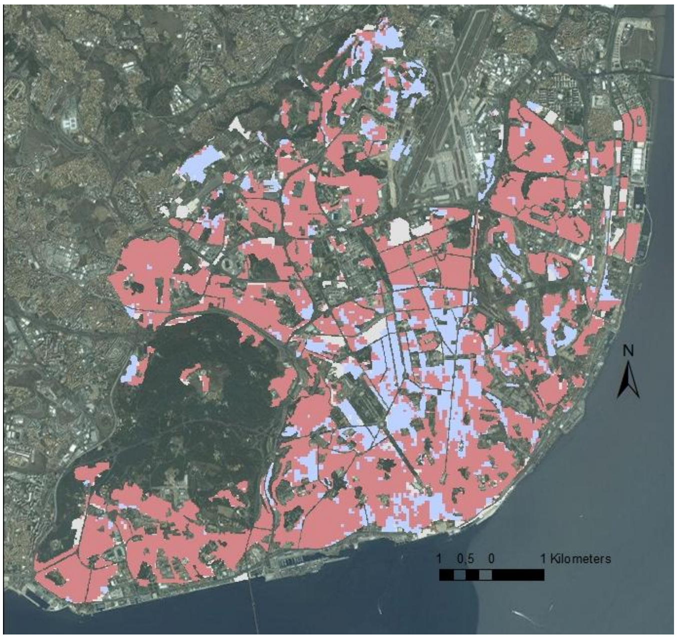


Figure 5-20 Geo-SOM versus SOM comparison

As we can see, the cluster mapping differs completely between both neural networks.

6. DISCUSSION

The use of two datasets (2001 and 2011) presented, in this study, some limitations. The subsections changed, as well the offered variables. In order to deal successfully with this constraints a dimension selection to match both datasets was performed and a raster analysis executed to deal with the BGRI evolutions. The variables removing and added ratios represented part of the preprocess task which is responsible by the project final quality.

The executed models aligned with the input variables were critically assessed based on the sum of the square errors and the visualization outputs, (Lee & Rinner, 2014) used similar approach based on the explanation (Openshaw, 1983).

.

The SOM resulting model is more heterogeneous, once it not taken into consideration the geolocation. On the other side, the Geo-SOM selected resulting model express the spatial dependence of the dataset, as stated by Tobler's First Law.

Towards the city depiction, the spatio-temporal mutation is part of the urban organic growth. Analyzing the spatial situation over time and local temporal variations over space (G. Andrienko et al., 2010, 2008) allows a deeper comprehension of what is the path taken by the city. It is important to understand the patterns to provide resources in a more efficient way by anticipating future needs. Regarding this idea, a comparison was made with both neural networks applied to both datasets (2001 and 2011).

Both algorithms showed a preservation of the visual patterns, although, recurring to the selected comparison method it was possible to identify the changes that occurred in the decennium timeframe. Particularly newsworthy, the city center revealed in both algorithms a positive trend in the area. Besides we didn't identify a change in the population age (keeps the old frame), the education level increased and the rent rate decreased clearly. These results are in line with the empiric sense of the residents.

The executed generalization process enabled an easy interpretation of the Lisbon County in the broad sense, but a large sort of information was removed in the pre-process stage. Although the differences between the datasets were overcame and the final results matched the expectations.

7. CONCLUSIONS

The goals of identify spatio-temporal changes linked with the Smart cities Initiative Framework were achieved successfully, once it was possible to use the already available information (census) and apply Unsupervised Neural Networks to generate new knowledge.

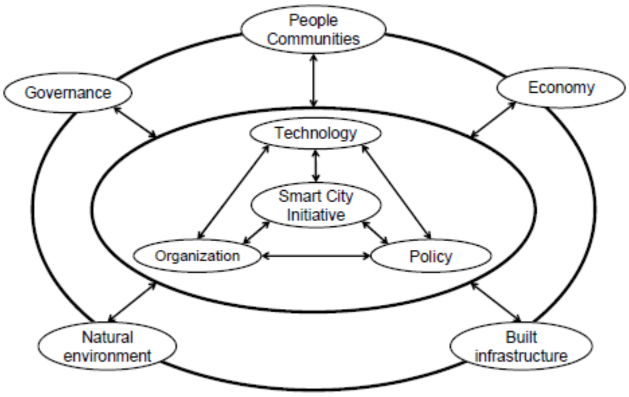


Figure 7-1 Smart City Initiatives framework(Chourabi et al., 2012)

The identification of spatio-temporal patterns in the Lisbon Metropolitan Area between 2001 and 2011 taking into consideration the several aspects and characteristics of the selected region revealed the extent of possibilities enabled by the application of the latest computer sciences technologies and information technologies tools.

The application of different methods for exploratory spatial data analysis and clustering were applied. Essentially to main techniques were used, the standard SOM (Kohonen, 1995, 2013) and variant Geo-SOM (Bação et al., 2004; R. Henriques et al., 2012). In order to improve the human perception of the output results, the GIS tools were weighty to provide the desired visualization ability.

It was identified that the presence of subsections (block level) without residents, housing buildings or classical families influence negatively the learning process, once it generates their own cluster, or belong to other cluster types where they influence final centroid components values. In that way, it was intersected the BGRI, Landscape Occupation Map and a 50x50 meters grid to overcame the spatio-temporal challenge.

The final components choice, it was only possible after several iterations and comparisons between the centroids outputs and relative errors. The ratio between gains of interpretation versus the information loss is dependent on the defined goals. More accurate results, meaning less error, increases the representatives number. On the other side, the ability to generalize is one of the clustering goals leading to a depreciation of the accuracy.

Facing this challenge and all the related issues presented by the “curse of dimensionality” (Ceotto, Tantardini, & Aspuru-Guzik, 2011) there was a set of variables related to the buildings construction age that weren’t considered . It became clear that the bylaws weren’t connected with the buildings age but directly connected with the location.

With the objective to provide a tool to be used by key stakeholders, this methodology and study could enable the increment in resources planning allocation. The identification of areas where there are older people living alone, bigger unemployment rates, younger people, less education level, or other features necessarily need different approaches among them.

Thus, the presented study aimed to present a wider arrangement of the Lisbon County. However, the execution of different studies could lead to a different variables selection, meeting the analysis subject. In this later case it is possible to require different datasets, and/or use techniques over the components transforming them. In further analysis, Principal Component Analysis (PCA) might be used to create new components. The PCA in partnership with ratios or base variables represents an opportunity to future work.

The original datasets supplied by the Statistics Portugal represent the most reliable information, although some limitations were found. The final components were obtained over the iterations under 2011 dataset, when applying the 2001 classification it was identified that some variables were missing and others needed a deep transformation to fit the centroids components. Besides that, the BGRI suffered mutations which were overcame by the raster introduction.

The resulting clusters, obtained with the followed methodology, were successfully presented in choropleth maps, where it is possible to identify areas with different potentials to IT end-users. The type of population and infrastructure is an indicator of how capable is already that area, or in the opposite, which measures must be taken to evolve or re-adapt the IT solutions considering the digital divide (Cruz-Jesus, Oliveira, & Bacao, 2012).

The images provided with the objective of suppling the basis to implement real-life applications and services demanded by the smart city context. IoT and IoS will be the enablers. The use of GIS allows the visualization of the performed Artificial Neural Networks outputs. The ability to ally the computer sciences with the human interaction presents an opportunity to evolve the city in a more “fancy” way.

Based on the general results and related conclusions, the Lisbon County City Center presents a mutation phenomenon based on social changes which indicates a process of gentrification (Smith, 1979). The Lisbon analysis performed showed that the majority of the historical center changed from lower level clusters to higher ones. The increasing of the ownership, instead of renting, shows the appellative investment that the city center claims. At certain point the population also changed, the higher qualified took place from the lower educated, creating a different demand to the city council. Although it is necessary to evaluate if the gentrification effect, generally substitution of poorer population by wealthier one, is already in place (Lees, Slater, & Wyly, 2008).

The possibility of reborn the Lisbon city center with younger population wasn’t expected to show up in the applied methods once the Portuguese population is suffering a strong aging (Statistics Portugal, 2014).

In future works, besides the improvement of the applied methodologies, the inclusion of variables and data sources targeted to the gentrification phenomena will be an asset to smart city context.

8. {BIBLIOGRAPHY}

Allouche, M., & Moulin, B. (2005). Amalgamation in cartographic generalization using Kohonen's feature nets. *International Journal of Geographical Information Science*, 19(8): 899 - 914.

Andrienko, G., Andrienko, N., Bremm, S., Schreck, T., Landesberger, T. v., Bak, P., & Keim, D. (2010). Space-in-Time and Time-in-Space Self-Organizing Maps for Exploring Spatiotemporal Patterns. *Eurographics/ IEEE-VGTC Symposium on Visualization 2010* (pp. Volume 29, Number 3). G. Melançon, T. Munzner, and D. Weiskopf.

Andrienko, G., Andrienko, N., Dykes, J., Fabrikant, S., & Wachowicz, M. (2008). Geovisualization of dynamics, movement and change: key issues and developing approaches in visualization research. *Information Visualization,*, pp. 7(3-4), pp. 173-180.

Andrienko, N., & Andrienko, G. (2006). *Exploratory Analysis of Spatial and Temporal Data - A Systematic Approach.* Germany: Springer.

Bação, F., Lobo, V., & Painho, M. (2004). Geo-Self-Organizing Map (Geo-SOM) for Building and Exploring Homogeneous Regions. *Geographic Information Science, Proceedings*, 3234, 22-37.

Bação, F., Lobo, V., & Painho, M. (2005). The self-organizing map, the Geo-SOM, and relevant variants for geosciences. *Computers & Geosciences*, 31 155-163.

Birch, E. L., M., W. S., & Keating, A. (2015). Best Practice Methods for Cities: State of the Art. *Transforming Distressed Global Communities: Making Inclusive, Safe, Resilient, and Sustainable Cities*, 353.

Buttenfield, B., & McMaster, R. (1991). *Map Generalization: Making Rules for Knowledge representation.* London: Longman Scientific & Technical.

Ceylan, R., & Özbay, Y. (2007). Comparison of FCM, PCA and WT techniques for classification ECG arrhythmias using artificial neural network. *Expert Systems with Applications*, 33(2), 286-295.

Claussen, J. (2003). Winner-relaxing and winner-enhancing Kohonen maps: Maximal mutual Information from Enhancing the Winner. *Complexity*, 8(4) 15-22.

Cottrell, M., Fort, J., & Pagès, S. (1998). Theoretical Aspects of the SOM Algorithm. *Neurocomputing*, 21(1-3) 119-138.

ESRI. (1998). *ESRI Shapefile Technical Description.* United States of America: Environmental Systems Research Institute, Inc.

Gerometta, J., Haussermann, H., & Longo, G. (2005). Social innovation and civil society in urban governance: strategies for an inclusive city. *Urban Studies*, 42(11), 2007-2021.

Grant, H., Lubbe, S., & Klopper, R. (2011). The Impact of Information Quality on Information Research. *Alternation Special Edition*, pp. 4 288-305.

Henriques, R. (2010). *Artificial Intelligence in Geospatial Analysis: applications of Self-Organizing Maps in the context of Geographic Information Science.* Lisboa: Universidade Nova de Lisboa - Instituto Superior de Estatística e Gestão de Informação.

Henriques, R., Bação, F., & Lobo, V. (2012). Exploratory geospatial data analysis using the GeoSOM suite. *Computers, Environment and Urban Systems*, pp. 36 218–232.

Huang, L., Yan, L., & Wu, J. (2016). Assessing urban sustainability of Chinese megacities: 35 years after the economic reform and open-door policy. *Landscape and Urban Planning*, 145, 57-70.

Jain, A., & Dubes, R. (1988). *Algorithms for clustering data.* Englewood Cliffs, New Jersey: Prentice-Hall, Inc.

Kambhatla, N., & Leen, T. K. (1997). Dimension reduction by local principal component analysis. *Neural Computation*, 9(7), 1493-1516.

Kohonen, T. (2001). *Self-Organizing Maps 3rd.* New York: Springer.

Kohonen, T. (2013). Essentials of the self-organizing map. *Neural Networks*, 37 52-65.

Koua. (2003). Using self-organizing maps for information visualization and knowledge discovery in complex geospatial datasets. *21st International Cartographic Conference* (pp. 1694-1702). Durban, South Africa: The International Cartographic Association.

Koua, E., & Kraak, M.-J. (2004). Geovisualization to support the exploration of large health and demographic survey data. *International Journal of Health Geographics*, 3-12.

Lee, A., & Rinner, C. (2015). Visualizing urban social change with Self-Organizing Maps: Toronto neighbourhoods, 1996-2006. *Habitat International*, pp. 92-98.

Lobo, V., & Gorricha, J. (2011). Visualization of Clusters in Geo-referenced Data Using Three-dimensional Self-Organizing Maps. *Proceedings of the IF&GIS Workshop.* Brest, France.

Lobo, V., & Gorricha, J. (2012). Improvements on the visualization of clusters in geo-referenced data using Self-Organizing Maps. *Computers & Geosciences*, pp. 43 177–186.

Mennis, J., & Guo, D. (2009). Spatial data mining and geographic knowledge discovery—An introduction. *Computers, Environment and Urban Systems*, 33 403-408.

Nam, T., & Pardo, T. A. (2011). Conceptualizing Smart City with Dimensions of Technology, People, and Institutions. *The Proceedings of the 12th Annual International Conference on Digital Government Research* (pp. 282-291). Washington, DC: Association for Computing Machinery.

Pearson, K. (1901). LIII On Lines and Planes of Closest Fit to Systems of Points in Space. *Philosophical Magazine Series 6*, 2(11): 559 - 572.

Penn, B. (2005). Using self-organizing maps to visualize high-dimensional data. *Computers & Geosciences*, 31(5) 531-544.

Polèse, M. (2010). *The resilient city: on the determinants of successful urban economies.* Montreal, (Quebec) Canada: Urbanisation, culture, société, INRS.

Robinson, A., Morrinson, J., Muehrcke, P., Kimerling, J., & Guptill, S. (1995). *Elements of Carthography.* New York: John Wiley & Sons, Inc.

Roche, S. (2014). Geographic Information Science I: Why does a smart city need to be spatially enabled? *Progress in Human Geography*, Vol. 38(5) 703–711.

Sammon, J. (1969). A Nonlinear Mapping for Data Structure Analysis. *IEEE Transactions on Computers*, C-18 5.

Sester, M. (2005). Optimization approaches for generalization and data abstraction. *International Journal of Geographical Information Science*, 19(8): 871 - 897.

Skupin, A. (2002). A cartographic approach to visualizing conference abstracts. *Computer Graphics and Applications,*, 22.1 50-58.

Skupin, A., & Agarwal, P. (2008). *Self-Organizing Maps: Applications in geographic information science.* Chincester: John Wiley & Sons, Ltd.

Skupin, A., & Hagelman, R. (2005). Visualizing Demographic Trajectories with Self-Organizing Maps. *GeoInformatica*, pp. 9 159-179.

Tobler, W. (1970). A Computer Movie Simulating Urban Growth in the Detroit Region. *Economic Geography, Vol. 46, Supplement: Proceedings. International Geographical Union. Commission on Quantitative Methods*, 234-240.

Ultsch, A., & Siemon, H. (1990). Kohonen's self organizing feature maps for exploratory data analysis. *Proceedings of International Neural Network Conference* (pp. 305-308). Paris: Kluwer Academic Press.

Vesanto, J. (1999). SOM-based data visualization methods. *Intelligent Data Analysis*, 3 111-126.

9. ATTACHMENTS

Variable	Index	0%	25%	50%	75%	100%
Build5Floo	A	0,00	0,08	0,39	0,85	1,00
Buil80	B	0,00	0,59	0,92	1,00	1,00
Build81to00	C	0,00	0,00	0,02	0,16	1,00
Dens.Accom	D	0,000	0,005	0,009	0,014	45,324
EmptyAccom	E	0,00	0,07	0,14	0,21	0,92
Rent.Accom	F	0,00	0,12	0,29	0,41	1,00
Famil5plus	G	0,00	0,02	0,04	0,08	1,00
Age0to19	H	0,00	0,13	0,16	0,21	0,89
Age20to64	I	0,03	0,53	0,58	0,63	1,00
Age65	J	0,00	0,17	0,25	0,32	0,97
Edu.Prim	K	0,00	0,32	0,47	0,65	1,00
Edu.Sec	L	0,00	0,15	0,19	0,21	0,60
Edu.Tert	M	0,00	0,16	0,33	0,47	0,88
Unemploy	N	0,00	0,03	0,05	0,07	0,47
Dens.Pop	O	0,000	0,008	0,016	0,024	139,849

Table 9-1 Quantiles for the Input Selected Variables (Scale used in the Figures below)

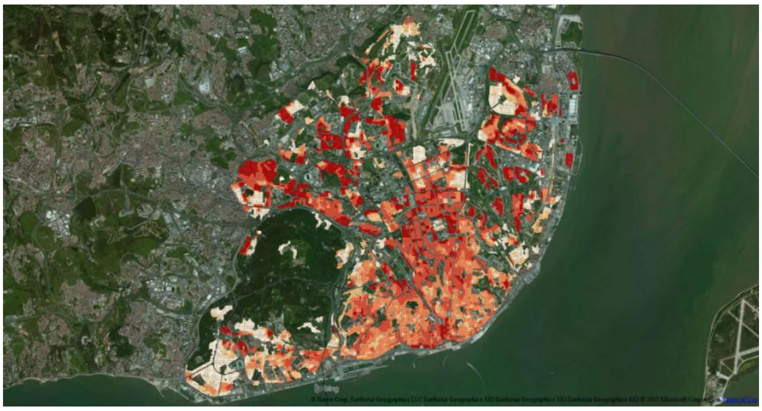


Figure 9-1 Variable A 2001

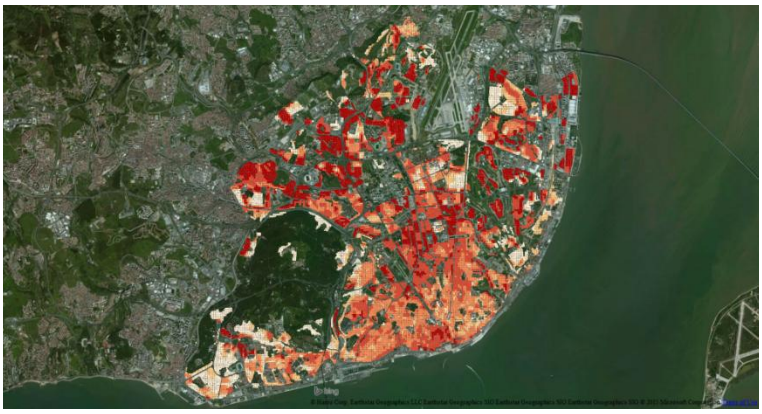


Figure 9-2 Variable A 2011

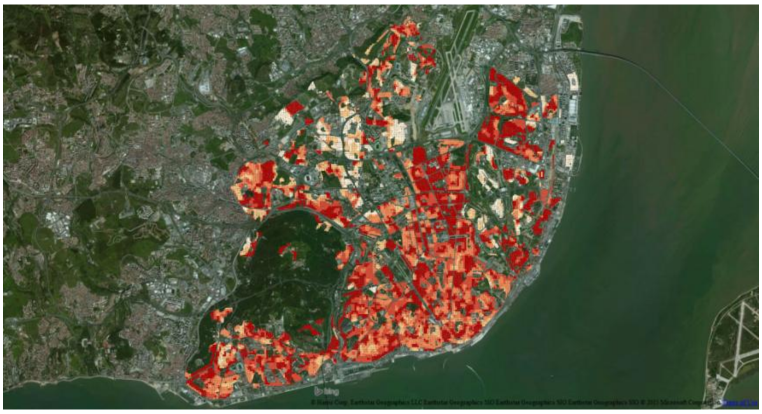


Figure 9-3 Variable B 2001

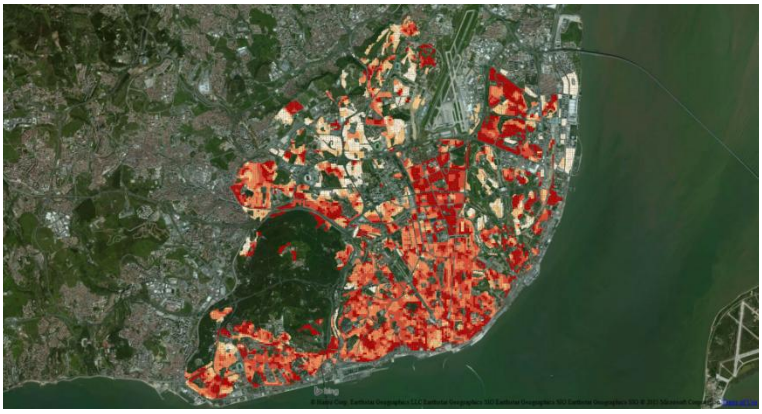


Figure 9-4 Variable B 2011

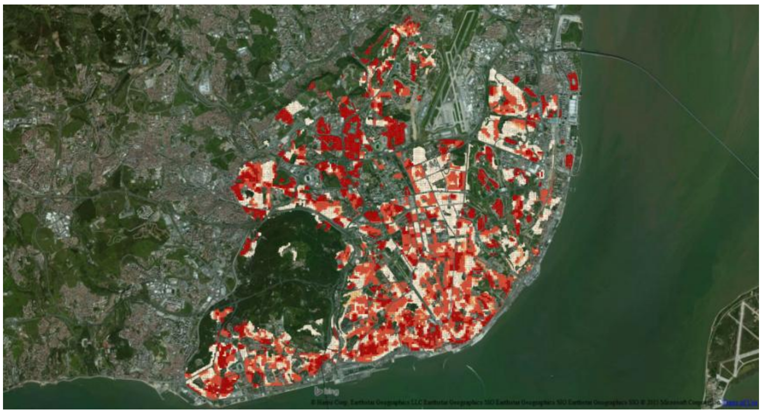


Figure 9-5 Variable C 2001

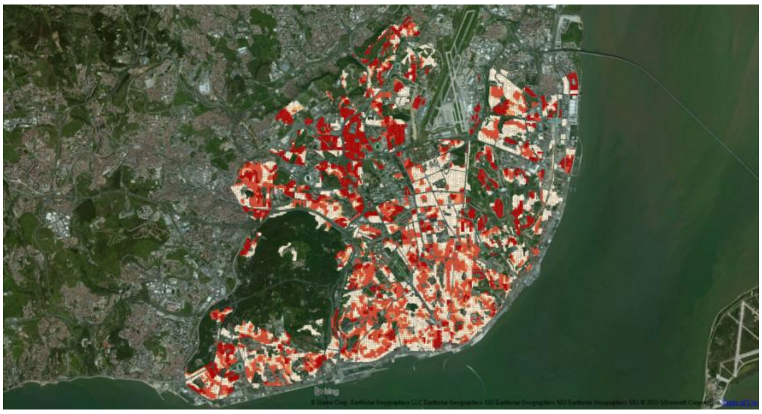


Figure 9-6 Variable C 2011

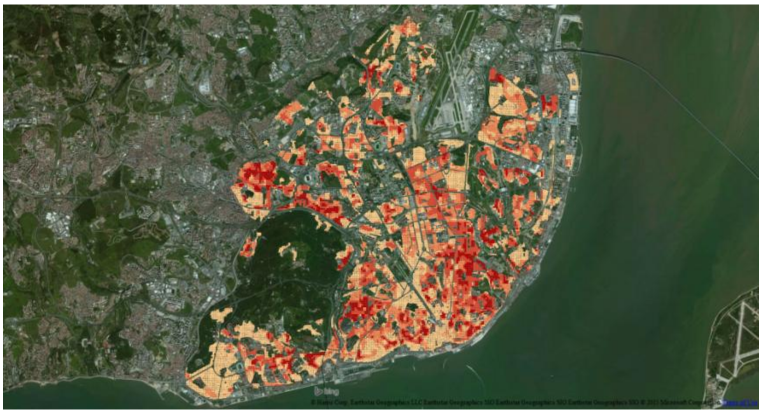


Figure 9-7 Variable D 2001

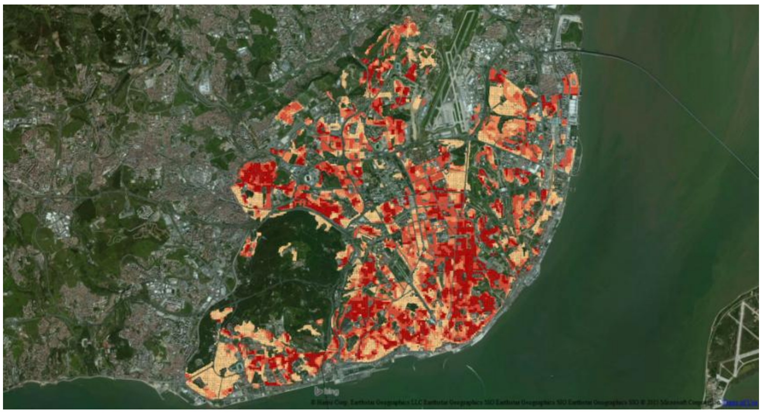


Figure 9-8 Variable D 2011

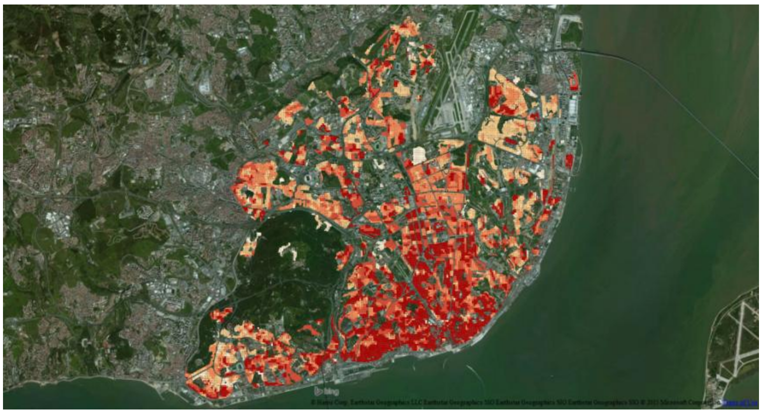


Figure 9-9 Variable E 2001

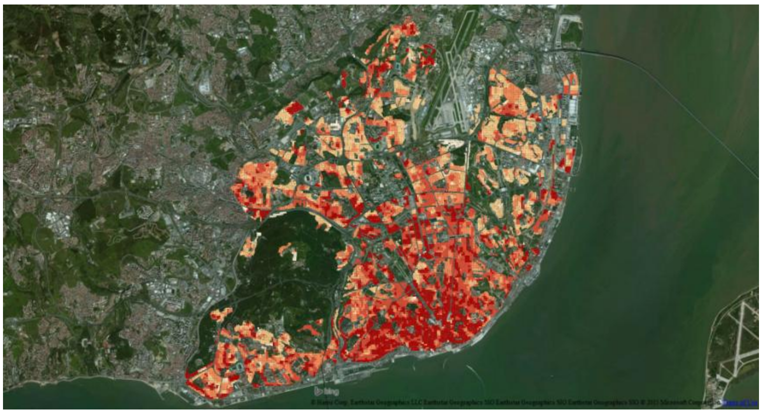


Figure 9-10 Variable E 2011

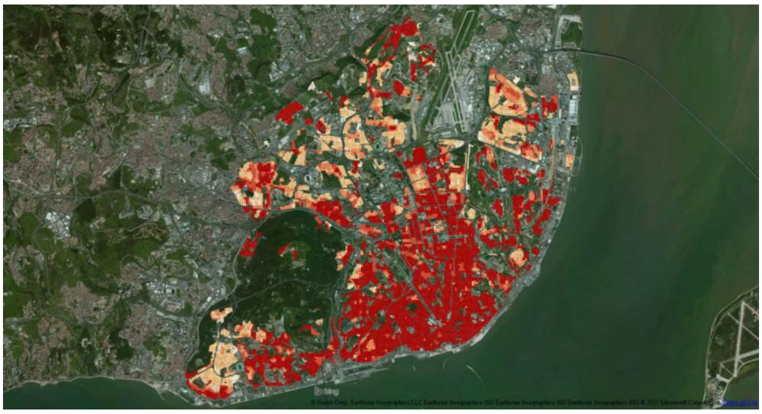


Figure 9-11 Variable F 2001

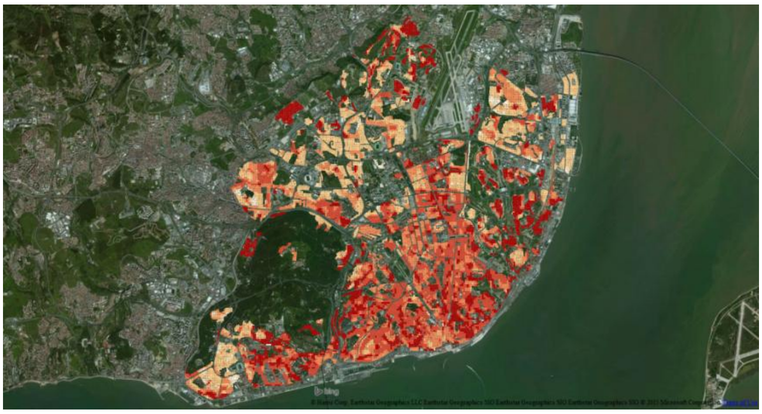


Figure 9-12 Variable F 2011

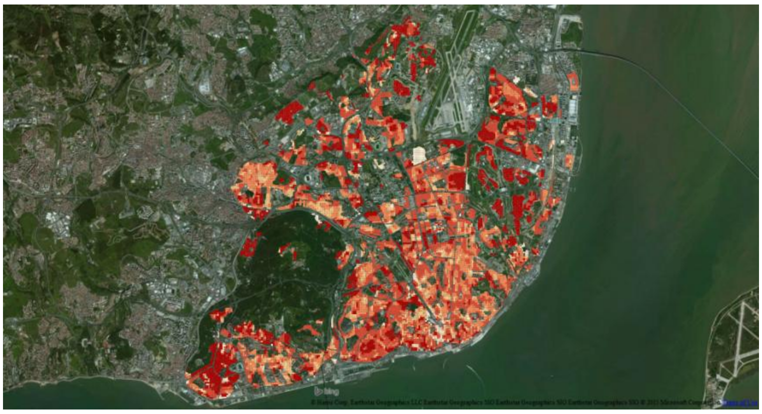


Figure 9-13 Variable G 2001

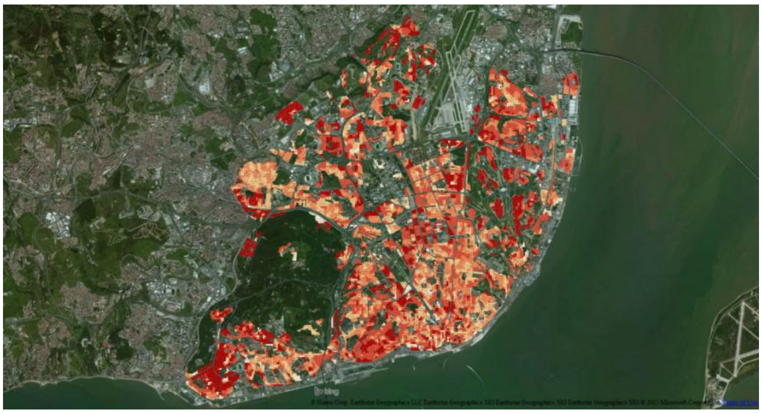


Figure 9-14 Variable G 2011

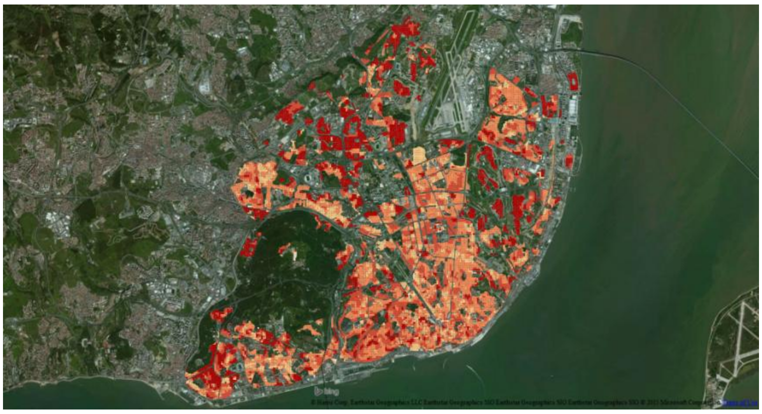


Figure 9-15 Variable H 2001

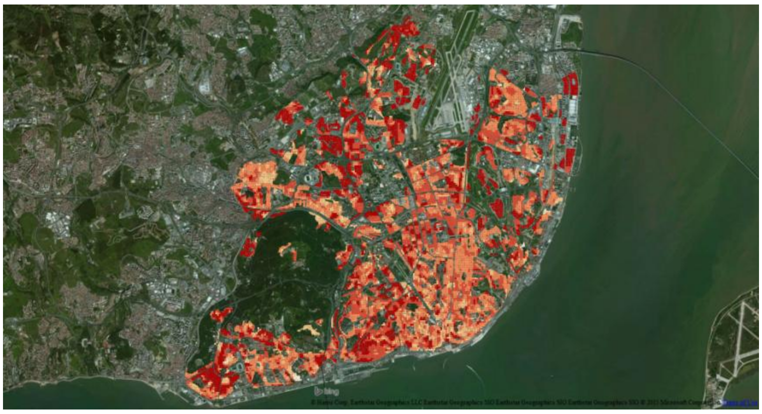


Figure 9-16 Variable H 2011

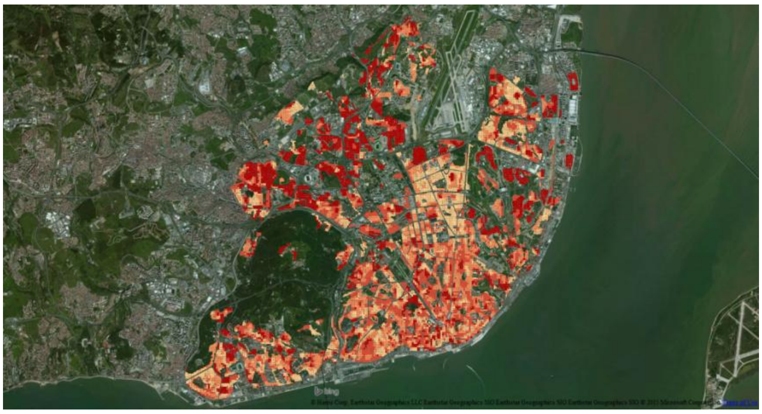


Figure 9-17 Variable I 2001

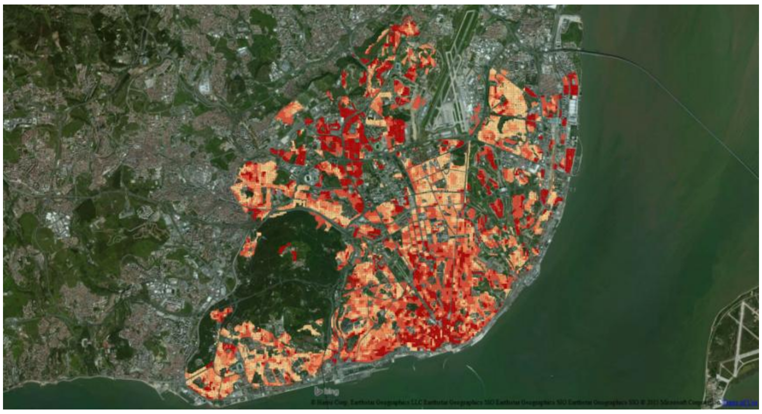


Figure 9-18 Variable I 2011

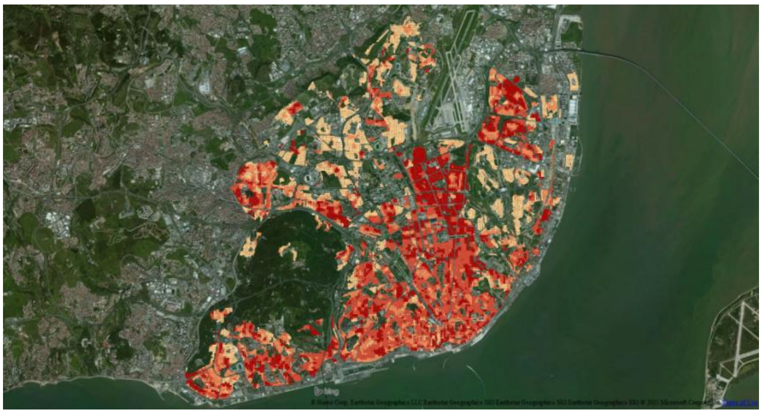


Figure 9-19 Variable J 2001

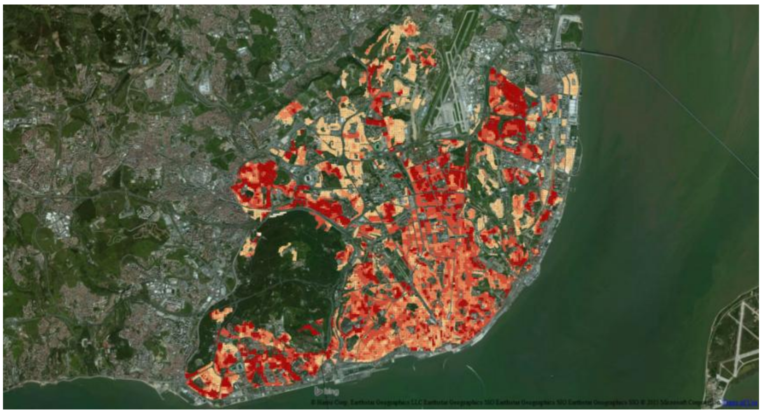


Figure 9-20 Variable J 2011

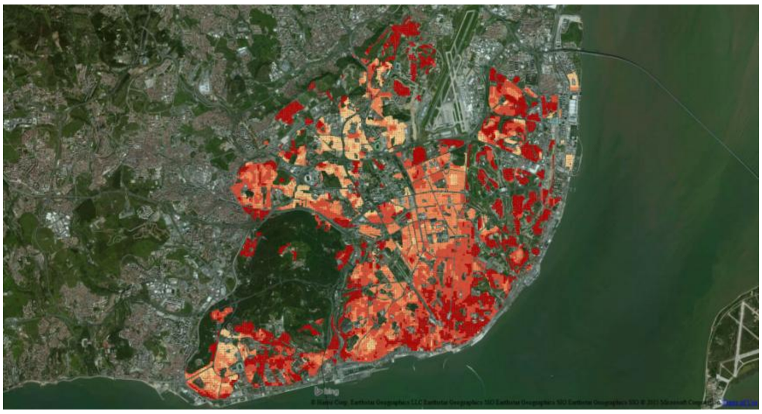


Figure 9-21 Variable K 2001

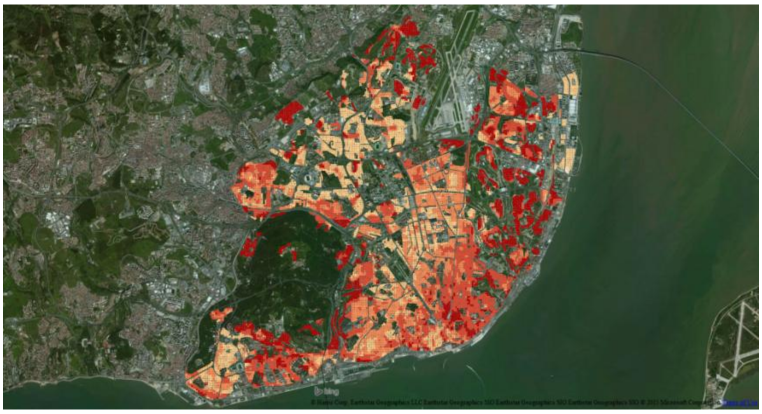


Figure 9-22 Variable K 2011

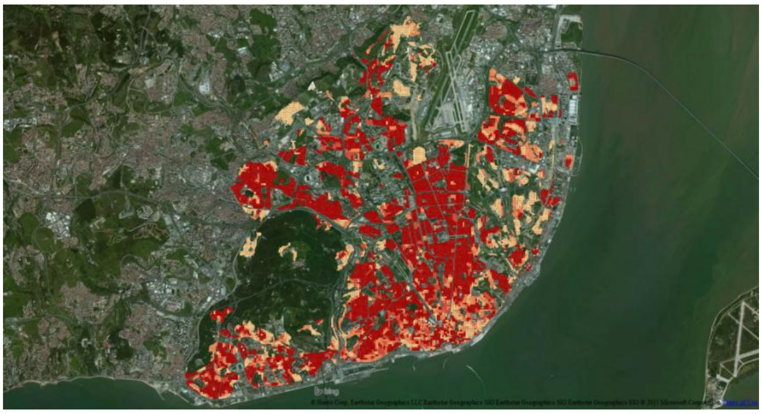


Figure 9-23 Variable L 2001

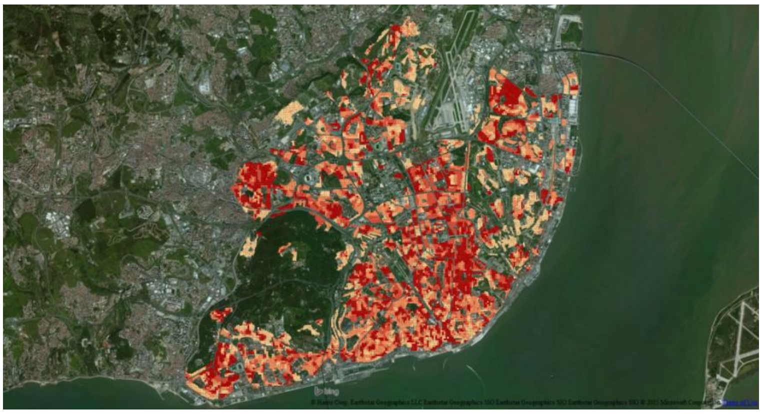


Figure 9-24 Variable L 2011

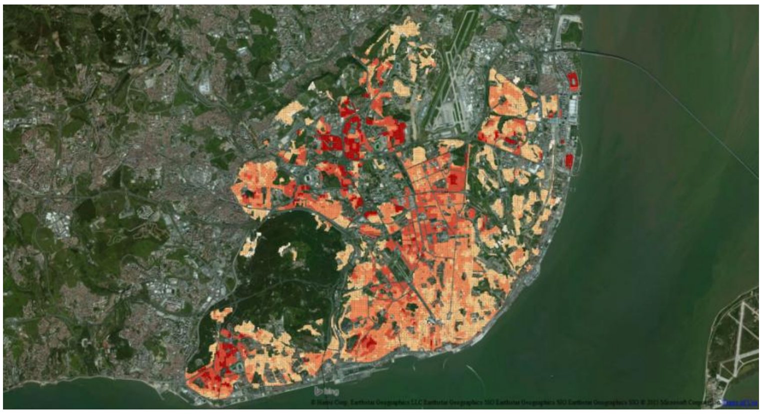


Figure 9-25 Variable M 2001

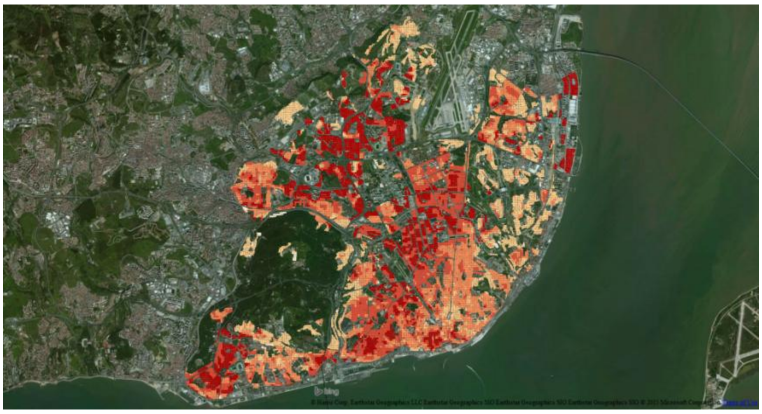


Figure 9-26 Variable M 2011

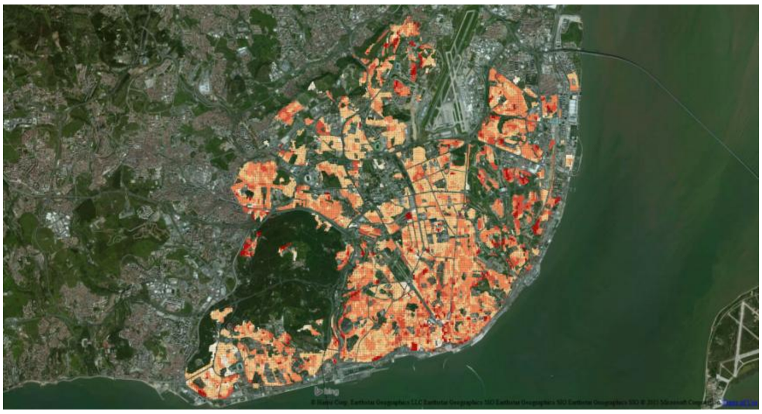


Figure 9-27 Variable N 2001

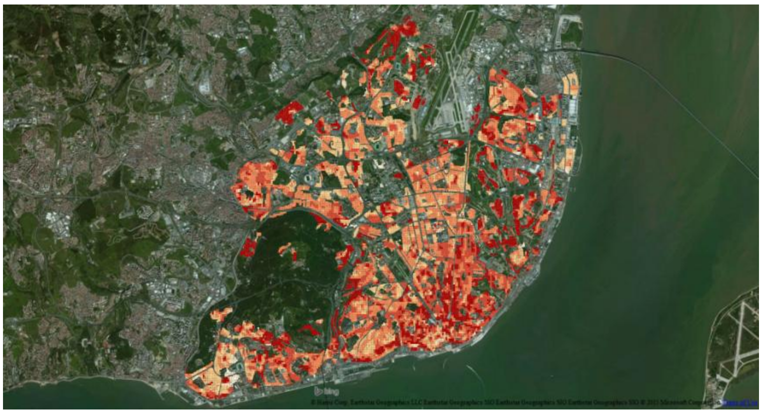


Figure 9-28 Variable N 2011

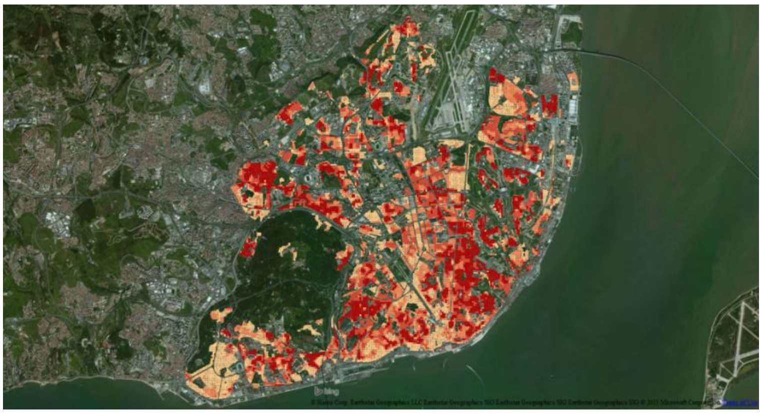


Figure 9-29 Variable O 2001

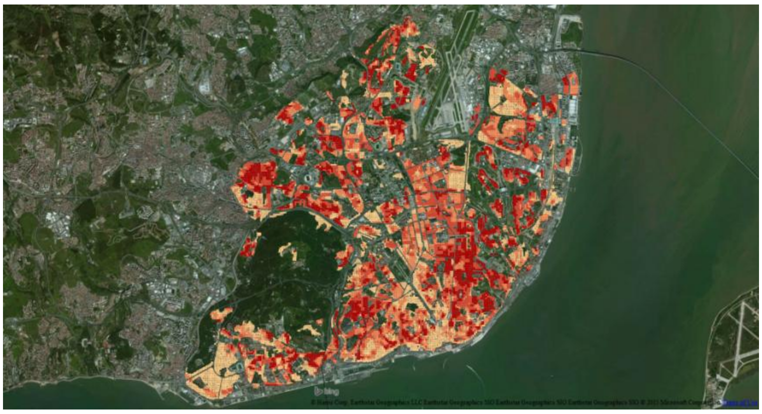
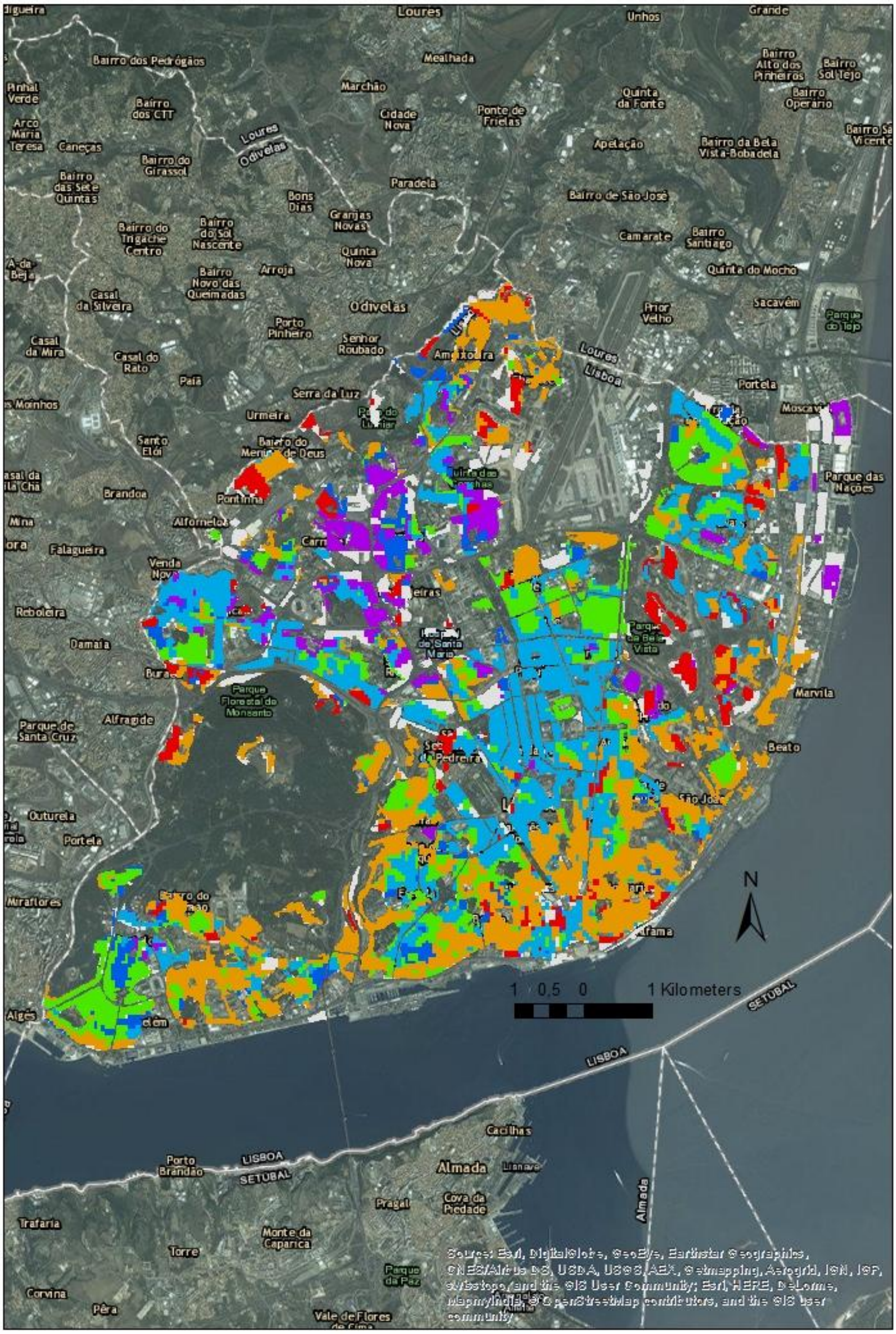
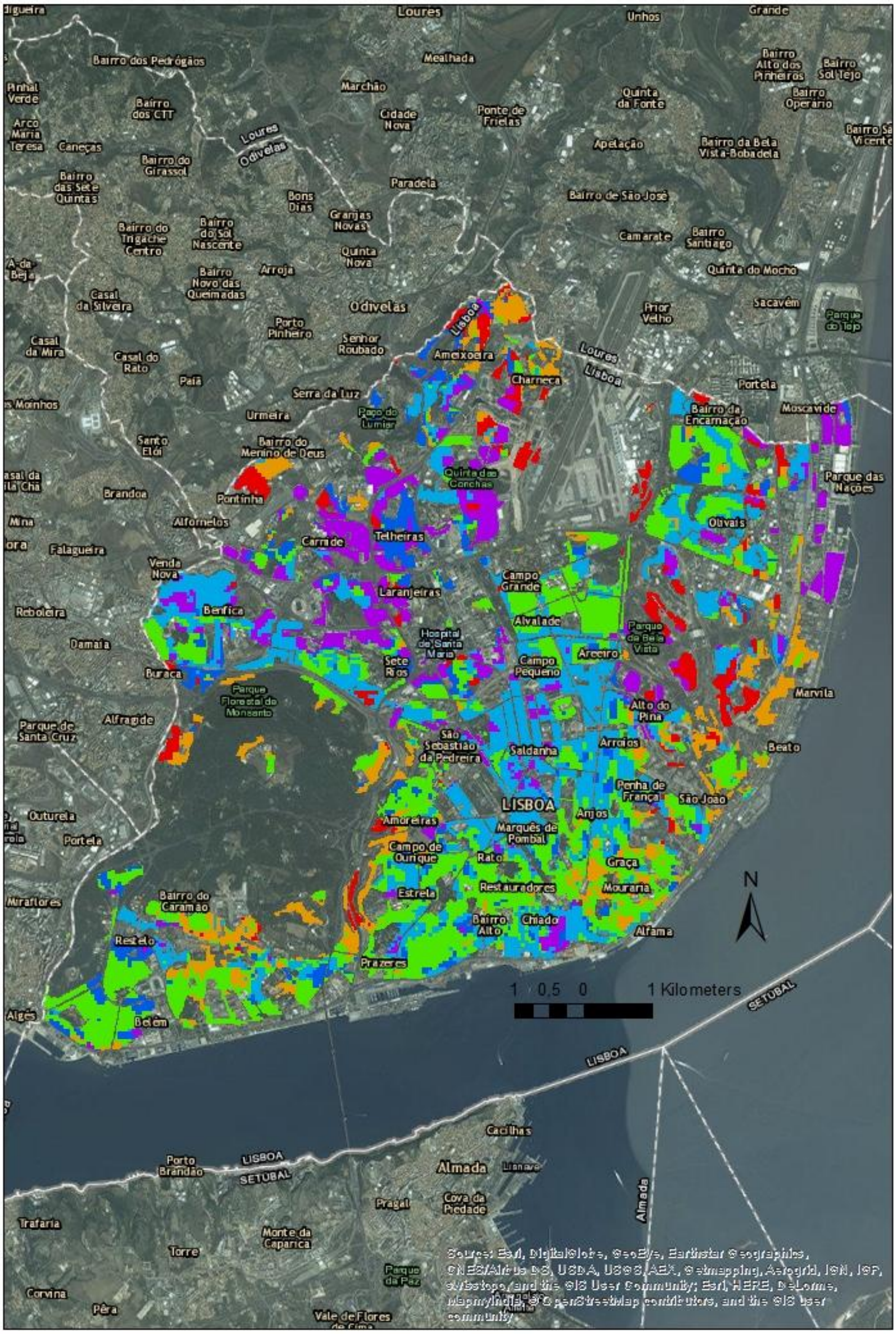


Figure 9-30 Variable O 2011



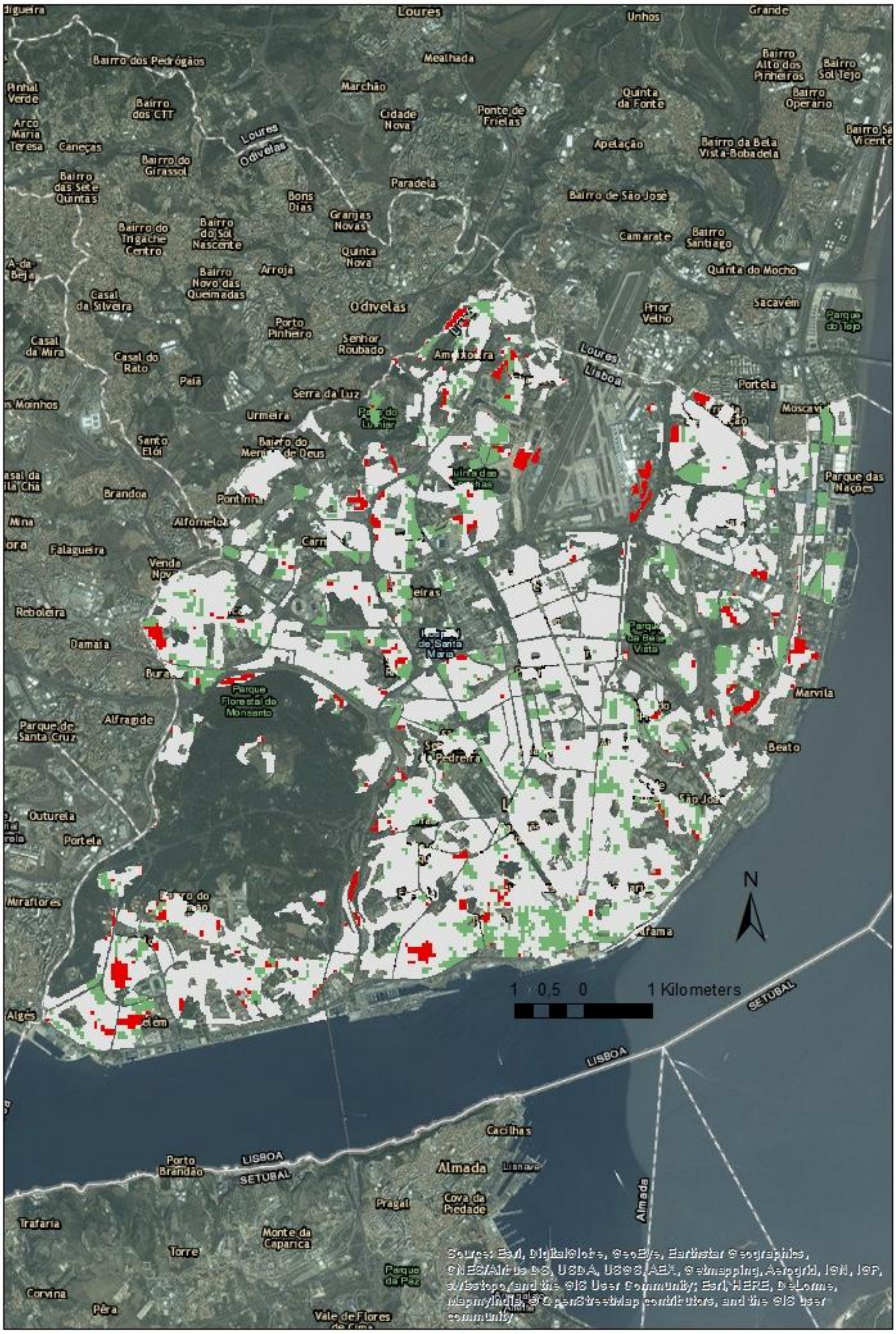
Source: Esri, DigitalGlobe, GeoEye, Earthstar Geographics, CNES/Airbus DS, USDA, USGS, AEX, Getmapping, Aerogrid, IGN, IGP, swisstopo, and the GIS User Community; Esri, HERE, DeLorme, MapmyIndia, © OpenStreetMap contributors, and the GIS user community

Figure 9-31 SOM Hierarchical Clustering (6) Choropleth 2001



Source: Esri, DigitalGlobe, GeoEye, Earthstar Geographics, CNES/Airbus DS, USDA, USGS, AEX, Getmapping, Aerogrid, IGN, IGP, swisstopo, and the GIS User Community; Esri, HERE, DeLorme, MapmyIndia, © OpenStreetMap contributors, and the GIS user community

Figure 9-32 SOM Hierarchical Clustering (6) Choropleth 2011



Source: Esri, DigitalGlobe, GeoEye, Earthstar Geographics, CNES/Airbus DS, USDA, USGS, AEX, Getmapping, Aerogrid, IGN, IGP, swisstopo, and the GIS User Community; Esri, HERE, DeLorme, MapmyIndia, © OpenStreetMap contributors, and the GIS user community

Figure 9-33 SOM Hierarchical Clustering (6) Compare 2001 versus 2011

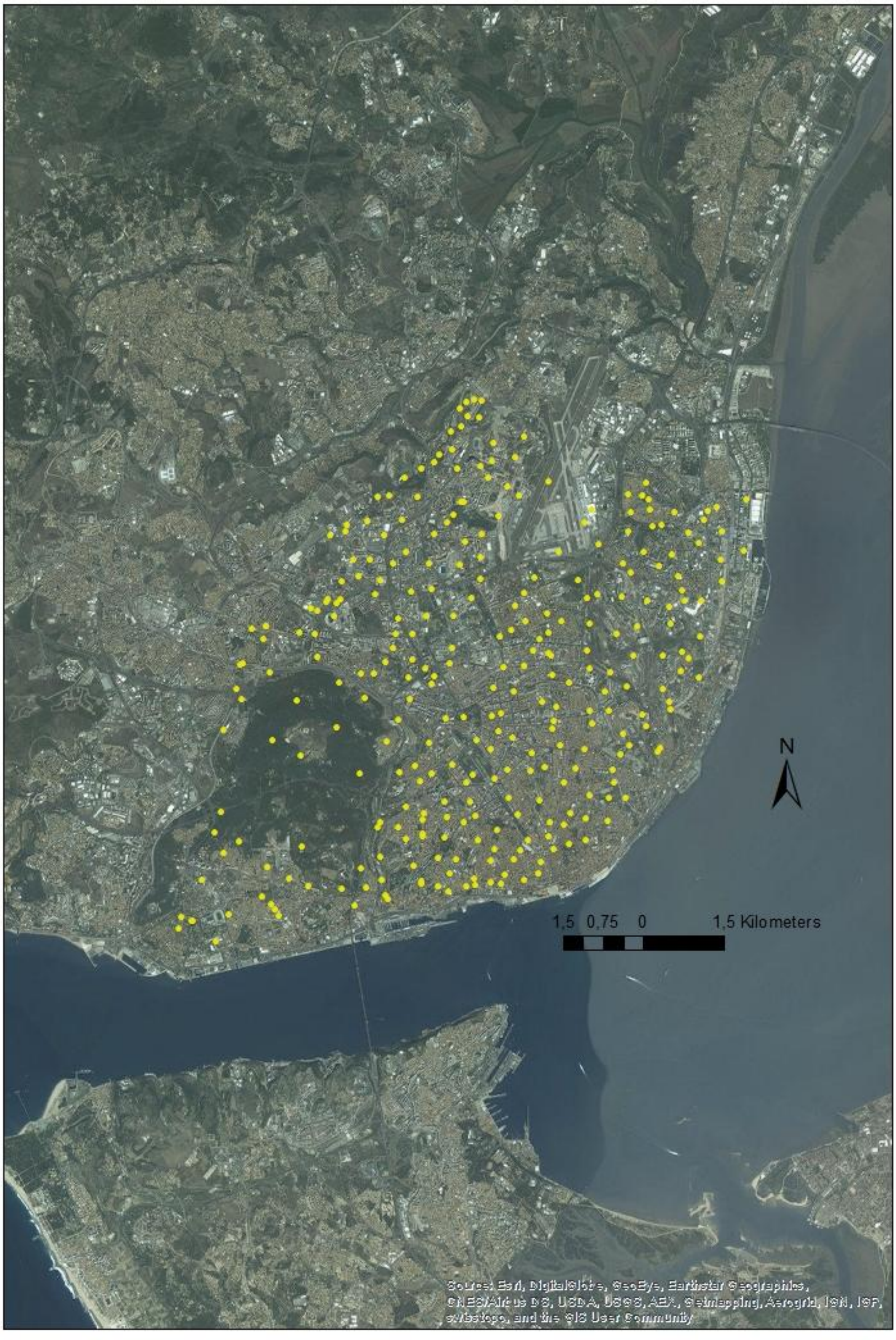

N
1,5 0,75 0 1,5 Kilometers
Source: Esri, DigitalGlobe, GeoEye, Earthstar Geographics, CNES/Airbus DS, USDA, USGS, AEX, Getmapping, Aerogrid, IGN, IGP, swisstopo, and the GIS User Community

Figure 9-34 Codebook Geo-SOM Choropleth

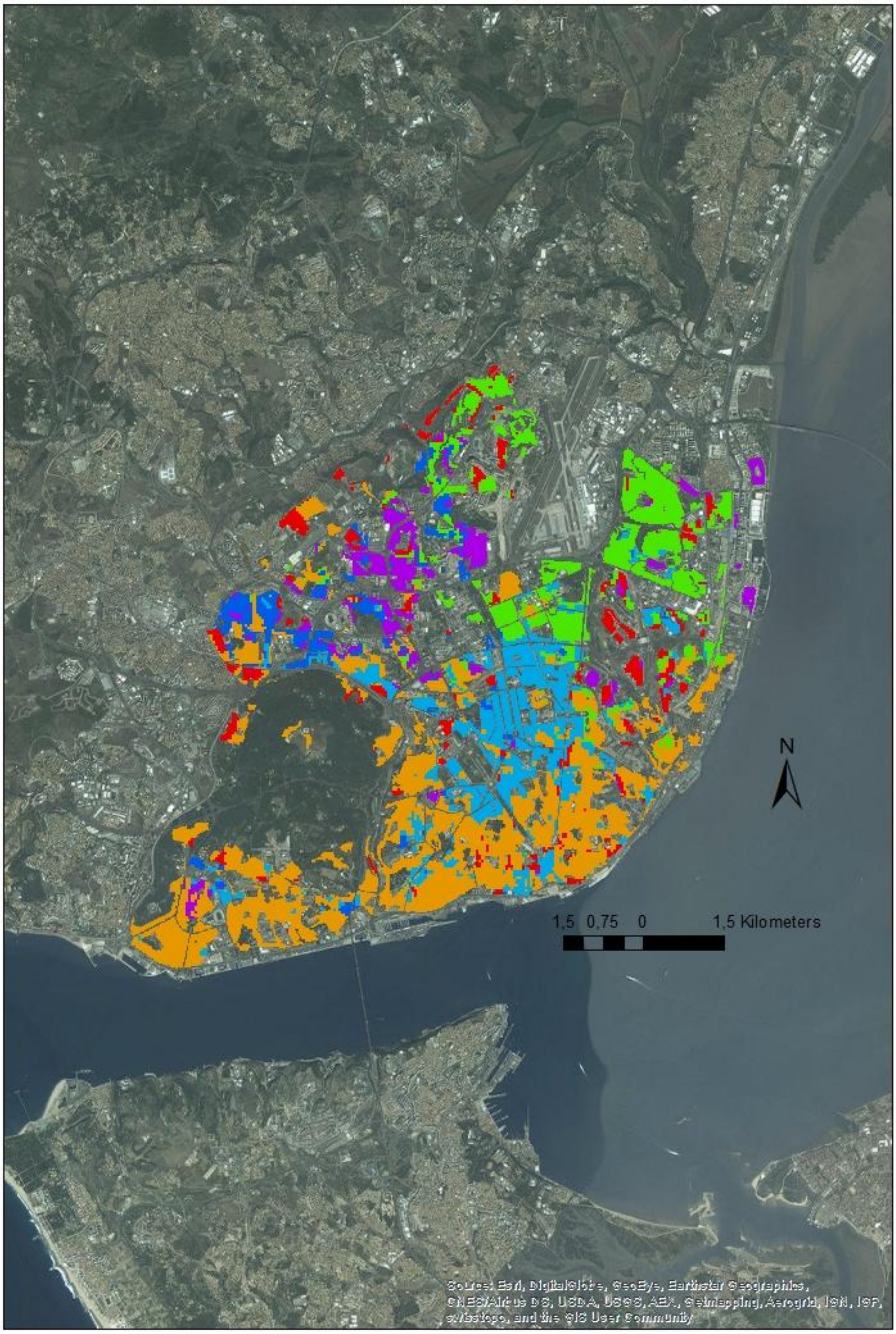

N
1,5 0,75 0 1,5 Kilometers
Source: Esri, DigitalGlobe, GeoEye, Earthstar Geographics, CNES/Airbus DS, USDA, USGS, AEX, Getmapping, Aerogrid, IGN, IGP, swisstopo, and the GIS User Community

Figure 9-35 Geo-SOM Hierarchical Clustering (6) Choropleth 2001

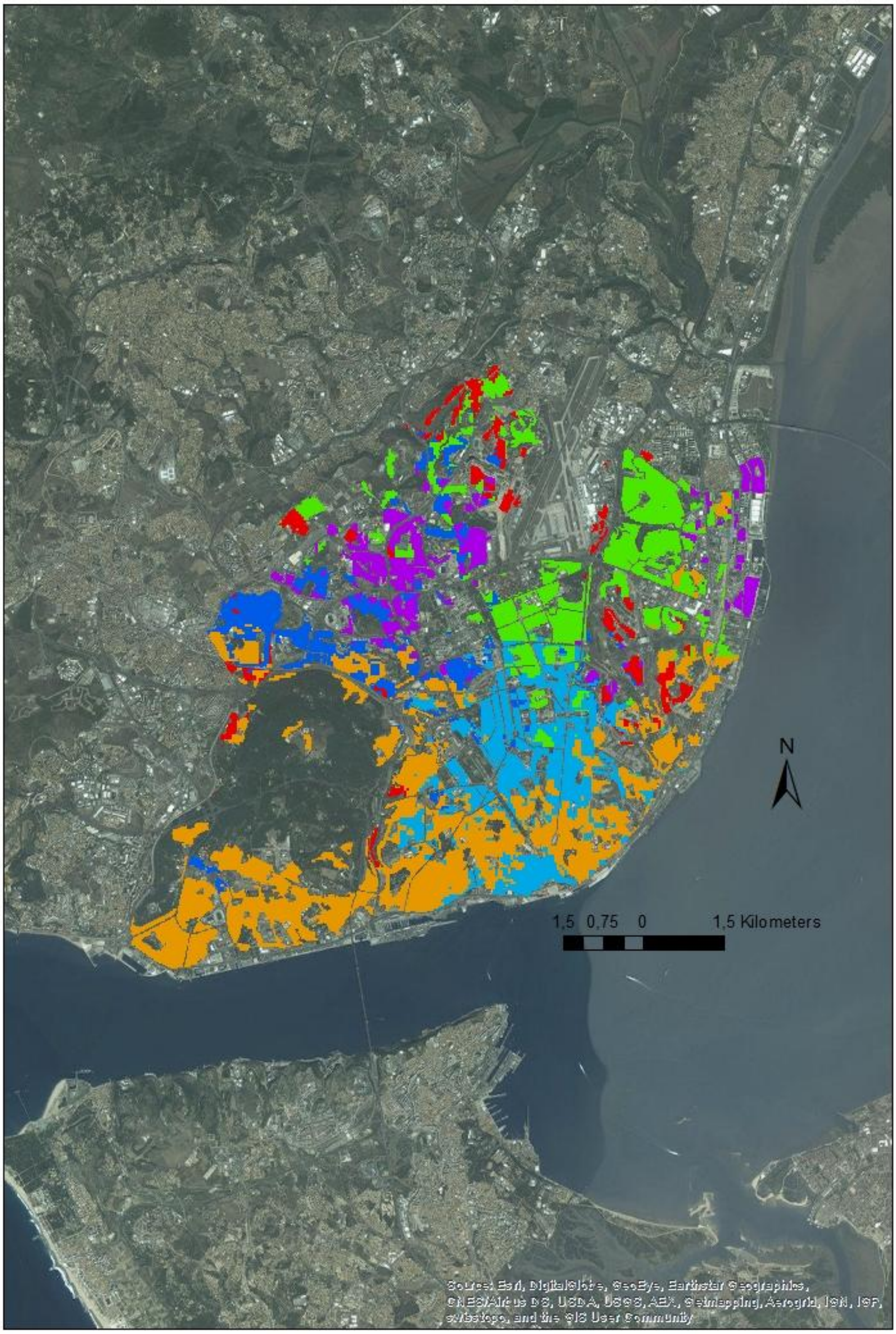
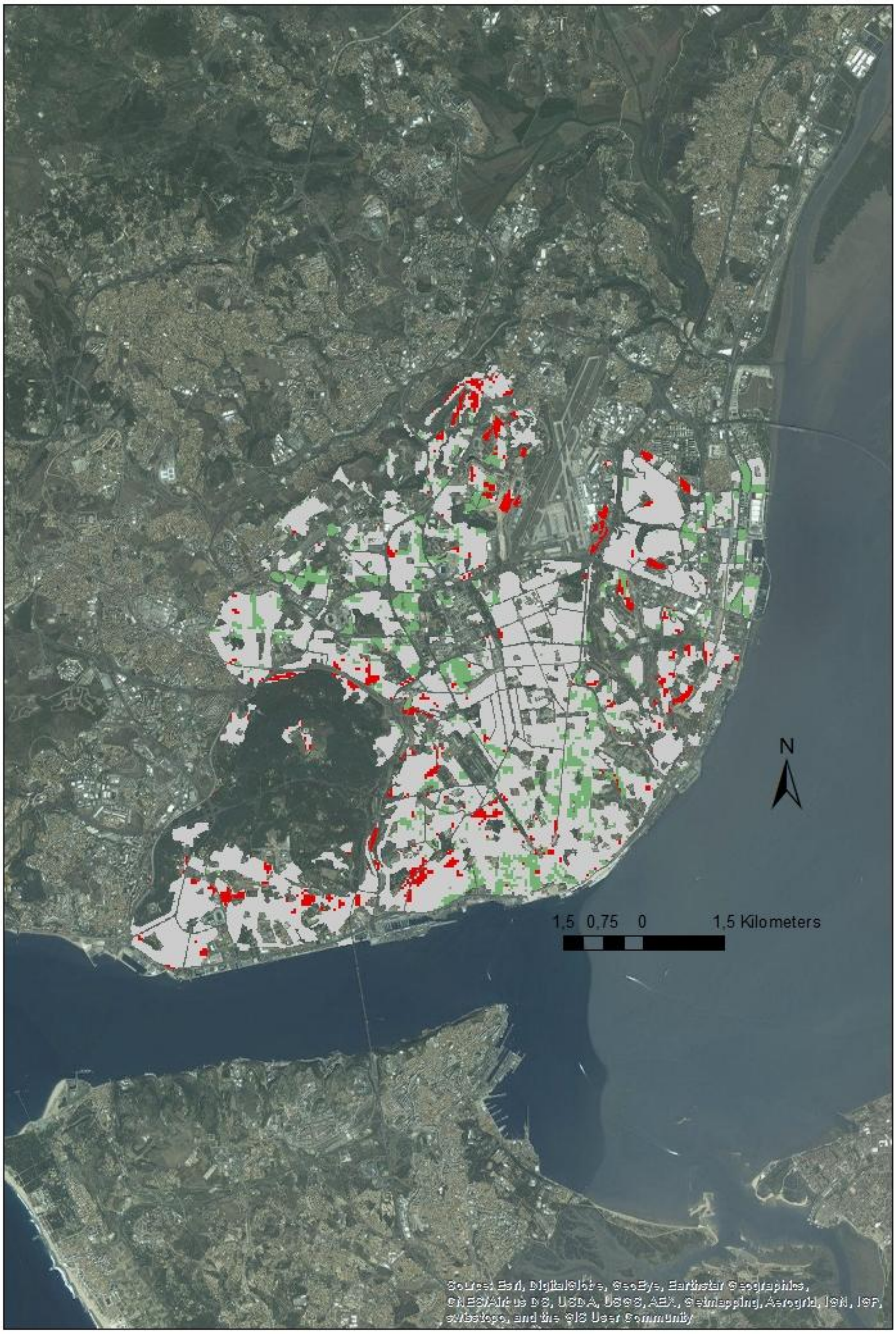

N
1,5 0,75 0 1,5 Kilometers
Source: Esri, DigitalGlobe, GeoEye, Earthstar Geographics, CNES/Airbus DS, USDA, USGS, AEX, Getmapping, Aerogrid, IGN, IGP, swisstopo, and the GIS User Community

Figure 9-36 Geo-SOM Hierarchical Clustering (6) Choropleth 2011



Source: Esri, DigitalGlobe, GeoEye, Earthstar Geographics, CNES/Airbus DS, USDA, USGS, AEX, Getmapping, Aerogrid, IGN, IGP, swisstopo, and the GIS User Community

Figure 9-37 Geo-SOM Hierarchical Clustering (6) Comparison 2001 versus 2011

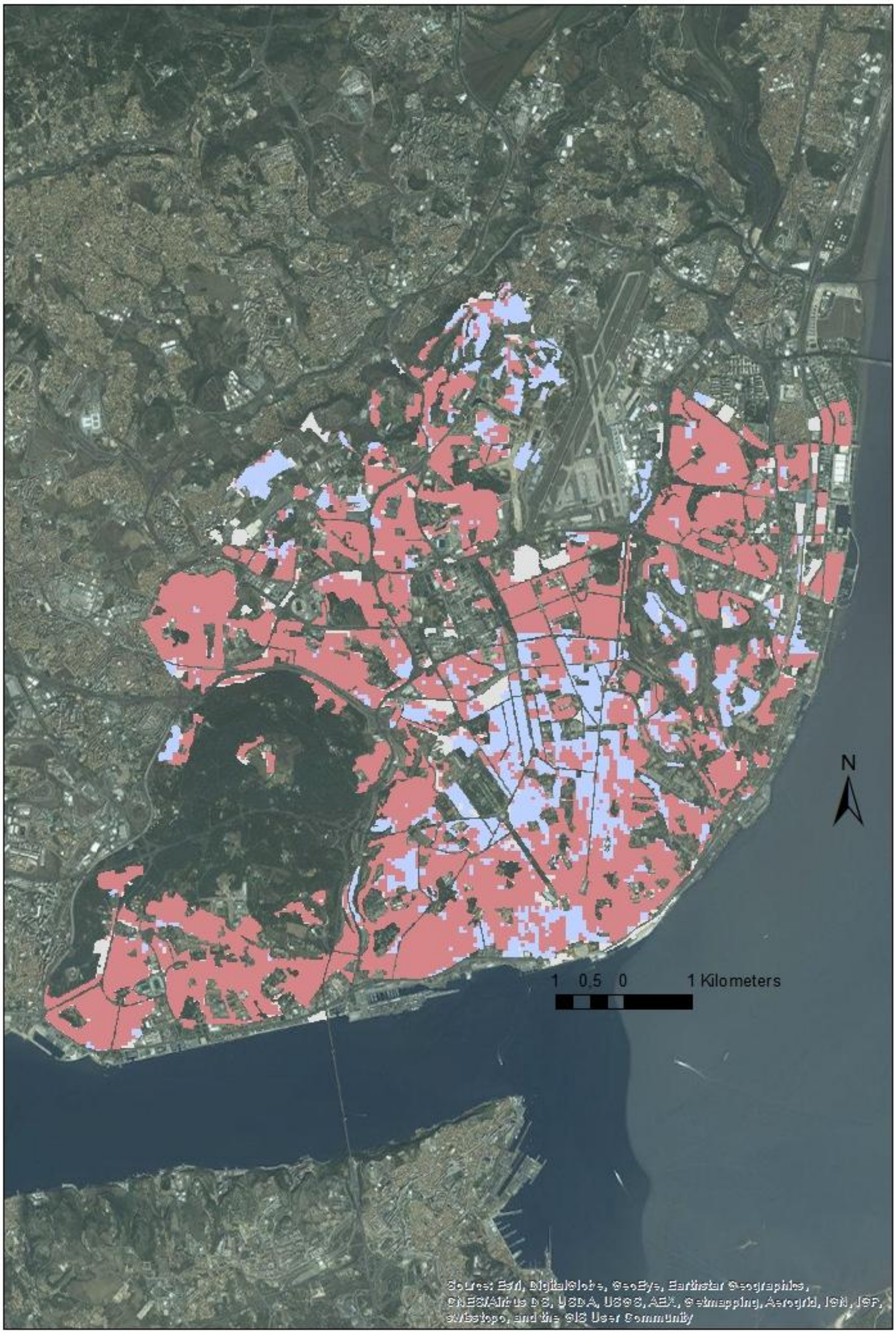

N
1 0,5 0 1 Kilometers
Source: Esri, DigitalGlobe, GeoEye, Earthstar Geographics, CNES/Airbus DS, USDA, USGS, AEX, Getmapping, Aerogrid, IGN, IGP, swisstopo, and the GIS User Community

Figure 9-38 Algorithm SOM and Geo-SOM Comparison

9.1. PROGRAMMING CODE

9.1.1. Pre-Process

```
#Clean the Workspace
rm(list = ls(all = TRUE))
name <- "Ratios"
# Set the new Working Directory
#Read all the clusters from the Umat
urbanBGRI <- read.csv("Interset_Urban_BGRI.csv",header=T, sep=";")
urbanBGRI <- subset(urbanBGRI,select=-c(wkt_geom,COSN2,AREA,PERIMETER,DTMN11,
                                FR11,SEC11,SS11,LUG11,LUG11DESIG))
library(dplyr)
#Now, BGRI subsections will have only the areas that area urban,
#so they will be decreased to the total urban subsection area (weiBGRI)
weiBGRI <- urbanBGRI %>%
    group_by(BGRI11) %>%
    summarize(totalBGRI = sum(newArea))
#Now, let's substitute the oldArea (which represents the total Area by the new one totalBGRI)
weigUrban <- merge(urbanBGRI,weiBGRI, by="BGRI11")
#Remove the unnecessary Columns
weigUrban <- subset(weigUrban,select=-c(oldArea))
#Now load the new SHP data with the grid data
#Load the data from the QGIS Pre-Process
#       The generated file with the Pixels, Subsections and applied areas
qgis <- read.csv("finalGridInters.csv",header=T, sep=";")
# 15 Variables, It is needed: ID; BGRI11[10]; Area[13]; newArea[15]
grid <- qgis[c("ID","BGRI11","newArea","smalArea")]
#Rename the Variables
```

```
names(grid) <- c("ID","GEO_COD","oldArea","newArea")
# It is necessary to load the full BGRI with all the attributes to join with the "spatial grid"
# Set the new Working Directory to search for the data
#Load BGRI
BGRI <- read.csv("BGRI2011_1106.csv",header=T, sep=";")
#Reduce the number of Variables and Rows
# Only Subsections are necessary
# "NIVEL" == 8
Dados <- BGRI[(BGRI$NIVEL==8),]
# Select the following Vars
#"GEO_COD";
#"N_EDIFICIOS_CLASSICOS"
#"N_EDIFICIOS_CLASSICOS_ISOLADOS"
#".N_EDIFICIOS_5OU_MAIS_PISOS"
#Calcular os Edificios Antes de 1980 (Soma)
#"N_EDIFICIOS_CONSTR_ANTES_1919"
#"N_EDIFICIOS_CONSTR_1919A1945"
#"N_EDIFICIOS_CONSTR_1946A1960"
#"N_EDIFICIOS_CONSTR_1961A1970"
#"N_EDIFICIOS_CONSTR_1971A1980"
#Calcular os edificios 1981 A 2000 (Soma)
#"N_EDIFICIOS_CONSTR_1981A1990"
#"N_EDIFICIOS_CONSTR_1991A1995"
#"N_EDIFICIOS_CONSTR_1996A2000"
#Calcular os edificios 2001 a 2011 (Soma)
#"N_EDIFICIOS_CONSTR_2001A2005"
#"N_EDIFICIOS_CONSTR_2006A2011"
```

```
#"N_ALOJAMENTOS"

#"N_CLASSICOS_RES_HABITUAL"

#"N_ALOJAMENTOS_RES_HABITUAL"

#"N_ALOJAMENTOS_VAGOS"

#"N_ALOJAMENTOS_FAM_CLASSICOS"

# Calcular Casas Menores 100 (Pequenas) (Soma)

#"N_RES_HABITUAL_AREA_50"

#"N_RES_HABITUAL_AREA_50_100"

#Calcular Casas Maiores 100 (Grandes) (Soma)

#"N_RES_HABITUAL_AREA_100_200"

#"N_RES_HABITUAL_AREA_200"

#Calcular Residencias Sem Estacionamento ("N_ALOJAMENTOS_FAM_CLASSICOS" -
("N_RES_HABITUAL_ESTAC_1"  + "N_RES_HABITUAL_ESTAC_2" + "N_RES_HABITUAL_ESTAC_3" ))

#"N_RES_HABITUAL_ESTAC_1"

#"N_RES_HABITUAL_ESTAC_2"

#"N_RES_HABITUAL_ESTAC_3"

#"N_RES_HABITUAL_ARREND"

#"N_FAMILIAS_CLASSICAS"

#Calcular Familias Numerosas ("N_FAMILIAS_CLASSICAS" - ("N_FAMILIAS_CLASSICAS_1OU2_PESS" +
"N_FAMILIAS_CLASSICAS_3OU4_PESS") )

#"N_FAMILIAS_CLASSICAS_1OU2_PESS"

#"N_FAMILIAS_CLASSICAS_3OU4_PESS"

#"N_INDIVIDUOS_RESIDENT"

#Calcular Individuos Jovens (soma)

#"N_INDIVIDUOS_RESIDENT_0A4"

#"N_INDIVIDUOS_RESIDENT_5A9"

#"N_INDIVIDUOS_RESIDENT_10A13"

#"N_INDIVIDUOS_RESIDENT_14A19"
```

```
#"N_INDIVIDUOS_RESIDENT_20A64"

#"N_INDIVIDUOS_RESIDENT_65"

#Calcular Ensino Básico (Soma)

#"N_IND_RESIDENT_ENSINCOMP_1BAS"

#"N_IND_RESIDENT_ENSINCOMP_2BAS"

#"N_IND_RESIDENT_ENSINCOMP_3BAS"

#Calcular Ensino Secundário (Soma)

#"N_IND_RESIDENT_ENSINCOMP_SEC"

#"N_IND_RESIDENT_ENSINCOMP_POSEC"

#"N_IND_RESIDENT_ENSINCOMP_SUP"

#Calcular Desempregados (Soma)

#"N_IND_RESID_DESEMP_PROC_1EMPRG"

#"N_IND_RESID_DESEMP_PROC_EMPRG"

Dados <- Dados[c("GEO_COD",

            "N_EDIFICIOS_CLASSICOS",

            "N_EDIFICIOS_CLASSICOS_ISOLADOS",

            ".N_EDIFICIOS_5OU_MAIS_PISOS",

            "N_EDIFICIOS_CONSTR_ANTES_1919",

            "N_EDIFICIOS_CONSTR_1919A1945",

            "N_EDIFICIOS_CONSTR_1946A1960",

            "N_EDIFICIOS_CONSTR_1961A1970",

            "N_EDIFICIOS_CONSTR_1971A1980",

            "N_EDIFICIOS_CONSTR_1981A1990",

            "N_EDIFICIOS_CONSTR_1991A1995",

            "N_EDIFICIOS_CONSTR_1996A2000",

            "N_EDIFICIOS_CONSTR_2001A2005",

            "N_EDIFICIOS_CONSTR_2006A2011",
            "N_ALOJAMENTOS",
            "N_CLASSICOS_RES_HABITUAL",
            "N_ALOJAMENTOS_RES_HABITUAL",
            "N_ALOJAMENTOS_VAGOS",
            "N_ALOJAMENTOS_FAM_CLASSICOS",
            "N_RES_HABITUAL_AREA_50",
            "N_RES_HABITUAL_AREA_50_100",
            "N_RES_HABITUAL_AREA_100_200",
            "N_RES_HABITUAL_AREA_200",
            "N_RES_HABITUAL_ESTAC_1",
            "N_RES_HABITUAL_ESTAC_2",
            "N_RES_HABITUAL_ESTAC_3",
            "N_RES_HABITUAL_ARREND",
            "N_FAMILIAS_CLASSICAS",
            "N_FAMILIAS_CLASSICAS_1OU2_PESS",
            "N_FAMILIAS_CLASSICAS_3OU4_PESS",
            "N_INDIVIDUOS_RESIDENT",
            "N_INDIVIDUOS_RESIDENT_0A4",
            "N_INDIVIDUOS_RESIDENT_5A9",
            "N_INDIVIDUOS_RESIDENT_10A13",
            "N_INDIVIDUOS_RESIDENT_14A19",
            "N_INDIVIDUOS_RESIDENT_20A64",
            "N_INDIVIDUOS_RESIDENT_65",
            "N_IND_RESIDENT_ENSINCOMP_1BAS",
            "N_IND_RESIDENT_ENSINCOMP_2BAS",
            "N_IND_RESIDENT_ENSINCOMP_3BAS",
            "N_IND_RESIDENT_ENSINCOMP_SEC",
          "N_IND_RESIDENT_ENSINCOMP_POSEC",

          "N_IND_RESIDENT_ENSINCOMP_SUP",

          "N_IND_RESID_DESEMP_PROC_1EMPRG",

          "N_IND_RESID_DESEMP_PROC_EMPRG")]

# Now let's group the variables that must be grouped

# Buildings Age "Antes 1980"

Dados$N_EDIFICIOS_CONSTR_ANTES_1980 <- Dados$N_EDIFICIOS_CONSTR_ANTES_1919 +

    Dados$N_EDIFICIOS_CONSTR_1919A1945 +

    Dados$N_EDIFICIOS_CONSTR_1946A1960 +

    Dados$N_EDIFICIOS_CONSTR_1961A1970 +

    Dados$N_EDIFICIOS_CONSTR_1971A1980

# Remove the obsolete variables

Dados <- subset(Dados,select=-c(N_EDIFICIOS_CONSTR_ANTES_1919,

                  N_EDIFICIOS_CONSTR_1919A1945,

                  N_EDIFICIOS_CONSTR_1946A1960,

                  N_EDIFICIOS_CONSTR_1961A1970,

                  N_EDIFICIOS_CONSTR_1971A1980))

# Buildings Age "1980 a 2000"

Dados$N_EDIFICIOS_CONSTR_1981A2000<-Dados$N_EDIFICIOS_CONSTR_1981A1990 +

    Dados$N_EDIFICIOS_CONSTR_1991A1995 +

    Dados$N_EDIFICIOS_CONSTR_1996A2000

# Remove the obsolete variables

Dados <- subset(Dados,select=-c(N_EDIFICIOS_CONSTR_1981A1990,

                  N_EDIFICIOS_CONSTR_1991A1995,

                  N_EDIFICIOS_CONSTR_1996A2000))

# Buildings Age "2001 a 2011"

Dados$N_EDIFICIOS_CONSTR_2001A2011<-Dados$N_EDIFICIOS_CONSTR_2001A2005 +
    Dados$N_EDIFICIOS_CONSTR_2006A2011

# Remove the obsolete variables

Dados <- subset(Dados,select=-c(N_EDIFICIOS_CONSTR_2001A2005,

                    N_EDIFICIOS_CONSTR_2006A2011))

#Houses sizes (Small menor 100m2)

Dados$N_RES_HABITUAL_AREA_PEQ <-
Dados$N_RES_HABITUAL_AREA_50+Dados$N_RES_HABITUAL_AREA_50_100

# Remove the obsolete variables

Dados <- subset(Dados,select=-c(N_RES_HABITUAL_AREA_50,

                    N_RES_HABITUAL_AREA_50_100))

#Houses sizes (Big Maior 100m2)

Dados$N_RES_HABITUAL_AREA_GRAND<-
Dados$N_RES_HABITUAL_AREA_100_200+Dados$N_RES_HABITUAL_AREA_200

# Remove the obsolete variables

Dados <- subset(Dados,select=-c(N_RES_HABITUAL_AREA_100_200,

                    N_RES_HABITUAL_AREA_200))

# Parking Slots (Lack of)

Dados$N_RES_HABITUAL_SEM_ESTAC <- Dados$N_ALOJAMENTOS_FAM_CLASSICOS -
(Dados$N_RES_HABITUAL_ESTAC_1  + Dados$N_RES_HABITUAL_ESTAC_2 +
Dados$N_RES_HABITUAL_ESTAC_3)

# Remove the obsolete variables

Dados <- subset(Dados,select=-c(N_RES_HABITUAL_ESTAC_1,

                    N_RES_HABITUAL_ESTAC_2,

                    N_RES_HABITUAL_ESTAC_3))

#Big Families (5 or more Persons)

Dados$N_FAMILIAS_CLASSICAS_5OUMAIS_PESS <- Dados$N_FAMILIAS_CLASSICAS -
(Dados$N_FAMILIAS_CLASSICAS_1OU2_PESS + Dados$N_FAMILIAS_CLASSICAS_3OU4_PESS)

# Remove the obsolete variables

Dados <- subset(Dados,select=-c(N_FAMILIAS_CLASSICAS_1OU2_PESS,
                    N_FAMILIAS_CLASSICAS_3OU4_PESS))

#Young (0 to 19)

Dados$N_INDIVIDUOS_RESIDENT_0A19 <-
Dados$N_INDIVIDUOS_RESIDENT_0A4+Dados$N_INDIVIDUOS_RESIDENT_5A9+Dados$N_INDIVIDUO
S_RESIDENT_10A13+Dados$N_INDIVIDUOS_RESIDENT_14A19

# Remove the obsolete variables

Dados <- subset(Dados,select=-c(N_INDIVIDUOS_RESIDENT_0A4,

                    N_INDIVIDUOS_RESIDENT_5A9,

                    N_INDIVIDUOS_RESIDENT_10A13,

                    N_INDIVIDUOS_RESIDENT_14A19))

#Education Primary level(s)

Dados$N_IND_RESIDENT_ENSINCOMP_BAS <- Dados$N_IND_RESIDENT_ENSINCOMP_1BAS +
Dados$N_IND_RESIDENT_ENSINCOMP_2BAS + Dados$N_IND_RESIDENT_ENSINCOMP_3BAS

# Remove the obsolete variables

Dados <- subset(Dados,select=-c(N_IND_RESIDENT_ENSINCOMP_1BAS,

                    N_IND_RESIDENT_ENSINCOMP_2BAS,

                    N_IND_RESIDENT_ENSINCOMP_3BAS))

#Education Secundary Level(s)

Dados$N_IND_RESIDENT_ENSINCOMP_SEC_POSEC <- Dados$N_IND_RESIDENT_ENSINCOMP_SEC +
Dados$N_IND_RESIDENT_ENSINCOMP_POSEC

# Remove the obsolete variables

Dados <- subset(Dados,select=-c(N_IND_RESIDENT_ENSINCOMP_SEC,

                    N_IND_RESIDENT_ENSINCOMP_POSEC))

# Unemployment

Dados$N_IND_RESID_DESEMP <- Dados$N_IND_RESID_DESEMP_PROC_1EMPRG +
Dados$N_IND_RESID_DESEMP_PROC_EMPRG

# Remove the obsolete variables

Dados <- subset(Dados,select=-c(N_IND_RESID_DESEMP_PROC_1EMPRG,

                    N_IND_RESID_DESEMP_PROC_EMPRG))
```

```
# GEO_COD of "Dados" has " ' ", let's remove it
Dados$GEO_COD <- gsub("'","",Dados$GEO_COD)
#Not forget Dados$GEO_COD and grid$BGRI belong to different classes right now
#In order to merge (inner join) the variable must have the same name
#Let's change the name of the "grid" dataframe
names(grid)[names(grid)=="BGRI11"] <- "GEO_COD"
#Merge Data to a final dataset with all Vars
merged <- merge(grid,Dados,by="GEO_COD")
# Let's select the final subset of variables to use
#"GEO_COD" <- Código da Subsecçao Geográfica
#"ID"  <- Código do Pixel na Grid
#"AREA"  <- Área total da Subsecçao "GEO_COD"
#"newArea"  <- Área da Subsecçao na Grid
#"N_EDIFICIOS_CLASSICOS" <- Edifícios classicos
#"N_EDIFICIOS_CLASSICOS_ISOLADOS"    <- Edificios clássicos isolados
#".N_EDIFICIOS_5OU_MAIS_PISOS"  <- Edifícios com 5 ou mais pisos
#"N_ALOJAMENTOS"       <- Total de Alojamentos
#"N_CLASSICOS_RES_HABITUAL"     <- Alojamentos clássicos de residência habitual
#"N_ALOJAMENTOS_RES_HABITUAL"    <- Alojamentos familiares de residência habitual
#"N_ALOJAMENTOS_VAGOS"  <- Alojamentos familiares vagos
#"N_ALOJAMENTOS_FAM_CLASSICOS"   <- Alojamentos familiares clássicos
#"N_RES_HABITUAL_ARREND"  <- Alojamentos familiares clássicos de residência habitual arrendados
#"N_FAMILIAS_CLASSICAS" <-  Total de famílias clássicas
#"N_INDIVIDUOS_RESIDENT"   <- Total de indivíduos residentes
#"N_INDIVIDUOS_RESIDENT_20A64"  <- Indíviduos residentes com idade entre 20 e 64 anos
#"N_INDIVIDUOS_RESIDENT_65"      <- Indíviduos residentes com idade superior a 64 anos
#"N_IND_RESIDENT_ENSINCOMP_SUP"      <- Indivíduos residentes com um curso superior completo
```

```
#"N_EDIFICIOS_CONSTR_ANTES_1980"     <- Edifícios construídos antes de 1981
#"N_EDIFICIOS_CONSTR_1981A2000"      <- Edifícios construídos entre 1981 e 2000
#"N_EDIFICIOS_CONSTR_2001A2011"     <- Edifícios construídos entre 2001 e 2011
#"N_RES_HABITUAL_AREA_PEQ"     <- Alojamentos familiares clássicos de residencia habitual com área até 100 m2
#"N_RES_HABITUAL_AREA_GRAND"    <- Alojamentos familiares clássicos de residencia habitual com área maior que 100 m2
#"N_RES_HABITUAL_SEM_ESTAC"    <- Alojamentos familiares clássicos de residencia habitual sem estacionamento p/ veículo
#"N_FAMILIAS_CLASSICAS_5OUMAIS_PESS"  <- Famílias clássicas com 5 ou mais pessoas
#"N_INDIVIDUOS_RESIDENT_0A19"       <- Indíviduos residentes com idade entre 0 e 19 anos
#"N_IND_RESIDENT_ENSINCOMP_BAS"      <- Indivíduos residentes com o 1º ou 2º ou 3º ciclo do ensino básico completo
#"N_IND_RESIDENT_ENSINCOMP_SEC_POSEC" <- Indivíduos residentes com o ensino secundário ou pós-secundário completo
#"N_IND_RESID_DESEMP"  <- Indivíduos residentes desempregados à procura de emprego
#Vamos eliminar as seguintes variáveis dado que potencialmente não acrescentam valor por serem redundantes
#"N_EDIFICIOS_CLASSICOS" ; "N_ALOJAMENTOS" ; "N_CLASSICOS_RES_HABITUAL" ; "N_ALOJAMENTOS_RES_HABITUAL"
merged <- subset(merged,select=-c(N_EDIFICIOS_CLASSICOS_ISOLADOS,
                        N_EDIFICIOS_CLASSICOS,
                        N_ALOJAMENTOS,
                        N_CLASSICOS_RES_HABITUAL,
                        N_ALOJAMENTOS_RES_HABITUAL))
#The contribution for each pixel from the SS (subsection) is the proportion between (Partial Area / Total Area)
#Assumption the items are uniformly distributed
#Create new variable "PROPORTION"
######ATTENTION# This part is different from the Lisbon Metropolitan Area
#merged$newArea <- as.numeric(as.character(merged$newArea))
```

```
merged$PROPORTION <- merged$newArea / merged$oldArea

#Let's create a new data.frame but with all the added attributes with the PROPORTIONAL applied

#Create the data.frame first

partialData <- merged[c(1:4,length(merged))]

#Let's add all the new Variables but already processed to the area in the pixel

#Buildings

#partialData$Ed_Isolados <- merged$PROPORTION * merged$N_EDIFICIOS_CLASSICOS_ISOLADOS

partialData$Ed_Altos <- merged$PROPORTION * merged$.N_EDIFICIOS_5OU_MAIS_PISOS

partialData$Aloj_Vagos <- merged$PROPORTION * merged$N_ALOJAMENTOS_VAGOS

partialData$Aloj_Fam <- merged$PROPORTION * merged$N_ALOJAMENTOS_FAM_CLASSICOS

partialData$Aloj_Arrendados <- merged$PROPORTION * merged$N_RES_HABITUAL_ARREND

partialData$Ed_Const_80 <- merged$PROPORTION * merged$N_EDIFICIOS_CONSTR_ANTES_1980

partialData$Ed_Const_81a00 <- merged$PROPORTION * merged$N_EDIFICIOS_CONSTR_1981A2000

partialData$Ed_Const_01a11 <- merged$PROPORTION * merged$N_EDIFICIOS_CONSTR_2001A2011

partialData$Aloj_Peq <- merged$PROPORTION * merged$N_RES_HABITUAL_AREA_PEQ

partialData$Aloj_Grand <- merged$PROPORTION * merged$N_RES_HABITUAL_AREA_GRAND

partialData$Aloj_Sem_ESTAC <- merged$PROPORTION * merged$N_RES_HABITUAL_SEM_ESTAC

#Families

partialData$Familias <- merged$PROPORTION * merged$N_FAMILIAS_CLASSICAS

partialData$Familias_Grand <- merged$PROPORTION *
merged$N_FAMILIAS_CLASSICAS_5OUMAIS_PESS

#Education

partialData$Ens_Bas <- merged$PROPORTION * merged$N_IND_RESIDENT_ENSINCOMP_BAS

partialData$Ens_Sec <- merged$PROPORTION * merged$N_IND_RESIDENT_ENSINCOMP_SEC_POSEC

partialData$Ens_Sup <- merged$PROPORTION * merged$N_IND_RESIDENT_ENSINCOMP_SUP

#Individuals

partialData$Ind_Res <- merged$PROPORTION * merged$N_INDIVIDUOS_RESIDENT

partialData$Ind_Res_0a19 <- merged$PROPORTION * merged$N_INDIVIDUOS_RESIDENT_0A19
```

```
partialData$Ind_Res_20a64 <- merged$PROPORTION * merged$N_INDIVIDUOS_RESIDENT_20A64
partialData$Ind_Res_65 <- merged$PROPORTION * merged$N_INDIVIDUOS_RESIDENT_65
partialData$Ind_Desemp <- merged$PROPORTION * merged$N_IND_RESID_DESEMP
#Now the proportional is done for each SS, lets convert this info into the pixel (summarize the data)
#Calculate the Absolute values for each pixel for each attribute
if(name=="Absolutos"){

    library(dplyr)
    Absolutos <- partialData %>%
        arrange(ID) %>%
        group_by(ID) %>%
        summarise(Area=sum(newArea),
             Ed_Isolados=sum(Ed_Isolados),
             Ed_Altos=sum(Ed_Altos),
             Aloj_Vagos=sum(Aloj_Vagos),
             Aloj_Fam=sum(Aloj_Fam),
             Aloj_Arrendados=sum(Aloj_Arrendados),
             Ed_Const_80=sum(Ed_Const_80),
             Ed_Const_81a00=sum(Ed_Const_81a00),
             Ed_Const_01a11=sum(Ed_Const_01a11),
             Aloj_Peq=sum(Aloj_Peq),
             Aloj_Grand=sum(Aloj_Grand),
             Aloj_Sem_ESTAC=sum(Aloj_Sem_ESTAC),
             Familias=sum(Familias),
             Familias_Grand=sum(Familias_Grand),
             Ens_Bas=sum(Ens_Bas),
             Ens_Sec=sum(Ens_Sec),
                Ens_Sup=sum(Ens_Sup),

                Ind_Res=sum(Ind_Res),

                Ind_Res_0a19=sum(Ind_Res_0a19),

                Ind_Res_20a64=sum(Ind_Res_20a64),

                Ind_Res_65=sum(Ind_Res_65),

                Ind_Desemp=sum(Ind_Desemp))


    #Now we must print out the new dataset to be used in the Geo-SOM suite

    #Change the Working Directory

    setwd("C:/Users/Jorge/OneDrive/Mestrado/gridSHP")

    #Save the file

    write.table(Absolutos,paste0("InputGrid_Abs",Sys.Date(),".txt"), sep=";")

}

#Para o caso de querermos densidades

if(name=="Densidade"){

    library(dplyr)

    Densidades <- partialData %>%

        arrange(ID) %>%

        group_by(ID) %>%

        summarise(Area=sum(newArea),

                Ed_Isolados=sum(Ed_Isolados)/sum(newArea),

                Ed_Altos=sum(Ed_Altos)/sum(newArea),

                Aloj_Vagos=sum(Aloj_Vagos)/sum(newArea),

                Aloj_Fam=sum(Aloj_Fam)/sum(newArea),

                Aloj_Arrendados=sum(Aloj_Arrendados)/sum(newArea),

                Ed_Const_80=sum(Ed_Const_80)/sum(newArea),

                Ed_Const_81a00=sum(Ed_Const_81a00)/sum(newArea),
                Ed_Const_01a11=sum(Ed_Const_01a11)/sum(newArea),
                Aloj_Peq=sum(Aloj_Peq)/sum(newArea),
                Aloj_Grand=sum(Aloj_Grand)/sum(newArea),
                Aloj_Sem_ESTAC=sum(Aloj_Sem_ESTAC)/sum(newArea),
                Familias=sum(Familias)/sum(newArea),
                Familias_Grand=sum(Familias_Grand)/sum(newArea),
                Ens_Bas=sum(Ens_Bas)/sum(newArea),
                Ens_Sec=sum(Ens_Sec)/sum(newArea),
                Ens_Sup=sum(Ens_Sup)/sum(newArea),
                Ind_Res=sum(Ind_Res)/sum(newArea),
                Ind_Res_0a19=sum(Ind_Res_0a19)/sum(newArea),
                Ind_Res_20a64=sum(Ind_Res_20a64)/sum(newArea),
                Ind_Res_65=sum(Ind_Res_65)/sum(newArea),
                Ind_Desemp=sum(Ind_Desemp)/sum(newArea))

    #Now we must print out the new dataset to be used in the Geo-SOM suite
    #Change the Working Directory
    setwd("C:/Users/Jorge/OneDrive/Mestrado/gridSHP")
    #Save the file
    write.table(Densidades,paste0("InputGrid_Dens",Sys.Date(),".txt"), sep=";")


}
#Para o caso de querermos RATIOS
if(name=="Ratios"){


    library(dplyr)
    Ratios <- partialData %>%
        arrange(ID) %>%

        group_by(ID) %>%

        summarise(Area=sum(newArea),

#Ed_Isolados=sum(Ed_Isolados)/(sum(Ed_Const_80)+sum(Ed_Const_81a00)+sum(Ed_Const_01a11)),

Build5Floo=sum(Ed_Altos)/(sum(Ed_Const_80)+sum(Ed_Const_81a00)+sum(Ed_Const_01a11)),

                EmptyAccom=sum(Aloj_Vagos)/sum(Aloj_Fam),

                #Aloj_Fam=sum(Aloj_Fam)/sum(newArea),

                Rent_Accom=sum(Aloj_Arrendados)/sum(Aloj_Fam),

Buil80=sum(Ed_Const_80)/(sum(Ed_Const_80)+sum(Ed_Const_81a00)+sum(Ed_Const_01a11)),

Buil81to00=sum(Ed_Const_81a00)/(sum(Ed_Const_80)+sum(Ed_Const_81a00)+sum(Ed_Const_01a11)),

#Ed_Const_01a11=sum(Ed_Const_01a11)/(sum(Ed_Const_80)+sum(Ed_Const_81a00)+sum(Ed_Const_01a11)),

                #Aloj_Peq=sum(Aloj_Peq)/sum(Aloj_Fam),

                #Aloj_Grand=sum(Aloj_Grand)/sum(Aloj_Fam),

                #Aloj_Sem_ESTAC=sum(Aloj_Sem_ESTAC)/sum(Aloj_Fam),

                #Familias=sum(Familias)/sum(newArea),

                Famil5plus=sum(Familias_Grand)/sum(Familias),

                Edu_Prim=sum(Ens_Bas)/(sum(Ens_Bas)+sum(Ens_Sec)+sum(Ens_Sup)),

                Edu_Sec=sum(Ens_Sec)/(sum(Ens_Bas)+sum(Ens_Sec)+sum(Ens_Sup)),

                Edu_Tert=sum(Ens_Sup)/(sum(Ens_Bas)+sum(Ens_Sec)+sum(Ens_Sup)),

                #Ind_Res=sum(Ind_Res)/sum(newArea),

                Age0to19=sum(Ind_Res_0a19)/sum(Ind_Res),

                Age20to64=sum(Ind_Res_20a64)/sum(Ind_Res),

                Age65=sum(Ind_Res_65)/sum(Ind_Res),
            Unemploy=sum(Ind_Desemp)/sum(Ind_Res),
            Dens_Pop=sum(Ind_Res)/sum(newArea),
            Dens_Accom=sum(Aloj_Fam)/sum(newArea),
            Resident=sum(Ind_Res))


    #Attention Ratios can generate NAs (Let's remove them)

    good <- complete.cases(Ratios)

    Ratios <- Ratios[good,]

    #At least "1" human must leave at this subsection of 2500m^2
    Ratios2011 <- subset(Ratios,Resident>1)
    Ratios2011 <- subset(Ratios2011,select=-c(Area,Resident))

    #Now we must print out the new dataset to be used in the Geo-SOM suite
    #Change the Working Directory
    setwd("C:/Users/Jorge/OneDrive/Documentos/GISApp/dasymetric/PreProcess2011")

    #Save the file
    write.table(Ratios2011,paste0("Ratios11_",Sys.Date(),".txt"), sep=";")
}
```

9.1.2. Hierarchical Clustering

```
#Clean Workspace
rm(list = ls(all = TRUE))
# Set the new Working Directory
```

```
#Read all the clusters from the Umat
umat <- read.csv("umat_G3.csv",header=T, sep=";")
names(umat) <- c("X","Y",
          "Build5Floo","EmptyAccom","Rent.Accom",
          "Buil80","Build81to00","Famil5plus","Edu.Prim","Edu.Sec","Edu.Tert",
          "Age0to19","Age20to64","Age65","Unemploy","Dens.Pop","Dens.Accom")
#The cluster will be the row.name
for(i in 1:nrow(umat)){
    rownames(umat)[i] <- paste0("Clu",i)
}
#Remove the cluster column to not enter into the calculation
#umat <- subset(umat,select=-c(Cluster))
#Perform the HC
hc = hclust(dist(umat,method="euclidean"),method = "ward.D")
#Plot it
dendrogram <- as.dendrogram(hc)
plot(hc,labels=FALSE,hang=-1)
abline(h=7,lty=2,col="red")
#Let's get the clusters
grid.cluster <- as.matrix(cutree(hc, k = 6),ncol=2)
grid.cluster <- as.data.frame(grid.cluster)
#Now we need to join datasets to obtain the Hierarchical Clusters applied to the grid
#rename the grid.cluster and aplly the cluster number in a fashion way
colnames(grid.cluster) <- "HierClus"
for (i in 1:nrow(grid.cluster)){
    grid.cluster$Clust[i] <- i
}

#To obtain the order for each 300 cluster

ordemSOM <- order.dendrogram(dendrogram)

write.table(ordemSOM,paste0("OrdemG4_",Sys.Date(),".txt"), sep=";")

#To create a shapefile we'll print out

#setwd("C:/Users/Jorge/OneDrive/Dresden/2011/GEOSOM/ClustResults/SOMuMAT")

write.table(grid.cluster,paste0("G3_HC_300",Sys.Date(),".txt"), sep=";")

#Obtain the Distance Matrix

#centroides_dist <- read.csv("HC6_SOM_Description.csv",header=T, sep=";")

#centroides_dist <- subset(centroides_dist,select=-c(HierClus))

#distancias <- dist(centroides_dist, method = "euclidean", diag = FALSE, upper = FALSE, p = 2)

#Now we must get the full data from each cluster

#Create 1:1 Clusters and import the data

#setwd("C:/Users/Jorge/OneDrive/Dresden/2011/GEOSOM")

fullData <- read.csv("G3_300Clust.txt",header=T, sep=";")

fullData <- fullData[c("X.ID.","X.Clust.")]

names(fullData) <- c("ID","Clust")

#Join both datasets per Clust

gridHier <- merge(fullData,grid.cluster,by="Clust")

#now 6 Clusters

write.table(gridHier,paste0("gridHier_6_",Sys.Date(),".txt"), sep=";")


#It is necessary to obtain the Hierarchical Clusters description

#Let's load again the umat, but keep the Cluster number

#setwd("C:/Users/Jorge/OneDrive/Dresden/2011/GEOSOM/Codebook")

#Read all the clusters from the Umat

umat <- read.csv("umat_G3.csv",header=T, sep=";")
```

```
for (i in 1:nrow(umat)){
umat$Clust[i] <- i
}
names(umat) <- c("X","Y",
                "Build5Floo","EmptyAccom","Rent.Accom",
                "Build80","Build81to00","Famil5plus","Edu.Prim","Edu.Sec","Edu.Tert",
                "Age0to19","Age20to64","Age65","Unemploy","Dens.Pop","Dens.Accom","Clust")
clusDescr <- merge(umat,grid.cluster,by="Clust")
library(dplyr)
HierClusters <- clusDescr %>%
        group_by(HierClus) %>%
        summarize(X=mean(X),
                  Y=mean(Y),
                  build5Floo=mean(Build5Floo),
                  emptyAccom=mean(EmptyAccom),
                  rent_Accom=mean(Rent.Accom),
                  buil80=mean(Build80),
                  buil81to00=mean(Build81to00),
                  famil5plus=mean(Famil5plus),
                  edu_Prim=mean(Edu.Prim),
                  edu_Sec=mean(Edu.Sec),
                  edu_Tert=mean(Edu.Tert),
                  age0to19=mean(Age0to19),
                  age20to64=mean(Age20to64),
                  age65=mean(Age65),
                  unemploy=mean(Unemploy),
                  dens_Pop=mean(Dens.Pop),
          dens_Accom=mean(Dens.Accom))

#Print out the HClus description
```

9.1.3. 2001 Classification

```
#Clean Workspace

rm(list = ls(all = TRUE))

#Determinar Working Directory

library(dplyr)

##Aqui definir carregar os dados de 2001 relativos ao municipio de Lisboa

BGRI <- read.csv("bgri2001_1106_vCompleta.csv",header=T, sep=";")

#Muitas variáveis e poucas de interesse e é necessario concatenar o BGRI

Dados <- subset(BGRI,select=-
c(NUT1_EU02,NUT2_EU02,NUT3_EU02,NUT1_EU02_DSG,NUT2_EU02_DSG,NUT3_EU02_DSG,

              DISTRITO,CONCELHO,FREGUESIA,

              TTE,ER,PR,PNR,PV2,PV4,EBAR,EARG,EPAT,EORE,

              TTA,AFRHEL,AFRHAG,AFRHRE,AFRHES,AFRHBN,

              AFCRH1_2D,AFCRH3_4D,AFCRHPO,

              AC,AFC,AFNC,

              TTFI,FCD_0,FCD_1,FCPME15,FCPMA65,

              TTNFR,NF_1FNC,NF_2FNC,NF_1NNC,NF_2NNC,NFF6,NGN6,


TTP,TTHR,TTMR,TTHP,TTMP,HR15_19,MR15_19,HR20_24,MR20_24,HR25_64,MR25_64,

              IRQA_001,IRNI_413,IRNI_423,IRNI_433,IRNI_513,IRNI_713,

              IRP_TCR,IRP_ECR,

              IR_SP,IR_SS,IR_ST,IR_PR,IR_EP,IR_SAC))


#Now, create the BGRI field with dummy data

Dados$GEO_COD <- 1:nrow(Dados)
```

```
#Convert the necessary fields to character
Dados$DD_EU02 <- as.character(Dados$DD_EU02)
Dados$CC_EU02 <- as.character(Dados$CC_EU02)
Dados$FF_EU02 <- as.character(Dados$FF_EU02)
Dados$SECCAO <- as.character(Dados$SECCAO)
Dados$SUBSECCAO <- as.character(Dados$SUBSECCAO)
#Now fill the missing zeros
Dados$CC_EU02 <- paste0("0",Dados$CC_EU02)
for (i in 1:nrow(Dados)){
    #To Freguesia
    if (nchar(Dados$FF_EU02)[i]==1){
        Dados$FF_EU02[i] <- paste0("0",Dados$FF_EU02[i])
    }
    #To Secçao
    if (nchar(Dados$SECCAO)[i]==1){
        Dados$SECCAO[i] <- paste0("00",Dados$SECCAO[i])
    }
    if (nchar(Dados$SECCAO)[i]==2){
        Dados$SECCAO[i] <- paste0("0",Dados$SECCAO[i])
    }
    #To Subsection
    if (nchar(Dados$SUBSECCAO)[i]==1){
        Dados$SUBSECCAO[i] <- paste0("0",Dados$SUBSECCAO[i])
    }
}
#Create the final GEO_COD
Dados$GEO_COD <-
paste0(Dados$DD_EU02,Dados$CC_EU02,Dados$FF_EU02,Dados$SECCAO,Dados$SUBSECCAO)
```

```
Dados <- subset(Dados,select=-c(DD_EU02,CC_EU02,FF_EU02,SECCAO,SUBSECCAO))

# Now let's group the variables that must be grouped

# Buildings Age "Antes 1980"

Dados$N_EDIFICIOS_CONSTR_ANTES_1980 <- Dados$E1919 + Dados$E1945 + Dados$E1960 +
Dados$E1970 + Dados$E1980

# Remove the obsolete variables

Dados <- subset(Dados,select=-c(E1919,E1945,E1960,E1970,E1980))

# Buildings Age "1981 a 2000"

Dados$N_EDIFICIOS_CONSTR_1981A2000<-Dados$E1985 + Dados$E1990 + Dados$E1995 +
Dados$E2001

# Remove the obsolete variables

Dados <- subset(Dados,select=-c(E1985,E1990,E1995,E2001))

#Big Families (5 or more Persons)

Dados$N_FAMILIAS_CLASSICAS_5OUMAIS_PESS <- Dados$TTFC - (Dados$FCR1_2 + Dados$FCR3_4)

# Remove the obsolete variables

Dados <- subset(Dados,select=-c(FCR1_2,FCR3_4))

#Total Individual Residents

Dados$N_INDIVIDUOS_RESIDENT <- (Dados$HR0_4 + Dados$MR0_4 + Dados$HR5_9 +
Dados$MR5_9 + Dados$HR10_13 + Dados$MR10_13 +

    Dados$HR14_19 + Dados$MR14_19 +

    Dados$HR20_64 + Dados$MR20_64 +

    Dados$HR65 + Dados$MR65)


#Young (0 to 19)

Dados$N_INDIVIDUOS_RESIDENT_0A19 <- (Dados$HR0_4 + Dados$MR0_4 + Dados$HR5_9 +
Dados$MR5_9 + Dados$HR10_13 + Dados$MR10_13 +

    Dados$HR14_19 + Dados$MR14_19)

# Remove the obsolete variables
```

```
Dados <- subset(Dados,select=-
c(HR0_4,MR0_4,HR5_9,MR5_9,HR10_13,MR10_13,HR14_19,MR14_19))

#Labour Force (20 to 64)

Dados$N_INDIVIDUOS_RESIDENT_20A64 <- (Dados$HR20_64 + Dados$MR20_64)

# Remove the obsolete variables

Dados <- subset(Dados,select=-c(HR20_64,MR20_64))

#Senior (>65)

Dados$N_INDIVIDUOS_RESIDENT_65 <- (Dados$HR65 + Dados$MR65)

# Remove the obsolete variables

Dados <- subset(Dados,select=-c(HR65,MR65))

#Education Primary level(s)

Dados$N_IND_RESIDENT_ENSINCOMP_BAS <- Dados$IRQA_110 + Dados$IRQA_120 +
Dados$IRQA_130

# Remove the obsolete variables

Dados <- subset(Dados,select=-c(IRQA_110,IRQA_120,IRQA_130))

#Education Secundary Level(s)

Dados$N_IND_RESIDENT_ENSINCOMP_SEC_POSEC <- Dados$IRQA_200 + Dados$IRQA_300

# Remove the obsolete variables

Dados <- subset(Dados,select=-c(IRQA_200,IRQA_300))

#Education Teritary Level

Dados$N_IND_RESIDENT_ENSINCOMP_SUP <- Dados$IRQA_400

# Remove the obsolete variables

Dados <- subset(Dados,select=-c(IRQA_400))

# Unemployment

Dados$N_IND_RESID_DESEMP <- Dados$IR_D1E + Dados$IR_DNE

# Remove the obsolete variables

Dados <- subset(Dados,select=-c(IR_D1E,IR_DNE))
```

```
Dados$N_EDIFICIOS_CLASSICOS <- Dados$TTEC
Dados$N_EDIFICIOS_5OU_MAIS_PISOS  <- Dados$PV5
#Alojamentos Familiares de Residencia habitual
Dados$N_ALOJAMENTOS_FAM_CLASSICOS <- Dados$AFRH
Dados$N_ALOJAMENTOS_RES_HABITUAL <- Dados$AFCRH
Dados$N_RES_HABITUAL_ARREND <- Dados$AFCRHARR
Dados$N_ALOJAMENTOS <- Dados$AF
Dados$N_ALOJAMENTOS_VAGOS <- Dados$AFV
Dados$N_FAMILIAS_CLASSICAS <- Dados$TTFC
Dados <- subset(Dados,select=-c(TTEC,PV5,AFRH,AFCRH,AFCRHARR,AF,AFV,TTR,TTFC))
##NOW WE HAVE THE BGRI DEFINED#
#Obtain the Coordinates of the ID
#This step is only necessary for the GeoSOM approach algorithm
setwd("C:/Users/Jorge/OneDrive/Documentos/GISApp/dasymetric/PreProcess2011")
coord <- read.csv("IDCoord.csv",header=T, sep=";")
coord <- coord[c("ID","X","Y")]
# Set the new Working Directory
setwd("C:/Users/Jorge/OneDrive/Documentos/GISApp/dasymetric/2001/Intersect01")
#Load the data from the QGIS Pre-Process
#	The generated file with the Pixels, Subsections and applied areas
qgis <- read.csv("intersect01_Tbl.csv",header=T, sep=";")
# a lot Variables,
grid <- qgis[c("ID","BGRI2001","newArea")]
#Transform the variable "newArea"
grid$newArea <- as.numeric(as.character(grid$newArea))
library(dplyr)
#Now, BGRI subsections will have only the areas that area urban (already selected in the 2011 dataset),
#so they will be decreased to the total urban subsection area (weiBGRI)
#In this case just sum all the "newArea"
weiBGRI <- grid %>%
    group_by(BGRI2001) %>%
    summarize(totalBGRI = sum(newArea))
#Now, let's substitute the oldArea (which represents the total Area by the new one totalBGRI)
weigUrban <- merge(grid,weiBGRI, by="BGRI2001")
#In order to merge (inner join) the variable must have the same name
#Let's change the name of the "grid" dataframe
names(weigUrban)[names(weigUrban)=="BGRI2001"] <- "GEO_COD"
#Merge Data to a final dataset with all Vars
merged <- merge(weigUrban,Dados,by="GEO_COD")
#Now we can create a PROPORTION based on partial Areas
#merged$newArea <- as.numeric(as.character(merged$newArea))
merged$PROPORTION <- merged$newArea / merged$totalBGRI
#Let's create a new data.frame but with all the added attributes with the PROPORTIONAL applied
#Create the data.frame first
partialData <- merged[c(1:4,length(merged))]
#Let's add all the new Variables but already processed to the area in the pixel
#Buildings
partialData$Total_Aloj <- merged$PROPORTION * merged$N_ALOJAMENTOS
partialData$Ed_Altos <- merged$PROPORTION * merged$N_EDIFICIOS_5OU_MAIS_PISOS
partialData$Aloj_Vagos <- merged$PROPORTION * merged$N_ALOJAMENTOS_VAGOS
partialData$Aloj_Fam <- merged$PROPORTION * merged$N_ALOJAMENTOS_FAM_CLASSICOS
partialData$Aloj_Arrendados <- merged$PROPORTION * merged$N_RES_HABITUAL_ARREND
partialData$Ed_Const_80 <- merged$PROPORTION * merged$N_EDIFICIOS_CONSTR_ANTES_1980
partialData$Ed_Const_81a00 <- merged$PROPORTION * merged$N_EDIFICIOS_CONSTR_1981A2000
```

```
#Families
partialData$Familias <- merged$PROPORTION * merged$N_FAMILIAS_CLASSICAS
partialData$Familias_Grand <- merged$PROPORTION *
merged$N_FAMILIAS_CLASSICAS_5OUMAIS_PESS
#Education
partialData$Ens_Bas <- merged$PROPORTION * merged$N_IND_RESIDENT_ENSINCOMP_BAS
partialData$Ens_Sec <- merged$PROPORTION * merged$N_IND_RESIDENT_ENSINCOMP_SEC_POSEC
partialData$Ens_Sup <- merged$PROPORTION * merged$N_IND_RESIDENT_ENSINCOMP_SUP
#Individuals
partialData$Ind_Res <- merged$PROPORTION * merged$N_INDIVIDUOS_RESIDENT
partialData$Ind_Res_0a19 <- merged$PROPORTION * merged$N_INDIVIDUOS_RESIDENT_0A19
partialData$Ind_Res_20a64 <- merged$PROPORTION * merged$N_INDIVIDUOS_RESIDENT_20A64
partialData$Ind_Res_65 <- merged$PROPORTION * merged$N_INDIVIDUOS_RESIDENT_65
partialData$Ind_Desemp <- merged$PROPORTION * merged$N_IND_RESID_DESEMP
#We want the data per ID and Ratios
library(dplyr)
Ratios2001 <- partialData %>%
    arrange(ID) %>%
    group_by(ID) %>%
    summarise(Area=sum(newArea),
        Build5Floo=sum(Ed_Altos)/(sum(Ed_Const_80)+sum(Ed_Const_81a00)),
        EmptyAccom=sum(Aloj_Vagos)/sum(Aloj_Fam),
        Rent_Accom=sum(Aloj_Arrendados)/sum(Aloj_Fam),
        Buil80=sum(Ed_Const_80)/(sum(Ed_Const_80)+sum(Ed_Const_81a00)),
        Buil81to00=sum(Ed_Const_81a00)/(sum(Ed_Const_80)+sum(Ed_Const_81a00)),
        Famil5plus=sum(Familias_Grand)/sum(Familias),
        Edu_Prim=sum(Ens_Bas)/(sum(Ens_Bas)+sum(Ens_Sec)+sum(Ens_Sup)),
        Edu_Sec=sum(Ens_Sec)/(sum(Ens_Bas)+sum(Ens_Sec)+sum(Ens_Sup)),
        Edu_Tert=sum(Ens_Sup)/(sum(Ens_Bas)+sum(Ens_Sec)+sum(Ens_Sup)),
        Age0to19=sum(Ind_Res_0a19)/sum(Ind_Res),
        Age20to64=sum(Ind_Res_20a64)/sum(Ind_Res),
        Age65=sum(Ind_Res_65)/sum(Ind_Res),
        Unemploy=sum(Ind_Desemp)/sum(Ind_Res),
        Dens_Pop=sum(Ind_Res)/sum(newArea),
        Dens_Accom=sum(Aloj_Fam)/sum(newArea),
        Resident=sum(Ind_Res))
#Attention Ratios can generate NAs (Let's remove them)
good <- complete.cases(Ratios2001)
Ratios2001 <- Ratios2001[good,]
#1 Human must leave per "pixel"
Ratios2001 <- subset(Ratios2001,Resident>1)
Ratios2001 <- subset(Ratios2001,select=-c(Area,Resident))
#Only use this option when Geo-SOM algorithm
newGRatios <- merge(Ratios2001,coord,by="ID")
rm("Ratios2001")
Ratios2001 <- newGRatios
rm("newGRatios")
#Normalize the data
#Now we must import the clusters to atribute each cluster to each ID
#Let's find them
setwd("C:/Users/Jorge/OneDrive/Documentos/GISApp/dasymetric/GeoSOM/G3")
#Here are the clusters
clusters <- read.csv("HC_DescrG3_Coord_6_2015-11-17.txt",header=T, sep=";")
```

```
#Reorder the variables
clusters <- subset(clusters,select=c(1,4:18,2,3))
#Normalize must keep the maximum and minimum to Normalize the Clusters
#RatiosMAX <- apply(Ratios2001,2,max)
#RatiosMIN <- apply(Ratios2001,2,min)
#Ratios2001
for (p in 2:length(Ratios2001)){
    Ratios2001[p] <- (Ratios2001[p]-min(Ratios2001[p]))/(max(Ratios2001[p])-min(Ratios2001[p]))
}
#Clusters is different
#for (p in 2:length(clusters)){
 #          clusters[p] <- (clusters[p]-RatiosMIN[p])/(RatiosMAX[p]-RatiosMIN[p])
  #     }
##Iniciar para o número de clusters(j)
#Remove ID and Area
classify <- select(Ratios2001, -(c(1)))
##Definir numero de variaveis (Isto é necessário para manter os dados originais e tratamento)
varNum <- as.integer(length(classify))
#Este valor é necessário sempre que se faz o reset so ciclo
Keeps <- c(1:varNum)
#Normalize the data min-max or Range [0;1]
#for (p in 1:length(dados)){
 #      dados[p] <- (dados[p]-min(dados[p]))/(max(dados[p])-min(dados[p]))
  #     clusters[p+1] <- (clusters[p+1]-min(clusters[p+1]))/(max(clusters[p+1])-min(clusters[p+1]))
#}
```

```
for(j in 1:nrow(clusters)){


    ##O ciclo for vai ter em conta que as variáveis estão em letras

    ##O i vai até ao número de colunas (variáveis)

    ##length - 1 -> pk a ultima variavel é o GEO_COD

    for(i in 1:length(classify)){

        ##Calcular a distância


        z <- (classify[c(i)]-clusters[[j,i+1]])^2

        ##Agrupar as distânicas à data.frame inicial

        classify <- cbind(classify,z)

        ##Atribuir novos nomes (neste caso números porque são muitas variáveis)

        names(classify)[length(classify)] <- i

    }

    ##o w serve para ter a Table das distâncias de forma singular(entre cada elemento)

    w<-classify[c((length(classify)-varNum):length(classify))]


    ##Calcular a soma das distâncias


    somaDist <- as.data.frame(rowSums(w))

    somaDist <- somaDist^(1/2)


    #Remover o w(temporário)

    rm("w")

    #Reset aos dados

    classify <-classify[Keeps]

    ##Aqui, gerar uma nova grelha com as colunas de distancias ao vários clusters
```

```
	assign(paste0("cluster",j),somaDist)

	rm("somaDist")

}
##Agora já temos as distancias de cada elemento aos clusters
##Criar Table com todos eles
#resumoDist <- cbind(cluster1,cluster2,cluster3,cluster4,cluster5)
#colnames(resumoDist) <- c("dist1","dist2","dist3","dist4","dist5")
if(nrow(clusters)==2){
	resumoDist <- cbind(cluster1,cluster2)
	colnames(resumoDist) <- c("dist1","dist2")
} else if (nrow(clusters)==3){
	resumoDist <- cbind(cluster1,cluster2,cluster3)
	colnames(resumoDist) <- c("dist1","dist2","dist3")
} else if (nrow(clusters)==4){
	resumoDist <- cbind(cluster1,cluster2,cluster3,cluster4)
	colnames(resumoDist) <- c("dist1","dist2","dist3","dist4")
} else if (nrow(clusters)==5){
	resumoDist <- cbind(cluster1,cluster2,cluster3,cluster4,cluster5)
	colnames(resumoDist) <- c("dist1","dist2","dist3","dist4","dist5")
} else if (nrow(clusters)==6){
	resumoDist <- cbind(cluster1,cluster2,cluster3,cluster4,cluster5,cluster6)
	colnames(resumoDist) <- c("dist1","dist2","dist3","dist4","dist5","dist6")
} else {
	resumoDist <- cbind(cluster1,cluster2,cluster3,cluster4,cluster5,cluster6,cluster7)
```

```
        colnames(resumoDist) <- c("dist1","dist2","dist3","dist4","dist5","dist6","dist7")
}
minDist <- as.data.frame(apply(resumoDist,1,which.min))
##Associar Cluster
clustering2001 <- cbind(Ratios2001$ID,minDist)
colnames(clustering2001) <- c("ID","Cluster")
##for (i in 1:nrow(clustering2001)){
#       if(clustering2001$CluFinal[i]==1){
#               clustering2001$SubCluster[i] <- "10"
#       } else if (clustering2001$CluFinal[i]==2) {
#               clustering2001$SubCluster[i] <- "11"
 #      } else if (clustering2001$CluFinal[i]==3) {
  #            clustering2001$SubCluster[i] <- "5"
   #    } else if (clustering2001$CluFinal[i]==4) {
    #          clustering2001$SubCluster[i] <- "6"
     #   } else if (clustering2001$CluFinal[i]==5) {
      #         clustering2001$SubCluster[i] <- "9"
       # } else {
        #       clustering2001$SubCluster[i] <- "0"
        #}
#}
#Compare 2001 with 2011
HC2011 <- read.csv("HC_G3.csv",header=T, sep=";")
#Give appropriate names
names(clustering2001) <- c("ID","HC01")
HC2011 <- subset(HC2011,select=-c(Clust))
names(HC2011) <- c("ID","HC11")
```

```
compare <- merge(clustering2001,HC2011,by="ID",all=TRUE)

for (i in 1:nrow(compare)){

    if (is.na(compare$HC01)[i]){

        compare$HC01[i]=0

    }

}

nice <- complete.cases(compare)

compare <- compare[nice,]

for (i in 1:nrow(compare)){

    #2011 -> 5 The best

    if(compare[i,3]==5 & (compare[i,2]==2 | compare[i,2]==6 | compare[i,2]==1 | compare[i,2]==4 |
compare[i,2]==0 )){

        compare$Eval[i]="Better"

    }

    else if(compare[i,3]==1 & (compare[i,2]==5 | compare[i,2]==3)){

        compare$Eval[i]="Same"

    }

    #2011 -> 3 2nd Best

    else if(compare[i,3]==3 & (compare[i,2]==6 | compare[i,2]==1 | compare[i,2]==4 |
compare[i,2]==0 )){

        compare$Eval[i]="Better"

    }

    else if(compare[i,3]==3 & (compare[i,2]==5 | compare[i,2]==3 | compare[i,2]==2)){

        compare$Eval[i]="Same"

    }

    #2011 -> 2 3rd Best

    else if(compare[i,3]==2 & (compare[i,2]==1 | compare[i,2]==4 | compare[i,2]==0)){
        compare$Eval[i]="Better"

    }

    else if(compare[i,3]==2 & (compare[i,2]==2 | compare[i,2]==3 | compare[i,2]==6)){

        compare$Eval[i]="Same"

    }

    else if(compare[i,3]==2 & (compare[i,2]==5)){

        compare$Eval[i]="Worst"

    }

    #2011 -> 6 4rd Best

    else if(compare[i,3]==6 & (compare[i,2]==4)){

        compare$Eval[i]="Better"

    }

    else if(compare[i,3]==6 & (compare[i,2]==2 | compare[i,2]==6 | compare[i,2]==1 |
compare[i,2]==0)){

        compare$Eval[i]="Same"

    }

    else if(compare[i,3]==6 & (compare[i,2]==5 | compare[i,2]==3)){

        compare$Eval[i]="Worst"

    }

    #2011 -> 1 5rd Best

    else if(compare[i,3]==1 & (compare[i,2]==6 | compare[i,2]==1 | compare[i,2]==4)){

        compare$Eval[i]="Same"

    }

    else if(compare[i,3]==1 & (compare[i,2]==5 | compare[i,2]==3 | compare[i,2]==2 |
compare[i,2]==0)){

        compare$Eval[i]="Worst"

    }

    #2011 -> 4 Last
```

```
        else if(compare[i,3]==4 & (compare[i,2]==1 | compare[i,2]==4)){

                compare$Eval[i]="Same"

        }

        else if(compare[i,3]==4 & (compare[i,2]==5 | compare[i,2]==3 | compare[i,2]==2 |
compare[i,2]==6 | compare[i,2]==0)){

                compare$Eval[i]="Worst"

        }

}


setwd("C:/Users/Jorge/OneDrive/Documentos/GISApp/dasymetric/2001/Classify/G3")


write.table(compare, paste0("compare_G3_",Sys.Date(),".txt"), sep=";")
```

9.1.4. Geo-SOM K Selection

```
rm(list = ls(all = TRUE))

# Define Working Directory

library(dplyr)

centroides <- read.csv("HC_DescrSOM_6_2015-11-16.txt",header = T, sep = ';')

centroides <- subset(centroides,select=c(2:16,1))

## It's necessary to create subsets for each Hierarchical cluster

#Must load 2 datasets

#grid -> cluster,ID and HC

#300Clust -> Cluster and ID

grid <- read.csv("gridHier_6_2015-11-16.txt",header = T, sep = ';')

allData <- read.csv("SOM_300Clust.txt",header = T, sep = ';')

#Remove the undesirable variables to process the error

allData <- subset(allData,select=-c(X.OBJECTID.,X.Shape_Leng.,X.Shape_Area.,

                                X.X1.,X.Y1.,X))
```

```
#Provide accurate names
names(allData) <- c("ID",
        "Build5Floo","EmptyAccom","Rent.Accom",
        "Buil80","Build81to00","Famil5plus","Edu.Prim","Edu.Sec","Edu.Tert",
        "Age0to19","Age20to64","Age65","Unemploy","Dens.Pop","Dens.Accom",
        "Clust")

#Join both datasets per ID
gridHier <- merge(allData,grid,by="ID")
gridHier <- subset(gridHier,select=-c(Clust.y))
#Reorder the data
gridHier <- subset(gridHier,select=c(2:16,1,17,18))
#Let's generate "subsets" of each cluster
for (t in 1:nrow(centroides)){
    subset <- filter(gridHier, HierClus==t)
    assign(paste("grupo",t,sep=""),subset)
    rm("subset")
}
varsGrupo <- length(grupo1)
#Calcular as distancias e respectivos erros
for (j in 1:nrow(centroides)){

    for (i in 1:(length(centroides)-1)){
        if(j==1){
            z <- (abs(grupo1[,i]-centroides[[j,(i)]]))^2
            grupo1 <- cbind(grupo1,z)
            names(grupo1)[length(grupo1)]<-i
            distancias1<-grupo1[(varsGrupo+1):length(grupo1)]
        }
        if(j==2){
            z <- (abs(grupo2[,i]-centroides[[j,(i)]]))^2
            grupo2 <- cbind(grupo2,z)
            names(grupo2)[length(grupo2)]<-i
            distancias2<-grupo2[(varsGrupo+1):length(grupo2)]
        }
        if(j==3){
            z <- (abs(grupo3[,i]-centroides[[j,(i)]]))^2
            grupo3 <- cbind(grupo3,z)
            names(grupo3)[length(grupo3)]<-i
            distancias3<-grupo3[(varsGrupo+1):length(grupo3)]
        }
        if(j==4){
            z <- (abs(grupo4[,i]-centroides[[j,(i)]]))^2
            grupo4 <- cbind(grupo4,z)
            names(grupo4)[length(grupo4)]<-i
            distancias4<-grupo4[(varsGrupo+1):length(grupo4)]
        }
        if(j==5){
            z <- (abs(grupo5[,i]-centroides[[j,(i)]]))^2
            grupo5 <- cbind(grupo5,z)
            names(grupo5)[length(grupo5)]<-i
            distancias5<-grupo5[(varsGrupo+1):length(grupo5)]
        }
        if(j==6){
```

```
            z <- (abs(grupo6[,i]-centroides[[j,(i)]]))^2

            grupo6 <- cbind(grupo6,z)

            names(grupo6)[length(grupo6)]<-i

            distancias6<-grupo6[(varsGrupo+1):length(grupo6)]

        }


    }


}


totalErro <- sum(distancias1,distancias2,distancias3,distancias4,distancias5,distancias6)

quadro <- rbind(sum(distancias1),sum(distancias2),sum(distancias3),sum(distancias4),sum(distancias5),sum(distancias6),totalErro)

colnames(quadro)<-c("Erro")

rownames(quadro)<-c("1","2","3","4","5","6","total")

nomeErro <- paste0("Erro_",Sys.Date(),".txt")

write.table(quadro, nomeErro, sep=";")
```

9.1.1. Geo-SOM versus SOM

```
#Clean Workspace

rm(list = ls(all = TRUE))

#Compare SOM with Geo-SOM (2011 datasets clustering's)

#Obtain Geo-SOM classification

Geo3SOM <- read.csv("gridHier_6_2015-11-16.txt",header=T, sep=";")

Geo3SOM <- subset(Geo3SOM,select=-c(Clust))

Geo_DC <- read.csv("HC_DescrG3_6_2015-11-16.txt",header=T, sep=";")


setwd("C:/Users/Jorge/OneDrive/Documentos/GISApp/dasymetric/GeoSOM/SOM")
```

```
SOM <- read.csv("gridHC_6.csv",header=T, sep=";")

SOM <- subset(SOM,select=-c(Clust))

SOM_DC <- read.csv("HC_DescrSOM_6_2015-11-16.txt",header=T, sep=";")

#Give appropriate names

names(Geo3SOM) <- c("ID","Geo")

names(SOM) <- c("ID","SOM")

compareSOMGEO <- merge(SOM,Geo3SOM,by="ID",all=TRUE)

distancias <- rdist((SOM_DC),(Geo_DC))

for (i in 1:nrow(compareSOMGEO)){

    if(compareSOMGEO[i,2]==compareSOMGEO[i,3]){

        compareSOMGEO$Eval[i]="Same"

    }

    else compareSOMGEO$Eval[i]="Different"

}

setwd("C:/Users/Jorge/OneDrive/Documentos/GISApp/dasymetric/Input")

write.table(compareSOMGEO,paste0("SOMvsGEO_",Sys.Date(),".txt"), sep=";")

write.table(compare, paste0("compare_G3_",Sys.Date(),".txt"), sep=";")
```