POLITECNICO DI MILANO

SCUOLA DI INGEGNERIA INDUSTRIALE E DELL'INFORMAZIONE

Corso di Laurea Magistrale in Ingegneria Matematica

TESI DI LAUREA MAGISTRALE

Real and digital cities: Biclustering of Milan Neighbourhoods and Foursquare Venue Categories

Relatore: **Prof. Simone Vantini**

Candidato:
Jacopo Di Iorio
Matr. 823381

Anno Accademico 2014 - 2015

Now I see at last, Kostya, that in our kind of work, whether we're writers or actors, the important thing is not fame, or glory, not what I used to dream about, but learning how to endure.
I must bear my cross, and have faith.
If I have faith, it doesn't hurt so much, and when I think of my calling I'm not afraid of life.

Anton Checov - The Seagull

Abstract

Urbanscope is a project by Politecnico di Milano which aims at creating a macroscope of Milan able to describe how the city is lived by using big data deriving from social media. As the first main original contribution of this work, a new visualisation tool for the "City magnets" section of the Urbanscope website has been created. The purpose is to identify the neighbourhoods of Milan, called NILs, which are perceived similarly according to some venue categories. Data derive from Foursquare, a social networking system that enables a user to share their location with friends via the "check-in", thus providing valuable information regarding the most attractive categories in every NIL.

Precisely, after a review of the principal methods proposed in literature, it has been decided to perform the Cheng and Church's Biclustering algorithm in order to create the new website section. The application of this method and in general of Biclustering on social media data could be considered a unicum in the whole literature which is principally based on gene expression data. In order to have understandable biclusters, different Clustering procedures helped to describe the evolution of Milan month by month and to perform a dimensional reduction of the large number of categories. Therefore, it was possible to see how the popularity of NILs of Milan is highly correlated to the events they host.

In the end, the resulting biclusters had to be processed because of the wide and diversified audience of the Urbanscope project which aims at being comprehensible and interesting to everyone. Therefore, last efforts were made to reorganise biclusters in a better visualisation characterised by a user-friendly interface and interactive features.

Sommario

Urbanscope è un progetto del Politecnico di Milano che mira alla creazione di un macroscopio della città, cioè un insieme di strumenti in grado di descrivere come Milano sia vissuta e percepita grazie all'analisi di big data provenienti dai social media. Questa tesi rientra nell'obiettivo di arricchimento del macroscopio per mezzo di nuove visualizzazioni. In questo caso si è lavorato alla creazione della sezione "City magnets". L'obiettivo era di identificare quali zone di Milano, dette NIL, sono percepite in maniera simile per quanto riguarda alcune categorie di venues. I dati derivano da Foursquare, una piattaforma social che permette ai suoi utenti di condividere la propria posizione con gli amici per mezzo di "check-in". Il social network mette quindi a disposizione delle informazioni molto utili riguardo a quali siano le categorie di venues maggiormente attrattive in ogni NIL di Milano.

Precisamente, dopo una revisione dei principali metodi proposti dalla letteratura, è stato deciso di applicare il Cheng and Church's Biclustering algorithm in modo tale da creare la nuova sezione del sito internet. L'applicazione di questo metodo e in generale di qualsiasi altro algoritmo di Biclustering su dati di tipo social potrebbe essere considerata un unicum all'interno dell'intera letteratura, principalmente basata su problemi di tipo genetico. Prima di intraprendere lo studio con tale tecnica, però, diversi approcci di Clustering sono stati d'aiuto per descrivere la dinamica evolutiva della città di Milano mese dopo mese e per ridurre l'elevato numero di categorie. Quindi, è stato possibile avere la conferma di come la popolarità dei NIL di Milano sia altamente correlata con gli eventi che essi ospitano.

Infine, i bicluster risultanti, troppi farraginosi per un pubblico non esperto, sono stati modificati in modo tale da risultare comprensibili a tutti i possibili utenti del sito di Urbanscope. Di conseguenza, tutti gli ultimi sforzi sono stati effettuati nella direzione di riorganizzare i bicluster in una migliore visualizzazione, caratterizzata da un'interfaccia naturale da usare e il più possibilmente interattiva.

Contents

List of Figures

List of Tables

Introduction

Understanding in which way a city is changing is not a trivial task. In fact, cities can apparently be immutable reflecting the solid marble façades of their buildings and, at the same time, change convulsively. In fact, citizens can change their habits and, accordingly, the essence of the whole city is transformed. Therefore, cities are not mere physical and organisational structures but they are also invisible and existing essence. Today, however, thanks to the fact that everyone started speaking "social" by sharing contents online, a large amount of data expressing the essence of a city is available.

The Urbanscope project, powered by Politecnico di Milano, aims at creating and divulging privileged views of Milan using social data in order to foster comprehension and decision making. The study proposed in this thesis was set out to enlarge the Urbanscope macroscope by creating the "Analyse" section of the "City magnets" module whose goal is to reveal which are the attractive zones in Milan, by considering Foursquare data.

Foursquare is a social networking system that allows its users to share their positions with friends, via the "check-in". Precisely, the position can be selected by the user from a list of venues that the application locates nearby. Every venue is characterised by two features: geo-reference and category. Every venue is geo-referenced using the names of the neighbourhoods, chosen by the municipality of Milan. Instead, categories group venues according to their function.

Using Foursquare data, it has been decided to perform a Biclustering algorithm in order to indentify biclusters, i.e. groups of similar NILs according to groups of categories. As one of the main innovations of this work, the application of Biclustering techniques to social data is an innovative exemplar in the whole literature, which is principally grounded on genetic problems.

Therefore, in Chapter 1, the reader will be acquainted with Biclustering methods. The concept of Biclustering is introduced thanks to its connections with Clustering algorithms and to some practical examples. Then a general framework is given in order to have a formal mathematical point of view. After that, an analysis of Biclustering is illustrated according to the already extensive literature on Biclustering algorithms. The main methods are presented and classified along two dimensions: structures and typologies.

Due to the fact that it is necessary to select an algorithm in order to identify biclusters, it has been decided to use for the analysis the Cheng and Church's Biclustering algorithm. In Chapter 2, the popular Biclustering algorithm presented by Yizong Cheng and George M. Church in 2000 is explained in details. Starting from the historical and applicative setting that characterised the method, the reader will understand why, still today, the paper is considered the most important piece of literature in the gene expression biclustering field.

To complete the range of tools useful to understand the development of this work, Chapter 3 introduces the reader to the Urbanscope project and all the elements composing it. Precisely all the generalities, the aim of the macroscope, the internal subdivision in modules and in "Explore" and "Analyse" sections, are presented in order to show the structure of Urbanscope and the importance of studying the evolution of the city using social digital trace. Finally, after skimming all the visualisation tools that the website already presents, the reader is invited to consider the "City magnets" lens. In fact, this thesis aims at creating a brand new analysis for "City magnets" module by implementing its "Analyse" section thanks to Biclustering procedures. However, before applying the method described in Chapter 2, it is necessary to understand the raw data and to process them in order to reduce dimensions and to have the more convenient data structure of the array 3D.

Chapter 4, therefore, explains to the reader everything concerning data and its processing. The dataset is explained item by item and then processed in order to have a more convenient data structure called array 3D. Then, thanks to a vectorial clustering approach, it is modified by cutting down all the categories resulting not important to describe the urban dynamics. Moreover, the study of both NILs and categories is carried out and used as cornerstone to synthesise and interpret data.

Finally, in Chapter 5 it is explained how the notions and procedures previously introduced lead to the completion of this thesis: the creation of the "Analyse" section in the Urbanscope "City Magnets" module. Due to the fact that Urbanscope is an educational project for everyone's use, it is important to remark how it is necessary to intersect many different fields of knowledge in order to build a comprehensible and easy-to-read tool. Therefore, after the conclusion of the analytical trial by performing the Cheng and Church's Biclsutering algorithm, results are synthesised from a managerial and visual point of view in order to create a prototype of the "Analyse" section.

Chapter 1

Biclustering

Wolde you bothe eate your cake,
and have your cake?

John Heywood

In this chapter the reader will be acquainted with Biclustering methods. In Section 1.1 the concept of Biclustering is introduced thanks to its relationship with Clustering algorithms and to some practical examples. Then a general framework is given in order to have a formal mathematical point of view.

1.1 An Introduction to Biclustering

In this first section the concept of Biclustering is presented starting from a qualitative explanation of its connection with Clustering algorithms in 1.1.1. In Section 1.1.2 the reader will face a real problem where Biclustering methods represent an efficient solution. After these clarifications, the original genetical issue that gives birth to the majority of Biclustering algorithms is explained (Section 1.1.3) while in Section 1.1.4 and in Section 1.1.5 the reader is invited to consider a general framework and an important connection between Biclustering and Graph Theory.

1.1.1 From Clustering to Biclustering

One of the fundamental needs in data mining is to group a given set of objects according to some measure of similarity. This deceptively simple problem has given rise to a wide number of algorithms and methods and the most famous of them are known under the name of Clustering Methods, usually applicable to data arranged in a data matrix.

The main element of Clustering is the similarity between rows and columns in the data matrix and, in the first case, it is often a function of the row vectors involved, while, in the second case, it is a function of the column vectors. Any such formula leads to the discovery of some similarity group at the expense of obscuring some other similarity groups: having row groups, for example, means the impossibility of having column similarity groups. In fact Clustering can be applied to either rows or columns, implicitly directing the analysis to a particular aspect of the system under the study. Furthermore, exception due to Overlapping Clustering, these algorithms seek a disjoint cover of the set of elements, requiring that no column or row element belongs to more than one cluster group at the same time.

Therefore the application of standard Clustering methods to some non trivial kind of problems can lead to frugal and inefficient results. For this reason, a number of algorithms that perform simultaneous Clustering on both the dimensions of the data matrix has been proposed under the name of Biclustering Methods or, simply, Biclustering.

This new notion gives rise to a more flexible computational framework: the bicluster, a submatrix, that is, a subgroup of rows and a subgroup of columns, where the row elements exhibit highly correlated activities for every column element. Therefore, Biclustering approaches overcome some problems associated with traditional Clustering methods, by allowing simultaneous clustering of rows and columns and overlapped grouping that provides a better representation for row elements usually described by a multitude of column elements.

Just to reassure the reader about the real utility of this newly introduced method, an easy and funny example, inspired by the one proposed in *An Introduction to Biclustering* by Kemal Eren, is shown in the next paragraph.

1.1.2 Throwing a Party

Bruce Wayne, great businessman and part-time superhero, is planning a house-warming party for his new three-ballroom mansion. Each ballroom has a separate hi-tech sound system so he wants to play different music in each room. As a conscientious host, Bruce wants everyone to enjoy the music. Therefore, the best way not to seem impolite is to distribute albums and guests to each room in order to be sure that each guest hears their favourite songs.

This time, for his exclusive party, Bruce has invited only fifty guests. Considering that he owns only 30 albums, he asked to his loyal butler Alfred to send to each guest a survey in order to understand if they like or dislike each album. After receiving their responses, all the data has been recorded into a 50 x 30 binary matrix $\mathbf{M}$, where $M_{ij} = 1$ if guest i likes album j.

To ensure that everyone is happy with the music, Bruce decides to distribute people and albums evenly among the three ballrooms of his house. However due to guests' demands, in each room there should be enough albums to avoid repetitions. Concisely, Bruce would like to define three different biclusters describing the situation in each ballroom, that is, the guests and the selected albums.

Unfortunately for him, Bruce does not know anything about Biclustering so he will try to group his guests according to their musical tastes by maximizing the following objective function:

$$s(\mathbf{M}, \mathbf{r}, \mathbf{c}) = b(\mathbf{r}, \mathbf{c}) \sum_{i,j,k} M_{ij} r_{ki} c_{kj} \tag{1.1}$$

where r_{kj}, a binary variable, explains if guest i is in cluster k, c_{kj}, a binary variable too, explains if album j is in cluster k and $b \in [0, 1]$ penalizes unbalanced solutions, i.e. those who give rise to biclusters of different size. In fact, Mr. Wayne does really want to give to each group of guests enough space to dance. Therefore, b decays as the difference in sizes between the largest and the smallest bicluster grows:

$$b(\mathbf{r}, \mathbf{c}) = \exp\left(\frac{-(\max(\Omega) - \min(\Omega))}{\varepsilon}\right)$$

where ε is used to set the aggressiveness of the penalty and Ω is a set of bicluster sizes.

Bruce decides to maximize the objective function, proposed at (1.1), using the following approach: starting with random clusters, characterized by a random assignment of rows and columns, he reassigns rows and columns trying to reach the convergence of the objective function.

However, the strategy proposed by Mr. Wayne is still naive because it does not guarantee an optimal solution and it could require trying every possible combination of clusters, resulting $k^{n+p} = 3^{80}$ candidate solutions. Our party planner and part-time secret superhero has a new villain in town: the NP-complexity of Biclustering problems.

Assuming that a nice solution is found, it is possible to explain the biclusters obtained by reordering all the columns and the rows of the data matrix (survey response) to show the assignment of guests and albums to rooms. In Figure 1.1 an example of the original data set and the clusters that Bruce found is shown. Bruce already knows that his exigent guests are really strict about music but some of them could love even some tracks belonging to other ballrooms. Therefore, Mr. Wayne admits the possibility of some overlapping situation, i.e. a guest who appreciates the tracklists or part of the tracklists proposed in more than one room.

Considering that some white square are still present in the three clusters showed in Figure 1.1 it is necessary to accept that not everyone will enjoy every album, however this solution ensures that most guests will enjoy most albums.

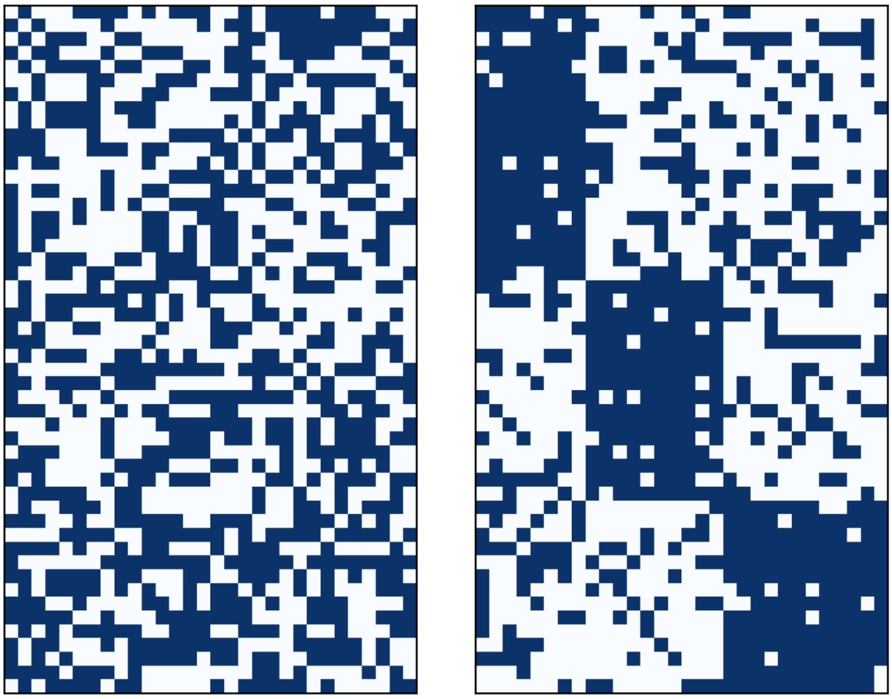


Figure 1.1: The original data matrix compared to the biclustered one. In both graphs every row represents a different guest while every column represents a different album. A blue square is representative to the pleasure of guest i for album j. Courtesy of Kemal Eren in An Introduction to Biclustering.

1.1.3 Gene Expression Data

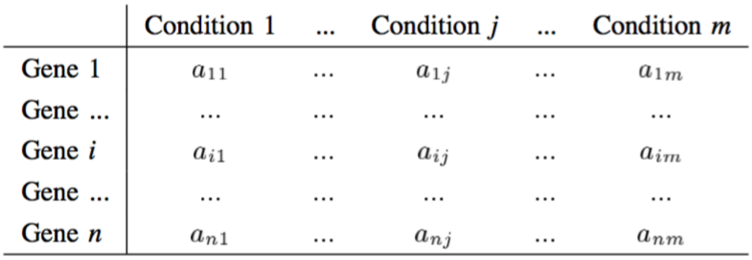
In Section 1.1.2 a contrived example explained the utility of Biclustering techniques but these methods are not limited to throwing great parties. In fact, any data which can be represented as a matrix is amenable to Biclustering.

Everytime there could be an interest in finding homogenous groups of objects, the method of Biclustering could be the right choice: examples from different application areas include market data, in which the task is to group customers (rows) under product features (columns), and text mining, where the aim is to group documents (rows) under words (columns).

The principal area of application for Biclustering, however, is gene expression data. Gene expression data are being generated by DNA chips and microarray techniques able to measure the expression level of a large number of genes within different experimental samples or conditions. These latter ones include different time points, different environmental conditions, different organs, from toxic or healthy tissues, or even different individuals or different species. They can be considered as the molecular fingerprint of tissues or cells in different biological states.

The typical representation is a data matrix, where each row corresponds to a different gene and each column to a different condition. Each element is a real number representing the expression level of a gene under a specific situation, obtained as the logarithm of the mRNA measurement.

To clear everything up, the common gene expression presents the expression level of gene i under condition j in the elements a_{ij}, which is in the intersection of row i and column j of the matrix A. An example is shown in Figure 1.2.



	Condition 1	...	Condition *j*	...	Condition *m*
Gene 1	a_{11}	...	a_{1j}	...	a_{1m}
Gene ...	...	...	...	...	...
Gene *i*	a_{i1}	...	a_{ij}	...	a_{im}
Gene ...	...	...	...	...	...
Gene *n*	a_{n1}	...	a_{nj}	...	a_{nm}

Figure 1.2: Every row i corresponds to a different gene while every column to a particular condition. The expression level of gene i under condition j is the element a_{ij}

It is therefore evident how gene expression data are a powerful source of information and have revolutionised the way to study and to understand functions in biological systems.

To deeply analyse all the informations given by these typologies of data, the well-known problem of the simultaneous clustering of objects and the selection of variables for each cluster, firstly proposed by Hartigan (1972) without a solution, became fashionable again. In fact, thanks to the application of Biclustering, whose seminal paper was written by Cheng and Church (2000) , biologists are able to identify molecular fingerprints that can help with the classification and diagnosis of the patient status and to guide treatment protocols. The usual goal is to group gene expressions under multiple conditions and to understand the functions of each gene, due to the fact that genes with similar expression patterns are likely to be regulated by the same factors and, therefore, may share similar functions. In this way, it is possible to link all the information about the function of a known gene to an unknown gene in the same cluster.

1.1.4 A general framework

Even if the majority of applications of Biclustering algorithms deals with gene expression matrices, it is evident that there are many other fields where the methods here introduced could be a great solution. Therefore, transcending from the aspiration of saving humanity or becoming the best party planner in the world, the general case of a data matrix is presented in this subsection. The data here considered are arranged in a n by m matrix called A, whose elements a_{ij} are, in general, given real values. A can also be denoted as (X, Y) where $X = \{x_1, \ldots, x_n\}$ is its set of rows and $Y = \{y_1, \ldots, y_m\}$ is its set of columns. Taking $I \subseteq X$ a subset of the rows set X and $J \subseteq Y$ a subset of the columns set Y, it is natural to identify the submatrix of the data matrix A as (I, J), the couple of I and J. Therefore $A_{IJ} = (I, J)$ denotes the sub-matrix containing only the elements a_{ij} belonging to the sub-matrix with set of rows I and set of columns J.

This notation, presented in many Biclustering algorithms, can be really helpful to understand the difference between the result of Clustering and Biclustering techniques. For example a cluster of rows is a subset of rows that exhibits similar behaviours across the set of all columns according to some distance criterion. A possible representation is given by $A_{IY} = (I, Y)$ which is a subset of rows defined over the set of all columns Y, where $I = \{i_1, \ldots, i_k\}$ is a subset of rows ($I \subseteq X$ and $k \leq n$). Thus, a cluster of rows can be defined as a $k < n$ by m sub-matrix of the original data matrix A.

The consideration just presented can be similarly made for a cluster of columns. It can be represented as $A_{XJ} = (X, J)$, a subset of columns defined over the set of all rows X, where $J = \{j_1, \ldots, j_s\}$ is a subset of columns ($J \subseteq Y$ and $s \leq m$). A cluster of columns can thus be defined as a n by $s < m$ sub-matrix of the original data matrix A.

In Figure 1.3, examples of cluster of rows and columns are showed.

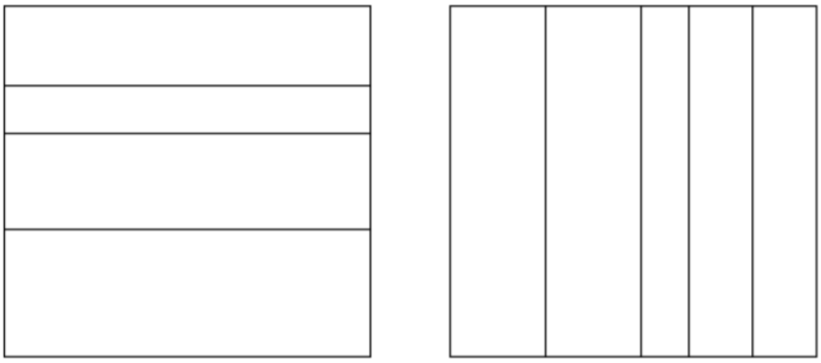


Figure 1.3: The visual difference of a matrix clustered by rows (left) and by columns (right)

Differently from clusters, a bicluster is a subset of rows that exhibits similar behaviour across a subset of columns, and vice-versa. It can be expressed as a sub-matrix of the original data matrix A. Precisely, the bicluster $A_{IJ} = (I, J)$ is a subset of rows and a subset of columns where $I = \{i_1, \ldots, i_k\}$ is a subset of rows ($J \subseteq Y$ and $s \leq m$), and $J = \{j_1, \ldots, j_s\}$ is a subset of rows ($J \subseteq Y$ and $s \leq m$). It is evident that it can be defined as a k by s sub-matrix of the data matrix A. Many different types of biclusters are possible and they will be introduced in the next section.

Independently from their typologies, the submatricial structure of biclusters allows to better define the specific result of these algorithms. The aim, as previously implied, is to identify a set of biclusters $B_k = (I_k, J_k)$ such that each singular element of B_k satisfies some characteristic of homogeneity or some measure of quality. Therefore it is now evident the sub-matricial nature of the results.

1.1.5 A Bipartite Graph Point of View

Considering the fact that the majority of algorithms are based on graph optimization, it is necessary to establish an important and interesting connection between data matrices and graph theory. A data matrix can be converted into a bipartite weighted graph. In graph theory a graph is a representation of a set of objects where some pairs of objects are connected by links. The interconnected objects are represented by mathematical abstractions called vertices (also called nodes or points), and the links that connect some pairs of vertices are called edges (also called arcs or lines).

A graph $G = (V, E)$, where V is the set of vertices and E is the set of edges, is said to be bipartite if its vertices can be partitioned into two sets U and D such that every edge in E has exactly one end in U and the other in $D : V = U \cup D$. In Figure 1.4 there is an example of bipartite graph: it is easy to notice that there are two disjoint sets of vertices with no edges within sets.

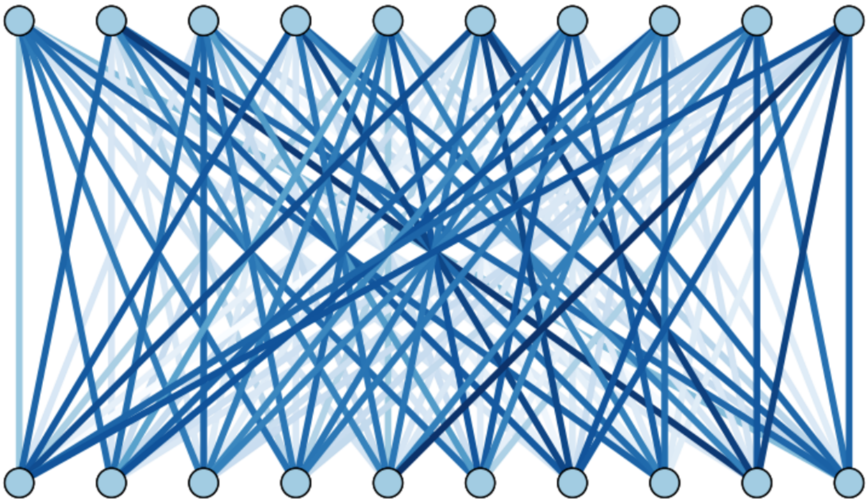


Figure 1.4: In a bipartite graph the distinction between the two sets of nodes U and D is very neat. Higher weight are visually represented with a stronger hue of blue

In order to pass from a matrix representation to a graph one, every row element was turned in a node $n_i \in U$ and every column element in a node $n_j \in D$ while the edge between node n_i and node n_j has weight a_{ij} denoting the element of the matrix in the intersection between row i and column j. For example, in the case of gene expression data, the nodes in U are the genes, the nodes in D are the samples, the weight on every edge is the gene expression level. Similarly, in the Bruce Wayne's problem the two set of nodes are respectively made by guests and albums, while the weights can be only *ones* or *zeros*: 1 if guest i loves album j and 0 otherwise.

To find biclusters it is necessary to partition the graph so that edges within partitions have heavy weights and edges between partitions have light weights.

1.2 Structures and Types

In this section an analysis of Biclustering is illustrated. According to the already extensive literature on Biclustering algorithms, it is important to structure the analysis following some criterion. Therefore in this section it has been decided to classify all the methods along two dimensions:

- The structure of biclusters - considering number and overlapping strategies;
- The type of biclusters - considering the type of homogeneity sought;

In the first subsection an analysis of the structure of biclusters is proposed whereas, in the second one, the aim is to present different typologies. In both the cases typical algorithms are briefly presented.

1.2.1 The structure of biclusters

The majority of Biclustering algorithms assumes the existence of several biclusters whose number K is defined a priori or calculated according to constraints or quality indexes. However some algorithm aims at finding only one bicluster which is usually the best among many according to some criterion.

Focusing on multi-bicluster methods, it is possible to make other classifications based on repetition of rows/columns, position or overlapping proprieties. In fact rows or columns can belong to one bicluster or to many biclusters at the same time; all the results can be reorganised in order to have all biclusters on the diagonal of the data matrix, in a tree, checkerboard or random structure; biclusters can be without any overlapping situation or with complete or partial overlap.

Therefore it is possible to identify nine different cases:

(a) Single bicluster;

(b) Exclusive row and column biclusters;

(c) Non-Overlapping biclusters with checkerboard structure;

(d) Exclusive-rows biclusters;

(e) Exclusive-columns biclusters;

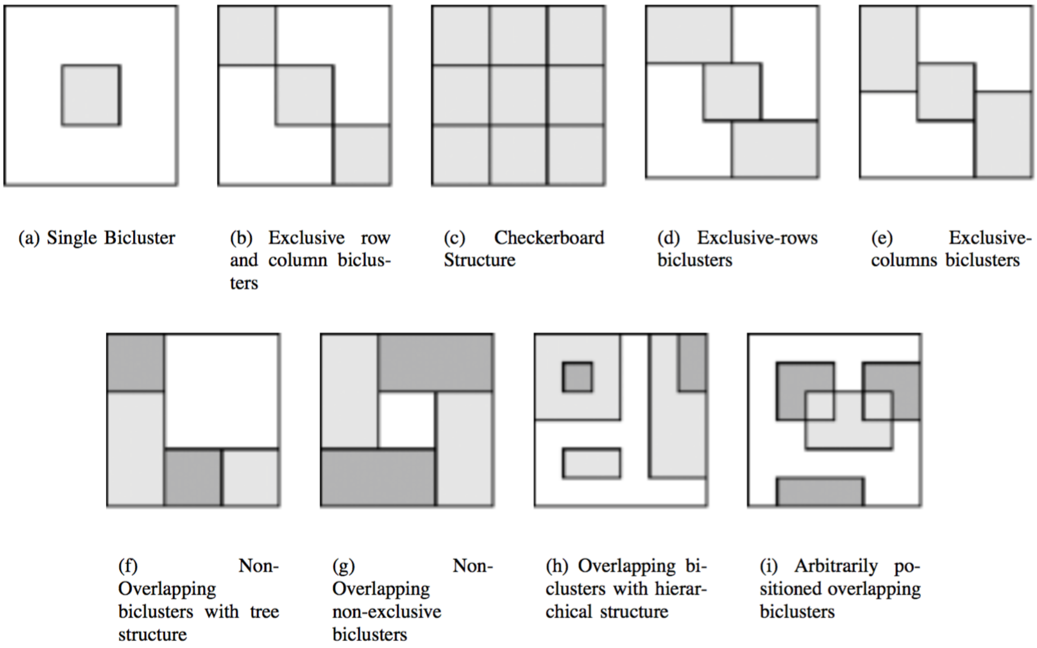
(f) Non-Overlapping biclusters with tree structure;

(g) Non-Overlapping non-exclusive biclusters;

(h) Overlapping biclusters with hierarchical structure;

(i) Arbitrarily positioned overlapping biclusters;

In Figure 1.5 all the possibilities are visually represented.



(a) Single Bicluster

(b) Exclusive row and column biclusters

(c) Checkerboard Structure

(d) Exclusive-rows biclusters

(e) Exclusive-columns biclusters

(f) Non-Overlapping biclusters with tree structure

(g) Non-Overlapping non-exclusive biclusters

(h) Overlapping biclusters with hierarchical structure

(i) Arbitrarily positioned overlapping biclusters

Figure 1.5: All the nine different cases of biclusters. Courtesy of Madeira and Oliveira, 2004

The most natural way to find biclusters is using colours. In fact it is easy to associate to each element a_{ij} a colour depending to its value. This procedure will result in a colour image. At this point one can manipulate the position of all the columns and rows in order to form an image with the possibly largest blocks of similar colours. Every block, made by a subset of columns and rows with similar hues (therefore similar values), is a bicluster. All the other elements which do not belong to any bicluster form a general random-valued background, also called "ragbag" cluster.

In Figure 1.5(a) an example of Single Bicluster is presented. The Biclustering algorithm resulting in this particular structure aims to show the best and largest group of rows and columns. In practical applications, a single Biclustering method is usually made only in order to reduce the dimension of the data.

The biclusters presented in Figure 1.5(b) are exclusive row and column biclusters. In this case every row and every column in the data matrix belongs exclusively to one of the biclusters considered, i.e. every row in the row-block is expressed within, and only within, those columns in condition-block. The most effective way to represent this structure is to reorder the data matrix so that an image with some number K of rectangular blocks on the diagonal is produced. It is important to stress that each block would be nearly uniformly coloured, and the part of the image outside of these diagonal blocks would be of a neutral background colour representing the "miscellaneous" or "ragbag" cluster. The problem presented in subsection 1.1 results in this particular structure, however, as for Single Biclustering, it has long been recognized (see Needham, 1965) that such an ideal reordering will seldom exist in real data.

Facing this fact, it is mandatory to consider that rows and columns may belong to more than one bicluster at the same time: many different bicluster structures are thus available. The first one is showed in Figure 1.5(c) and it is known with the name of Checkboard Structure. It assumes only the existence of K non-overlapping and non-exclusive biclusters where each row and each column belongs to the exact same number of groups. This structure is typical of Kluger et al. (2003) algorithm.

Other Biclustering approaches assume that rows can only belong to one bicluster, while columns, which generally correspond to feature conditions, can belong to several biclusters: it is the case presented in Figure 1.5(d). In this structure exclusive-rows biclusters are assumed, however it is possibile to get exclusive-columns biclusters, as showed in Figure 1.5(e), by transposing the data matrix. This means that the columns of the data matrix can only belong to one bicluster while the rows can belong to one or more biclusters.

One can notice that all the structures proposed in Figure 1.5(b) to Figure 1.5(e) are characterized by the property of exhaustiveness, meaning that every row and and every column belong at least to one bicluster. Also the structures showed in Figure 1.5(f) and in Figure 1.5(g) are exhaustive. The first one is a non-overlapping bicluster with tree structure, as the one resulting from the clustering algorithm proposed by Tibshirani et al. (1999). The second one is called non-overlapping non-exclusive bicluster and was assumed by Wang et al. (2002).

The previous bicluster structures are restrictive in many ways. Some of them assume that, for visualisation purposes, all the identified biclusters should be observed directly on the data matrix and displayed as a contiguous representation after performing a common reordering of their rows and columns. Others assume that the biclusters are exhaustive that is, every row and every column in the data matrix belong to at least one bicluster. However, it is more likely that, in real data, some rows or columns do not belong to any bicluster at all and that biclusters overlap in some places.

Therefore, overlapping structure biclusters are introduced in Figure 1.5(h) and in Figure 1.5(i). They are the most general situations and they allow overlapping and inclusions. In addition, some rows and columns do not belong to any bicluster.

In order to conclude this subsection, it is important to remark that making some variation to exhaustiveness and exclusivity properties allows the reader to create new structures.

1.2.2 The type of biclusters

The second criterion used to evaluate a Biclustering algorithm concerns the identification of the type of biclusters the algorithm is able to find. Four major classes of biclusters can be identified:

(1) Bicluster with constant values;

(2) Bicluster with constant values on rows and columns;

(3) Biclusters with coherent values;

(4) Biclusters with coherent evolutions;

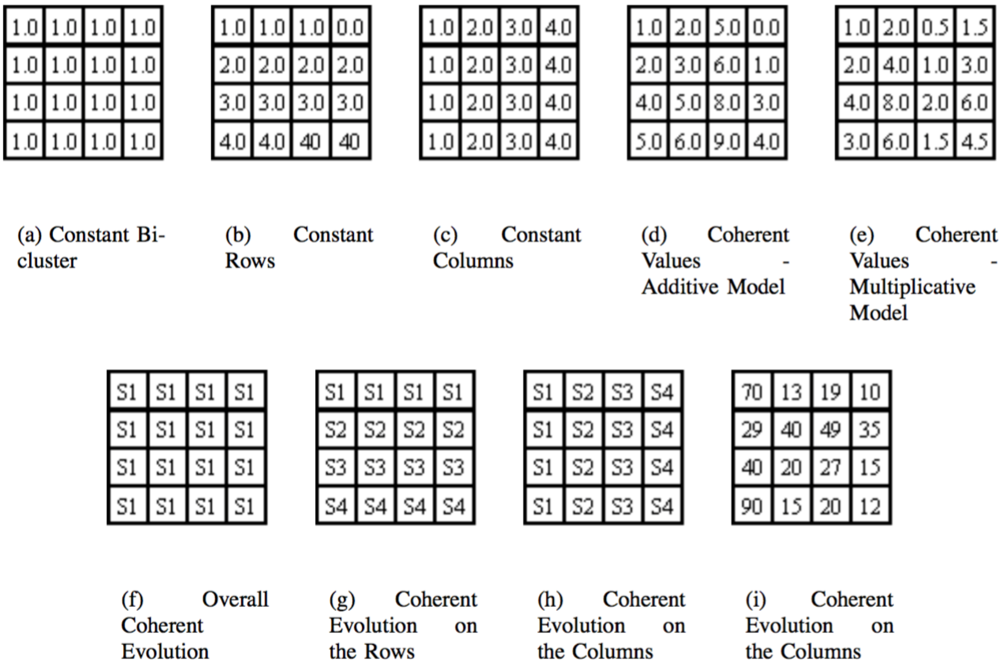
In Figure 1.6 examples of different typologies of biclusters are shown.

According to the specific properties of each problem, one or more of these different types of biclusters are generally considered interesting. The choice of the method is strongly related with the characteristics of biclusters one aims at finding..

When the goal is to discover subsets of rows and subsets of columns (biclusters) with similar values, the natural step is to reorder similar rows and similar columns of the data matrix in order to create group of similarity: this is what, in general, costant bicluster algorithms tend to do. The perfect constant bicluster, for example the one showed in Figure 1.6 (a), is a sub-matrix (I, J), where all values within the bicluster are equal for all $i \in I$ and for all $j \in J$:

$$a_{i,j} = \mu$$

The merit function used to compute and to evaluate constant biclusters is, in general, the variance or some metric based on it. Usually, working with gene expression data



1.0	1.0	1.0	1.0
1.0	1.0	1.0	1.0
1.0	1.0	1.0	1.0
1.0	1.0	1.0	1.0

(a) Constant Bicluster

1.0	1.0	1.0	0.0
2.0	2.0	2.0	2.0
3.0	3.0	3.0	3.0
4.0	4.0	40	40

(b) Constant Rows

1.0	2.0	3.0	4.0
1.0	2.0	3.0	4.0
1.0	2.0	3.0	4.0
1.0	2.0	3.0	4.0

(c) Constant Columns

1.0	2.0	5.0	0.0
2.0	3.0	6.0	1.0
4.0	5.0	8.0	3.0
5.0	6.0	9.0	4.0

(d) Coherent Values - Additive Model

1.0	2.0	0.5	1.5
2.0	4.0	1.0	3.0
4.0	8.0	2.0	6.0
3.0	6.0	1.5	4.5

(e) Coherent Values - Multiplicative Model

S1	S1	S1	S1
S1	S1	S1	S1
S1	S1	S1	S1
S1	S1	S1	S1

(f) Overall Coherent Evolution

S1	S1	S1	S1
S2	S2	S2	S2
S3	S3	S3	S3
S4	S4	S4	S4

(g) Coherent Evolution on the Rows

S1	S2	S3	S4
S1	S2	S3	S4
S1	S2	S3	S4
S1	S2	S3	S4

(h) Coherent Evolution on the Columns

70	13	19	10
29	40	49	35
40	20	27	15
90	15	20	12

(i) Coherent Evolution on the Columns

Figure 1.6: Examples of different types of biclusters. Courtesy of Madeira and Oliveira, 2004

matrix, constant biclusters are masked by noise, therefore noise reduction methods need to be used. However in many real cases, such as the one proposed in this thesis, there is no issue concerning noise.

Figure 1.6 (b) and Figure 1.6 (c) present two new examples of biclusters, one with constant rows and the other one with constant columns. In fact there exists great practical interest in discovering biclusters that exhibit coherent variations on the rows or on the columns of the data matrix. In the case of gene expression data, a bicluster with constant values in the rows identifies a subset of genes with similar expression values across a subset of conditions, allowing the expression levels to differ from gene to gene. The same reasoning can be applied to identify a subset of conditions within which a subset of genes present similar expression values assuming that the expression values may differ from condition to condition. A perfect bicluster with constant rows is a sub-matrix (I, J), where all the values within the bicluster can be obtained using one of the following expressions:

$$a_{i,j} = \mu + \alpha_i$$

$$a_{i,j} = \mu \times \alpha_i$$

where μ is the typical value within the bicluster and α_i is an additive or multiplicative adjustment for row $i \in I$.

Similarly, the same expressions can be applied to explain the perfect bicluster (I, J) with constant columns:

$$a_{i,j} = \mu + \beta_j$$

$$a_{i,j} = \mu \times \beta_j$$

where μ is the typical value within the bicluster and β_j is an additive or multiplicative adjustment for column $j \in J$.

This kind of biclusters can not be evaluated just by variance of its values and, in general, the typical approach used to identify non-constant biclusters is to normalize the rows or the columns of the data matrix using the row mean and the column mean, respectively. The normalization transforms Figure 1.6(b) and Figure 1.6(c) in 1.6(a): there is thus a relationship between constant bicluster and constant rows/columns bicluster. This bond is therefore the core of many algorithms.

Other methods, instead, aim at finding δ-valid ks-pattern, defined as a subset of rows, I, with size k, and a subset of columns, J, with size s, such that the maximum and minimum value of each row in the chosen columns differ less than δ. Precisely, for each row $i \in I$:

$$\max_{j \in J}(a_{i,j}) - \min_{j \in J}(a_{i,j}) < \delta$$

It is straightforward the application to columns instead of rows, as also proposed by Califano et al. (2000).

An overall improvement over the methods considered in the previous section, which presented biclusters with constant values either on rows or columns, is to consider biclusters with coherent values on both rows and columns. In Figure 1.6(d) and in Figure 1.6(e) some examples of this type of biclusters are showed. This class of biclusters are harder to be found than the previous ones due to the fact that the values within the bicluster are not given by additive or multiplicative models that consider an adjustment for either the rows or the columns. However the Biclustering algorithms that look for biclusters with coherent values can be viewed as based on a more complex additive model. In this case, a perfect bicluster (I, J) with coherent values is defined as a subset of rows and a subset of columns, whose values are given by the following expression:

$$a_{ij} = \mu + \alpha_i + \beta_j$$

It is therefore evident that, differently from the types of bicluster previously explained, there are two adjustments and not only one: α_i for the row $i \in I$ and β_j for the column $j \in J$. As usual μ is the typical value within the bicluster. It is thus possible to notice how the cases showed in Figure 1.6(b) and Figure 1.6(c) can be considered special cases of the general multi-parameter additive model just presented. Figure 1.6(d) is an example.

However, as for the constant rows and constant columns biclusters, the approaches that look for biclusters with coherent values can also be viewed as based on a multi-parameter multiplicative model. Precisely:

$$a_{ij} = \mu' \times \alpha'_i \times \beta'_j$$

where μ' is the typical value within the bicluster, α'_i is the adjustment parameter for the row $i \in I$ and β_j is the adjustment parameter for the column $j \in J$. The approach based on additive model and the one based on multiplicative model are the same when $\mu = \log \mu'$, $\alpha'_i = \alpha_i$ and $\beta'_j = \beta_j$. Even in this case the bicluster in Figure 1.6 (b) can be considered special cases of this multiplicative model when $\alpha'_i = 0$, while bicluster in Figure 1.6 (c) can also be considered special cases but in the case of $\beta'_i = 0$.

An example is given in Figure 1.6 (e).

Many Biclustering algorithms aim at discovering biclusters with coherent values assuming either additive or multiplicative models. The first attempt was made by Cheng and Church (2000) and their mean squared residue approach, analysed in detail in the next chapter. In particular, they aim at finding large and maximal biclusters with scores below a certain threshold δ.

A generalized definition of the newly introduced δ-bicluster was given by Yang et al. (2002) and their FLOC (FLexible Overlapped biClustering) algorithm, which modified the Cheng and Church approach with an occupancy threshold θ.

Kluger et al. (2003)also addressed the problem of identifying biclusters with coherent values and looked for checkerboard structures (Figure 1.5) in the data matrix by integrating Biclustering of rows and columns with normalization of the data matrix. They assumed that after a particular normalization, which was designed to accentuate biclusters if they exist, the contribution of a bicluster is given by a multiplicative model.

Many approaches are based on the iteration of one-way clustering in order to produce coherent biclusters. Examples include the Interrelated Two-Way Clustering (ITWC) by Tang et al. (2001) and the Coupled Two-Way Clustering (CTWC) by Getz et al. (2000).

The previous Biclustering approaches are based either on additive or multiplicative models, which evaluate separately the contribution of each bicluster without taking into consideration the interactions between biclusters. Lazzeroni and Owen (2002) introduced plaid models where the value of an element in the data matrix is viewed as a sum of terms called layers. Therefore the data matrix, according to the colour image analogy, is described as a linear function of layers, which are variables, corresponding to its biclusters. The plaid model described can be seen as a generalization of the additive model and, for this reason, it is also called the general additive model.

Other Biclustering algorithms address to a wider problem: find coherent evolutions across the rows and/or columns of the data matrix regardless of their exact values. We may be interested, for instance, in looking for up-regulated or down-regulated subsets of rows across subsets of columns without taking into account their actual values in the data matrix. Changing our fields from values to evolutions, it is possible to define the same structures previously presented and talk about overall coherent evolution in Figure 1.6(f), coherent evolution on the rows in Figure 1.6(g), coherent evolution on the columns in Figure 1.6(h) and Figure 1.6(i). The most famous algorithms aim at finding coherent evolution biclusters are presented.

Murali and Kasif (2003) proposed a method to find conserved gene expression motifs (xMOTIFs). They defined an xMOTIF as a subset of genes (rows) that is simultaneously conserved across a subset of conditions (columns). The conservation of the expression level of a gene happens across a subset of conditions if the gene is in the same state in each of the conditions in this subset. Therefore the analysis is not based on the values of the gene expression level but on a label, called state, obtained by discretization of the range. For instance, with two labels, only the down-regulated and the up-regulated cases are considered. Even if more than one xMOTIF can be found, Murali and Kasif aim at finding the largest one: the bicluster that contains the maximum number of conserved rows. The merit function used to evaluate the result is thus the size of the subset of rows that belong to the found bicluster.

The second algorithm here presented is known as SAMBA (Statistical-Algorithmic Method for Bicluster Analysis). It was firstly introduced by Tanay et al. (2002) and it looks for biclusters defined as a subset of rows that jointly respond across a subset of columns. In the case of gene expression data, a gene (row) is considered to respond to a certain condition if its expression level changes significantly at that condition (column) with respect to its normal level. After that data matrix is modelled as a bipartite graph with one edge for each significant change, SAMBA is applied in order to discover biclusters (sub-graphs) with an overall coherent evolution. Therefore, in the case of gene expression data, SAMBA does not try to find any kind of coherence on the values, a_{ij}, in the bicluster, assuming that regardless of its true value, a_{ij} is either 0 or 1, where 1 is up-regulation and 0 is down-regulation

Chapter 2

The Cheng and Church's algorithm

sible

Je choisis un bloc de marbre et de couper tout ce que je n'ai pas besoin.

François-Auguste Rodin

In this chapter the famous Biclustering algorithm presented by Yizong Cheng and George M. Church in 2000 is explained in details. In the first section the historical and applicative setting that characterized the baseline of this method is described. In this way the reader will understand why, still today, the paper signed by Cheng and Church is considered the most important literature in the gene expression biclustering field.

The second section, instead, points out the notation and illustrates every minutia which composes the algorithm. All the sub-algorithms that characterise the method are shown and explained. The generalities are presented according to the classification introduced in the previous chapter.

2.1 The importance of being Cheng and Church

In 2000, thanks to the high popularity of gene expression data generated by DNA chips and other microarray techniques, the problem of grouping genes according to their level under multiple conditions or, inversely, to group conditions based on the expression of a number of genes becomes relevant for biostatistics. In fact, a simultaneous Clustering of both genes and conditions might make possible to find co-regulation patterns in yeast and humans.

To make it possible, the usual practice was to use some agglomerative or divisive Clustering algorithm that partions the genes or conditions into mutually exclusive groups. As said in the previous chapter, every similarity group found by these techniques, principally based on functions that may include Euclidean distance between vectors, obscures some other groups. Using the idiomatic English proverb, “You can’t have your cake and eat it”.

Therefore, in their article entitled “Biclustering of Expression Data”, Cheng and Church introduced a new kind of grouping known with the name of “Biclustering”. It results in a subset of genes and a subset of conditions with a high similarity score, where a similarity score is not intended as a function of either genes or conditions but as a measure of coherency inside the obtained bicluster. Let us remark that this definition can be easily generalised by simply consider the genes and the conditions respectively as impersonal rows and columns.

In the case here considered, thus, the perfect bicluster is represented by a submatrix as large as possible, totally filled by constant values. Using the classification introduced in Section 1.2.2, Cheng and Church’s Biclustering algorithm aims at finding constant or coherent type biclusters. Many other methods for partitioning data into sets with approximately constant values had been proposed before Cheng and Church, for example, Morgan and Sonquist (1963) and Hartigan (1972). However the term “Biclustering” was firstly introduced by Mirkin in 1996 to describe “simultaneous clustering of both row and column sets in a data matrix”. Many other names like “direct Clustering” (Hartigan, 1972) and “box clustering” (Mirkin, 1996) were used to define similar ideas but they never became fashionable due to the lack of applications. In fact gene expression data were not still available nor famous. Furthermore, the Clustering methods above quoted principally result in a hierarchy of clusters rather than biclusters and, therefore, there were not any possibility of overlapping. This lack needed to be solved in order to analyse similar genetical patterns. In fact single genes may participate in multiple pathways that may or may not be co-activate under all conditions.

For the reasons just explained, Cheng and Church immediately become the fathers of a new technique called Biclustering and their paper gives birth to many other similar algorithms, the majority of which was presented in the previous chapter. It is not a case that the work signed by Cheng and Church is still considered a major goal in gene expression data analysis.

2.2 The Cheng and Church’s algorithm

The algorithm proposed by Cheng and Church is an efficient node-deletion method introduced to find submatrices in expression data that have low mean squared residue scores. In the following sections notations, generalities and modes of operation of the method are explained. Precisely the mean squared residue score is formally expressed; the bipartite graph point of view is presented; the algorithm is analysed following the theoretical basis explained in Section 1.2; a set of efficient algorithms that find these interesting submatrices are shown. Eventually, a masking method in order to find more than one bicluster and to iterate the process is briefly presented.

2.2.1 Notations and generalities

As already explained in Section 1.1.4, let X be the set of rows and Y the set of columns. The element a_{ij} of the matrix $A = (X, Y)$ is the value corresponding to the i-th row and j-th column. Let $I \subset X$ and $J \subset Y$ be subsets of rows and columns, therefore (I, J) specifies a submatrix called A_{IJ}. In the genetical application field, the set of rows is a set of genes while the set of columns is a set of conditions.

In order to express the mean squared residue score, the row and column means are defined in the following way:

$$a_{iJ} = \frac{1}{\mid J \mid} \sum_{j \in J} a_{ij}$$

$$a_{Ij} = \frac{1}{\mid I \mid} \sum_{i \in I} a_{ij}$$

Instead, the mean in the submatrix (I, J) is:

$$a_{IJ} = \frac{1}{\mid I \mid\mid J \mid} \sum_{j \in J, i \in I} a_{ij} = \frac{1}{\mid I \mid} \sum_{i \in I} a_{iJ} = \frac{1}{\mid J \mid} \sum_{j \in J} a_{Ij}$$

Therefore the following expression defines the mean squared residue score:

$$H(I, J) = \frac{1}{\mid I \mid\mid J \mid} \sum_{i \in I, j \in J} (a_{ij} - a_{iJ} - a_{Ij} + a_{IJ})^2$$

Cheng and Church explain the measure just defined in the following way: "it is the measure of coherence used to validate biclusters. It is the variance of the set of all elements in the bicluster, plus the mean row variance and the mean column variance". However it is possible to find a different model-based interpretation thanks to the assumption of underlying additive model to describe biclusters. In particular, taking inspiration from the first attempts of Biclustering by Hartigan, 1972, the additive model used in Cheng and Church's algorithm is part of a two-factor analysis of variance model. The general goal in ANOVA studies is to estimate and test the effects of the factor levels on the mean of the response variable. In the case here proposed the sources of variation come from two factors: rows and columns or, in the typical case, genes and conditions. Using the ANOVA point of view just introduced, the mean squared residue score corresponds to the variance arising from the error term.

The algorithm proposed by Cheng and Church aims at finding large and maximal biclusters with the score measure just defined below a certain threshold called δ. Precisely a submatrix A_{IJ} is called a δ-bicluster if $H(I, J) \leqslant \delta$ for some $\delta \geqslant 0$. Therefore, as said in Section 1.2.2, the result of the algorithm is a δ-bicluster. The problem of finding a maximum submatrix with a score lower than a threshold includes the problem of finding a maximum biclique (complete bipartite graph) in a bipartite graph as special case. As previously said in Section 1.1.5, everything can be analysed as a graph optimization problem. Thus, if a maximum biclique is one that maximizes its number of vertices (maximizing $\mid I \mid$ and $\mid J \mid$), then the problem can be solved using polynomial time max-flow algorithms.

At this point, one can easily imagine that the best situation is given by the lowest score $H(I, J) = 0$ meaning, thanks to the use of the additive model, a variance of the error term of the corresponding additive model of 0. It is thus evident that a particular case of perfect bicluster is a constant bicluster. However, sometimes, the optimal result is quite trivial and a masking process needs to be done in order to identify other interesting biclusters.

2.2.2 The algorithm

The Cheng and Church's algorithm can be considered as a three steps procedure based on bipartite graph optimization. Initially the nodes of the graph, resulting from the data matrix, are deleted in order to minimize the measure. Then the result of the deletion is modified by adding nodes which do not impact on the gained score. Therefore the maximal bicluster below a chosen threshold is identified. At this point the algorithm is iterated without considering the results already found. Algorithm 1 expresses in a formal way what already said.

Algorithm 1 Cheng and Church's Biclustering algorithm

```
Data: $A$ a matrix of real numbers with possible missing elements;
      $\alpha \geqslant 1$, a parameter for multiple node deletion;
      $\delta \geqslant 0$, the maximum acceptable mean square residue score;
      $n$ the maximum number of biclusters to be found [not necessary].
Result: a maximum of $n$ biclusters in $A$
Apply Algorithm 5 to perform node deletion
Apply Algorithm 6 to perform node addition to get the bicluster
Report the bicluster and mask the data
```

In the next subsections all three steps of the algorithm will be analysed and explained in details.

2.2.3 The node deletion

Every element a_{ij} in every matrix A is a trivial case of submatrix having the perfect score of $H(I, J) = 0$. However, biclusters one looks for should have a maximum size both in terms of the number of rows, $\mid I \mid$, and in terms of the number of conditions, $\mid J \mid$. This means that considering every element as an exhaustive solution is not a case. Therefore the question is how to select a maximal submatrix with a low H score.

Starting with a large matrix, one can decide to use a greedy method: remove rows or columns in order to achieve the largest decrease of the score. This requires the computation of the scores of all the submatrices that may be the consequences of any row or column removal, before each choice of removal can be made. Even if it seems to work as expressed for Bruce Wayne's party example, it is rather burdensome in terms of computational time. In fact it requires time in $O((n+m)nm)$, where n and m are the number of row and column of the matrix, just to find one bicluster. The method just explained is described in Algorithm 2.

Algorithm 2 Brute-Force node deletion

```
Data: $A$ a matrix of real numbers;
      $\delta \geqslant 0$, the maximum acceptable mean square residue score.
Result: $A_{IJ}$, a $\delta$-bicluster that is submatrix of $A$ with row set $I$ and column set $J$, with
        a score no larger than $\delta$
Initialize $I$ and $J$ to the whole row and column sets in the data: $A_{IJ} = A$
Compute the score $H$ for every possible row or column deletion and choose the action
    that decreases $H$ the most
Return $A_{IJ}$
```

It is evident that for a quick analysis of large data matrix Algorithm 2 will not be efficient enough, therefore it is necessary to utilise better methods able to perform node deletion for finding biclusters. At this point, Cheng and Church introduce two different alternatives whose combination provides a very efficient algorithm. The correctness and efficiency of these two procedures are based on a number of lemmas, in which rows (or columns) are treated as points in a space where a distance is defined. To get an idea, the reader is invited to read the original paper (Cheng and Church, 2000) or a general book about removal algorithms.

The first method is used to perform the removal of a single node and it deletes one node a time. After having initialized I and J to the whole row and column sets in the data, i.e. $A_{IJ} = A$, the row and column means a_{iJ} and a_{Ij} are computed for each $i \in I$ and $j \in J$. Then a_{IJ} and H are calculated and if $H(I, J) \leqslant \delta$ the maximal bicluster with respect to the threshold is already found. If this is not the case, it is necessary to perform the node deletion. The row $i \in I$ and the column $j \in J$ which largely contribute to the score H are identified and compared: the node (row or column) with the biggest contribution is removed. The method just explained is described in Algorithm 3.

Algorithm 3 Single node deletion

```
Data: $A$ a matrix of real numbers;
      $\delta \geqslant 0$, the maximum acceptable mean square residue score.
Result: $A_{IJ}$, a $\delta$-bicluster that is submatrix of $A$ with row set $I$ and column set $J$, with
        a score no larger than $\delta$
Initialize $I$ and $J$ to the whole row and column sets in the data: $A_{IJ} = A$. Compute $a_{iJ}$
for all $i \in I$, $a_{Ij}$ for all $j \in J$, $a_{IJ}$ and $H(I,J)$
while $H(I,J) \geq \delta$ do
    Compute $a_{iJ}$ for all $i \in I$, $a_{Ij}$ for all $j \in J$, $a_{IJ}$ and $H(I,J)$
    Find the row $i \in I$ with the largest
        $d(i) = \frac{1}{\mid J \mid} \sum_{j \in J} (a_{ij} - a_{iJ} - a_{Ij} + a_{IJ})^2$
    Find the column $j \in J$ with the largest
        $d(j) = \frac{1}{\mid I \mid} \sum_{i \in I} (a_{ij} - a_{iJ} - a_{Ij} + a_{IJ})^2$
    if $d_i > d_j$ then
        delete row $i$
    else
        delete column $j$
    end
end
Return $A_{IJ}$
```

It is easy to understand that thanks to the fact that there are only a finite number of rows and columns to remove, the procedure ends in maximum $n + m$ iterates, where n and m are, as usual, the number of rows and columns respectively. One can also notice that the algorithm needs to make a choice based on the contribution to the mean square residue score H and sometimes it could not be seem possible. In fact, it may happen that all $d(i)$ and $d(j)$ are equal to $H(I,J)$ for $i \in I$ and $j \in J$. In this case, the removal of one of them may still decrease the score, unless the score is already 0.

In term of efficiency Step 1 in each iterate requires time in $O(mn)$ and a complete recalculation of all d values in Step 2 is also an $O(mn)$ effort. The selection of the best row and column candidate that has to be removed takes $O(\log n + \log m)$ time. When the matrix is bi-level, i.e. filled only by 1 and 0, the update of variables in the first line of Algorithm 3 after the removal of a row takes only $O(m)$ time while after the deletion of a column takes only $O(n)$. Therefore, it is possible to have an overall running time in $O(mn)$ and it makes the algorithm very efficient for bi-level matrices.

Algorithm 4 Multiple node deletion

```
Data: $A$ a matrix of real numbers;
      $\alpha \geqslant 1$, a parameter for multiple node deletion;
      $\delta \geqslant 0$, the maximum acceptable mean square residue score.
Result: $A_{IJ}$, a $\delta$-bicluster that is submatrix of $A$ with row set $I$ and column set $J$, with
        a score no larger than $\delta$
Initialize $I$ and $J$ to the whole row and column sets in the data: $A_{IJ} = A$. Compute $a_{iJ}$
for all $i \in I$, $a_{Ij}$ for all $j \in J$, $a_{IJ}$ and $H(I,J)$
while $H(I,J) \geq \delta$ do
    Compute $a_{iJ}$ for all $i \in I$, $a_{Ij}$ for all $j \in J$, $a_{IJ}$ and $H(I,J)$
    Remove the rows $i \in I$ with
        $$\frac{1}{\mid J \mid}\sum_{j\in J}(a_{ij} - a_{iJ} - a_{Ij} + a_{IJ})^2 > \alpha H(I,J)$$
    Recompute $a_{Ij}$, $a_{IJ}$ and $H(I,J)$
    Remove the column $j \in J$ with
        $$\frac{1}{\mid I \mid}\sum_{i\in I}(a_{ij} - a_{iJ} - a_{Ij} + a_{IJ})^2 > \alpha H(I,J)$$
    if there is no deletions then
        Apply Algorithm 3
    end
end
Return $A_{IJ}$
```

Due to the fact that the single node deletion algorithm is not convenient for the application to non-bi-level matrix, it is necessary to introduce a new procedure that allows multi-node removal. Therefore, the new algorithm proposed by Cheng and Church is able to delete more than one node (row or column) a time. However, to regulate this new skill, the procedure presents the new parameter α as a threshold. In general, the method is really similar to the one previously presented with the main difference that there are no comparison: every node beyond a value identified by the threshold is removed. This generates an extremely fast procedure that may return too much shrunk matrices. Algorithm 4 presents what just said.

It is interesting to notice that it may happen that in the iterates of Algorithm 4 there are no reasons to perform a multiple deletion. This could be a problem because there would not be any way to escape from the while loop. Therefore, if it is the case, the procedure automatically calls for the help of the single node deletion algorithm. In fact Algorithm 3 is always able to delete a node making $H(I, J) < \delta$ possible.

At this point, the reader is ready to understand the node deletion method proposed by Cheng and Church. Their procedure smartly combines single and multi removal in order to shrink the data matrix. Precisely multiple node deletion is performed and then, in order to refine the removal, single node deletion follows.

Algorithm 5 Cheng and Church's node deletion

```
Data: $A$ a matrix of real numbers;
      $\alpha \geqslant 1$, a parameter for multiple node deletion;
      $\delta \geqslant 0$, the maximum acceptable mean square residue score.
Result: $A_{IJ}$, a $\delta$-bicluster that is submatrix of $A$ with row set $I$ and column set $J$, with
        a score no larger than $\delta$
Apply Algorithm 4 on $A$, $\delta$ and $\alpha$. If the row (column) are less than 100, do not perform the procedure on rows (columns). The resulting matrix is called $B$
Apply Algorithm 3 on $B$ and $\delta$. The resulting matrix is called $C$
Return $A_{IJ} = C$
```

Let us remark that it could be possible that the resulting bicluster would be too much shrunk. The procedure, in fact, may miss some large δ-biclusters. To solve this problem one may decide to use an adaptive α based on the score and size during the iteration. However, Cheng and Church suggest to use a node addition algorithm.

2.2.4 The node addition

After node deletion, the resulting bicluster may not be maximal because too shrunk. Therefore, some rows and columns may be added without increasing the mean squared residue score H. To prove it, Cheng and Church present Lemma 3 and Theorem 3 that guarantee how adding rows or columns does not increase H.

However, the resulting bicluster may not be maximal because every decision about adding a node it is made according to the current score H and not δ. It means that each iterate only adds rows and columns according to the current score, not δ. Therefore, the method usually returns submatrices whose H is a lot smaller than δ and some addable nodes are discriminated by this fact.

Algorithm 6 Node addition

```
Data: $A$ a matrix of real numbers;
      $I$, and $J$ representing a $\delta$-bicluster ;
Result: $I'$ and $J'$ such that $I' \subset I$ and $J' \subset J$ with the property that $H(I', J') \leqslant H(I, J)$
1  Compute $a_{iJ}$ for all $i \in I$, $a_{Ij}$ for all $j \in J$, $a_{IJ}$ and $H(I, J)$
2  while the first iteration needs to be done or at least a row $i$ or a column $j$ is added do
3      Add the columns $j \notin J$ with
           $$\frac{1}{\mid I \mid} \sum_{i \in I} (a_{ij} - a_{iJ} - a_{Ij} + a_{IJ})^2 \leqslant H(I, J)$$
4      Recompute $a_{iJ}$ for all $i \in I$, $a_{IJ}$ and $H(I, J)$
5      Add the rows $i \notin I$ with
           $$\frac{1}{\mid J \mid} \sum_{j \in J} (a_{ij} - a_{iJ} - a_{Ij} + a_{IJ})^2 \leqslant H(I, J)$$
6      For each row $i \notin I$ if $\frac{1}{|J|} \sum_{j \in J} (-a_{ij} + a_{iJ} - a_{Ij} + a_{IJ})^2 \leqslant H(I, J)$ then
7          add the inverse of $i$
8      end
9  end
10 Return $A_{I'J'}$
```

Talking about the procedure of node addition, shown in Algorithm 6, it is possible to notice how it seems really similar to the the multi node deletion method. However it presents some news, precisely in Step 6 where the iteration adds inverted rows into the bicluster. These nodes form “mirror image” of the rest of the rows in the bicluster and can be interpreted as corregulated but receiving the opposite regulation. At this point it is important to remark that inverted rows cannot be added at the beginning, because that will make all $a_{Ij} = 0$ and also $a_{IJ} = 0$. Let us remark, however, that Step 6, according to Cheng and Church's paper, is not mandatory: many implementations do not present this passage.

Concerning the efficiency of Algorithm 6, it can be easily compared to the multi node deletion method.

2.2.5 Masking procedure and Conclusion

After node deletion and node addition a bicluster is found and reported. Because the algorithms are all deterministic, repeated run will not discover new submatrices, unless the ones already discovered are masked. The masking procedure consists in replacing all the elements in the matrix representing the results already found with random values. This makes quite unlikely that elements covered by existing biclusters would contribute to any future pattern discovery. However, the mask is not used during node addition and it allows a real overlapping of biclusters (and not only an overlap by projection along one dimension).

Chapter 3

Urbanscope

Ho attraversato tutta la città. Poi ho salita un'erta, popolosa in principio, in là deserta, chiusa da un muricciolo: un cantuccio in cui solo siedo; e mi pare che dove esso termina termini la città.

Umberto Saba

In this chapter the reader will get acquainted with the Urbanscope project and he will discover all the elements that compose it: the aim of this city macroscope, the concept of lens and the subdivision in "Explore" and "Analyse" that offers to the user different approaches to the data.

Precisely, the first section will present all the generalities about Urbanscope, showing how interesting is to analyse the evolution of a city by social digital trace.
In the second section the structure of the project is presented through the major characteristics of the Urbanscope's website. Finally, after a brief description of all the different points of view that the macroscope permits with its lenses, the attention is drawn on the "City Magnets" lens, whose "Analyse" section has been developed in this thesis.

3.1 Urbanscope as a macroscope

Just imagine one wants to have a new representation of the city of Milan. One can decide to walk all around its streets in order to understand the dynamics that characterize it. Otherwise, one can behave as Umberto Saba describes in his poem entitled "Trieste": traverse the entire town, climb a steep slope and observe the metropolitan life from above.

There exists many ways to have a privileged vista of the city of Milan and, nowadays, thanks to the Urbanscope project the number of possibilities is enlarged. In fact the website, officially presented on 20th July of 2015 by Politecnico di Milano, gives to any citizen or, in general, any curious person a wide range of instruments to understand how Milan is evolving. These tools, in effect, can figure out another urban dimension that human sight is unable to perceive: all the information contained in the digital traces that everyday we leave behind us.

Figure 3.1: A snapshot taken from the welcoming video in the header of the homepage of www.urbanscope.polimi.it

Nowadays, “cities are not mere physical and organizational structures: they are informational landscapes where places are shaped more by the streams of contents and less by the traditional physical evidences”(see Urbanscope introduction) . As the Urbanscope homepage reminds, it is anachronistic to imagine every urban reality as a simple juxtaposition of buildings and people. In fact, it is important to consider also the interaction between users and environment and this interaction can be either materialistic or digital. “Two layers coexist: a thick and dynamic layer of digital traces that like an informative membrane grows everyday on top of the stable material layer, composed by the territory, the buildings, and the infrastructures”. According to these differences, the city can apparently stay the same in the solid marble façades of its buildings and, at the same time, change convulsively in the digital world. Therefore, considering that the number of people who started speaking "social" is increasing, the observation, the analysis, and the representation of the combination of the two layers previously introduced provide a valuable insights on how the city is perceived and lived.

The Urbanscope’s research team, based at Politecnico di Milano and composed by researchers with competences in Computing Engineering, Communication and Information Design, Management Engineering, and Mathematical Engineering, uses its knowledge in order to gain useful informations and produce tools. Precisely, compelling views on urban systems are created to foster comprehension and decision making. In order to make it possible, data coming from social media within the urban context of Milan are collected, analysed and then represented in a particular "lens", a tool designed to support the recognition of specific patterns and to enable new perspectives. Today, in fact, big data deriving from social media has two main advantages: the easiness to have measurement at individual level and the high cheapness if compared with traditional surveys. Moreover, due to the low cost and the high speed of data download, it is possible to collect and visualise data in real time.

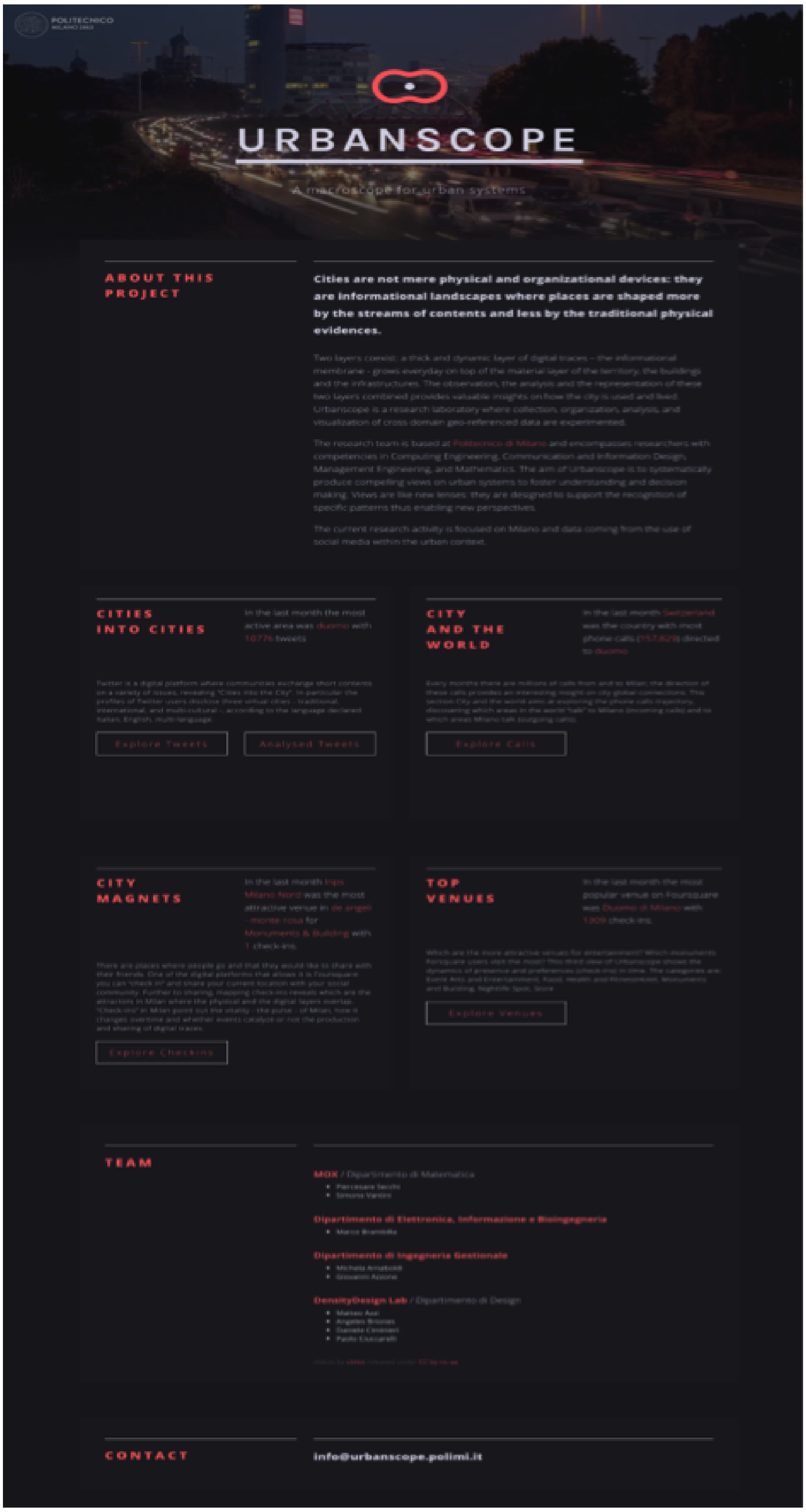
Therefore, Urbanscope can be also seen as a collection of privileged instantaneous point of views, a pair of powerful field glasses able to detect the digital traces that cover the territory and to make them visible. It is a fundamental macroscope to discover the city.

3.2 The structure of Urbanscope

The homepage of the Urbanscope online platform is composed by the following elements:

- Header - including the title and three time lapse videos of Milan;
- Description of the project - presenting few lines introducing the user to the finalities of the project;
- Modules - investigating a different aspect of the evolution of Milan and, thus, they can be considered like lenses of the macroscope;
- Team - listing all the researchers involved in the project;
- Contact - showing the e-mail address referring to the project;

Generally, the homepage works as the Urbanscope's entrypoint and it has the hard mission to introduce the user to the mascroscope. Therefore, it is important that every section seems as clear and as easy to recognise as possible. However the core of the project is represented by the different city visions that it allows: the modules. To make them stand out, they have been inserted in the perfect center of the dashboard.



POLITECNICO MILANO 1863

URBANSCOPE

A macroscope for urban systems

ABOUT THIS PROJECT

Cities are not mere physical and organizational devices: they are informational landscapes where places are shaped more by the streams of contents and less by the traditional physical evidences.

Two layers coexist: a thick and dynamic layer of digital traces – the informational membrane - grows everyday on top of the material layer of the territory, the buildings and the infrastructures. The observation, the analysis and the representation of these two layers combined provides valuable insights on how the city is used and lived. Urbanscope is a research laboratory where collection, organization, analysis, and visualization of cross domain geo-referenced data are experimented.

The research team is based at Politecnico di Milano and encompasses researchers with competencies in Computing Engineering, Communication and Information Design, Management Engineering, and Mathematics. The aim of Urbanscope is to systematically produce compelling views on urban systems to foster understanding and decision making. Views are like new lenses: they are designed to support the recognition of specific patterns thus enabling new perspectives.

The current research activity is focused on Milano and data coming from the use of social media within the urban context.

CITIES INTO CITIES

In the last month the most active area was duomo with 10776 tweets

Twitter is a digital platform where communities exchange short contents on a variety of issues, revealing "Cities into the City". In particular the profiles of Twitter users disclose three virtual cities – traditional, international, and multi-cultural –, according to the language declared: Italian, English, multi-language.

Explore Tweets

Analysed Tweets

CITY AND THE WORLD

In the last month Switzerland was the country with most phone calls (157,829) directed to duomo

Every months there are millions of calls from and to Milan; the direction of these calls provides an interesting insight on city global connections. This section City and the world aims at exploring the phone calls trajectory, discovering which areas in the world "talk" to Milano (incoming calls) and to which areas Milano talk (outgoing calls).

Explore Calls

CITY MAGNETS

In the last month Inps Milano Nord was the most attractive venue in de angeli - monte rosa for Monuments & Building with 1 check-ins.

There are places where people go and that they would like to share with their friends. One of the digital platforms that allows it is Foursquare: you can "check in" and share your current location with your social community. Further to sharing, mapping check-ins reveals which are the attractions in Milan where the physical and the digital layers overlap. "Check-ins" in Milan point out the vitality - the pulse - of Milan, how it changes overtime and whether events catalyze or not the production and sharing of digital traces.

Explore Checkins

TOP VENUES

In the last month the most popular venue on Foursquare was Duomo di Milano with 1309 check-ins.

Which are the more attractive venues for entertainment? Which monuments Foursquare users visit the most? This third view of Urbanscope shows the dynamics of presence and preferences (check-ins) in time. The categories are: Event Arts and Entertainment, Food, Health and Environment, Monuments and Building, Nightlife Spot, Store.

Explore Venues

TEAM

MOX / Dipartimento di Matematica

- Piercesare Secchi
- Simone Vantini

Dipartimento di Elettronica, Informazione e Bioingegneria

- Marco Brambilla

Dipartimento di Ingegneria Gestionale

- Michela Arnaboldi
- Giovanni Azzone

DensityDesign Lab / Dipartimento di Design

- Matteo Azzi
- Angeles Briones
- Daniele Ciminieri
- Paolo Ciuccarelli

[illegible]

CONTACT

info@urbanscope.polimi.it

Figure 3.2: The Urbanscope's homepage

3.2.1 The Modules

The modules analyse different phenomenal aspects of Milan: by changing the coloured lens of the Urbanscope's macroscope, they allow the detection of specific patterns. Every module presents a brief static description and a dynamic paragraph. The latter one gives already a preview based on the data and on the analysis in order to guide the user through the experience. At the moment few lines giving qualitative information are presented as a preview but soon they will be joined also by some visual details. Today, four different modules are available:

- Cities into cities - based on the digital platform of Twitter;
- City and the world - whose analysis has been carried out thanks to mobile phone data;
- City magnets - main issue of the study proposed in this thesis; based on checkins performed with the Foursquare mobile application;
- Top venues - that tries to explain the most attractive venues in the city using Foursquare;

Every lens presents two different ways of approach: the first one is defined by the "Explore" button, the other one by the "Analyse" button. At the moment, only the "Cities into cities" section is equipped with both the approaches. In fact, one of the task of this thesis is to provide for the creation of an "Analyse" page for the "City magnets" module.

The "Explore" pages usually give to the user the possibility to visualise the data and play with them. They allow to have a first description of the evolution of the city.

The "Analyse" section, instead, tries to deepen the understanding of the phenomena by presenting more complex analyses and visualisations. The views here presented are characterised by mathematical and statistical methods which are explicated in a user-friendly way.

Both the approaches, however, are usually based on the use of interactive maps. In fact maps, supported by Openstreetmap, are fundamental tools to represent data that are highly geo-correlated. In many cases maps are divided in NILs, "Nuclei di Identità Locale", which are the official divisions into neighbourhood proposed by the municipality of Milan. They are 88 and they cover all Milan's urban area. Figure 3.3 shows the city divided in NILs.

In order to make everything clearer to the reader, the next sections will review the different modules now available.

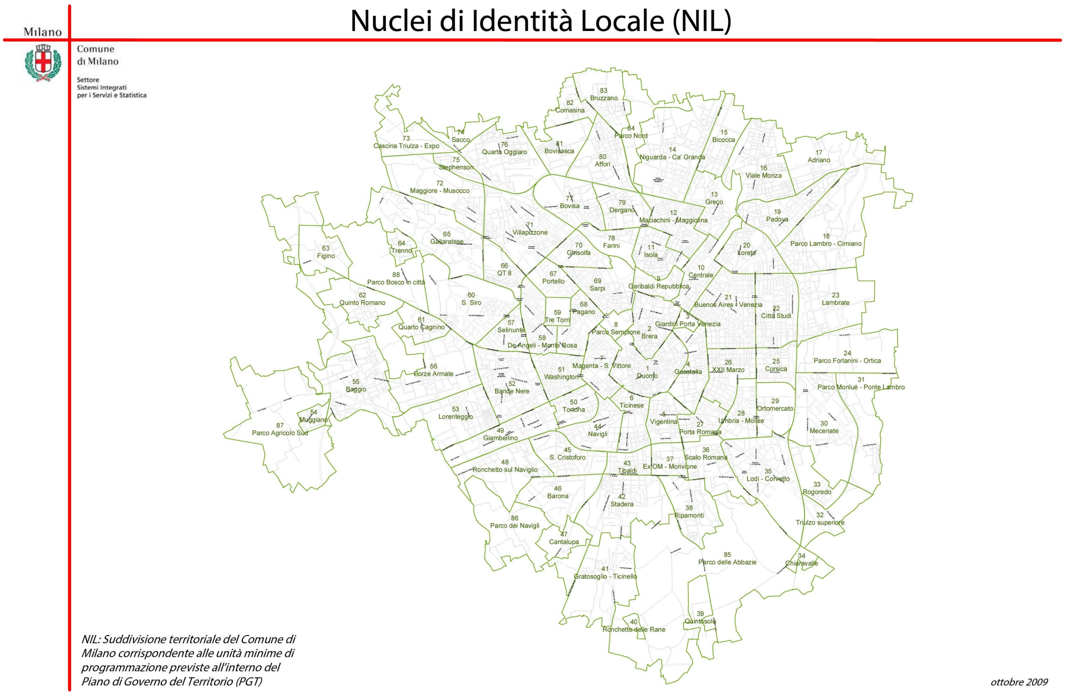


Figure 3.3: An image taken from the official site of the municipality of Milan showing the division of the city in 88 *NILs*

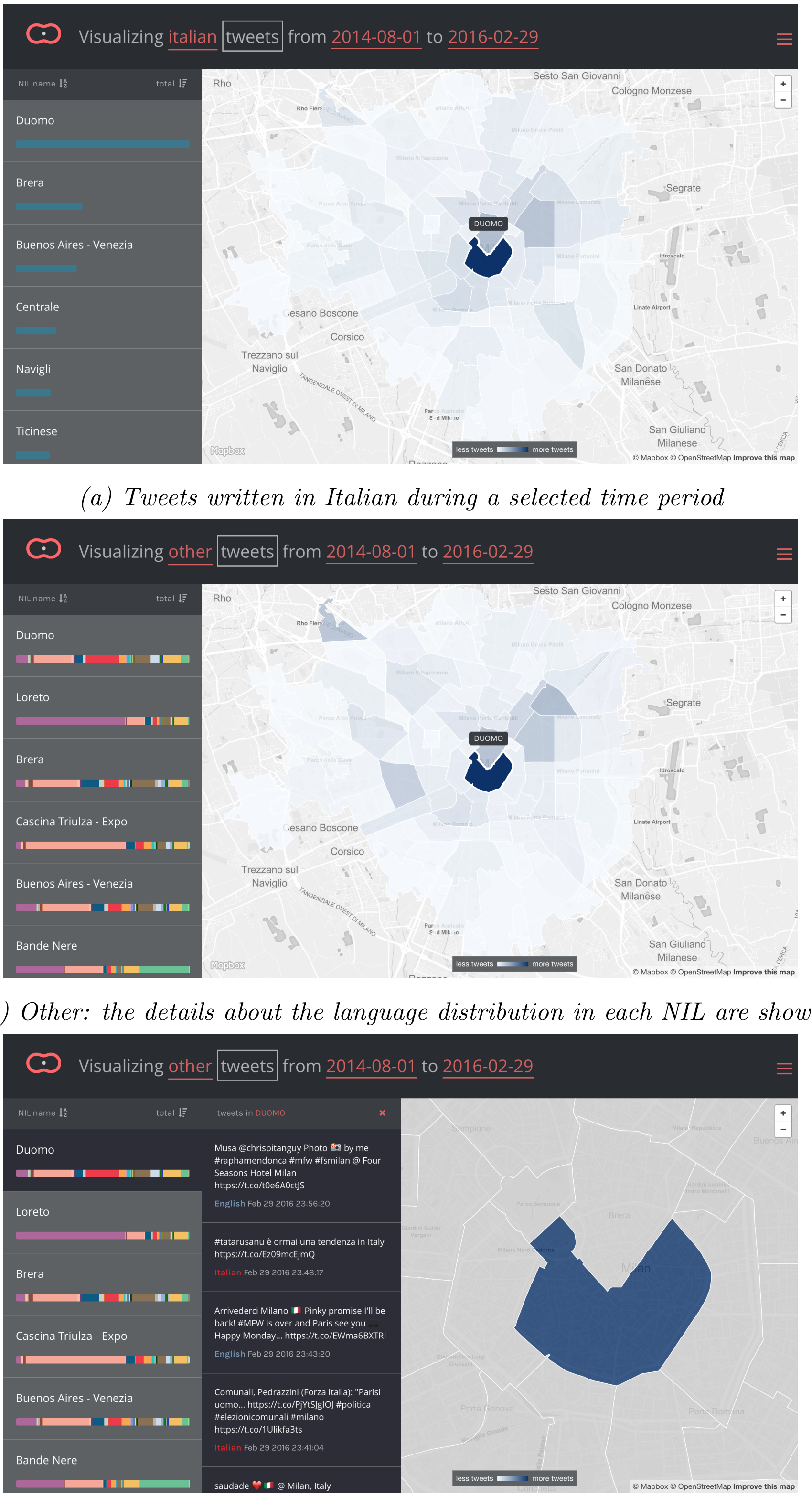
3.2.2 Cities into cities

The section concerning the multilingualism analysis of Twitter for the city of Milan is called "Cities into cities" and is in turn divided into two visualisations: "Explore Tweets" and "Analyse Tweets".

To have a first insight of the data one can click on the "Explore Tweets" button and visualise all the published tweets within the city of Milan since Urbanscope was first conceived. The visualisation is grounded on a map whose NILs are coloured according to the density of geo-referenced tweets posted. The user can interact with the view using different filters: it is possible to select the time window and the macro-language (Italian, English, or Other-languages). One can also easily identify NILs by clicking on the map and access to all the tweets geo-referenced in that zone during the chosen months. If the user selects "Other" among the macro-language option, then the details about the language distribution in each NIL are shown.

In order to make the interface more user-friendly, all the selections made by the users are recapped in a sentence in natural language form (e.g. "Visualizing English tweets from 2014-08-01 to 2015-11-30"). Figure 3.4a, Figure 3.4b and Figure 3.4c show some examples.

The visualisation named "Analysed Tweets" shows the details about the density of



Musa @chrispitanguy Photo 📸 by me #raphamendonca #mfw #fsmilan @ Four Seasons Hotel Milan https://t.co/t0e6A0ctJS

English Feb 29 2016 23:56:20

#tatarusanu è ormai una tendenza in Italy https://t.co/Ez09mcEjmQ

Italian Feb 29 2016 23:48:17

Arrivederci Milano 🇮🇹 Pinky promise I'll be back! #MFW is over and Paris see you 🔜 Happy Monday... https://t.co/EWma6BXTRI

English Feb 29 2016 23:43:20

Comunali, Pedrazzini (Forza Italia): "Parisi uomo... https://t.co/PjYtSJgIOJ #politica #elezionicomunali #milano https://t.co/1Ulikfa3ts

Italian Feb 29 2016 23:41:04

saudade ❤️ 🇮🇹 @ Milan, Italy

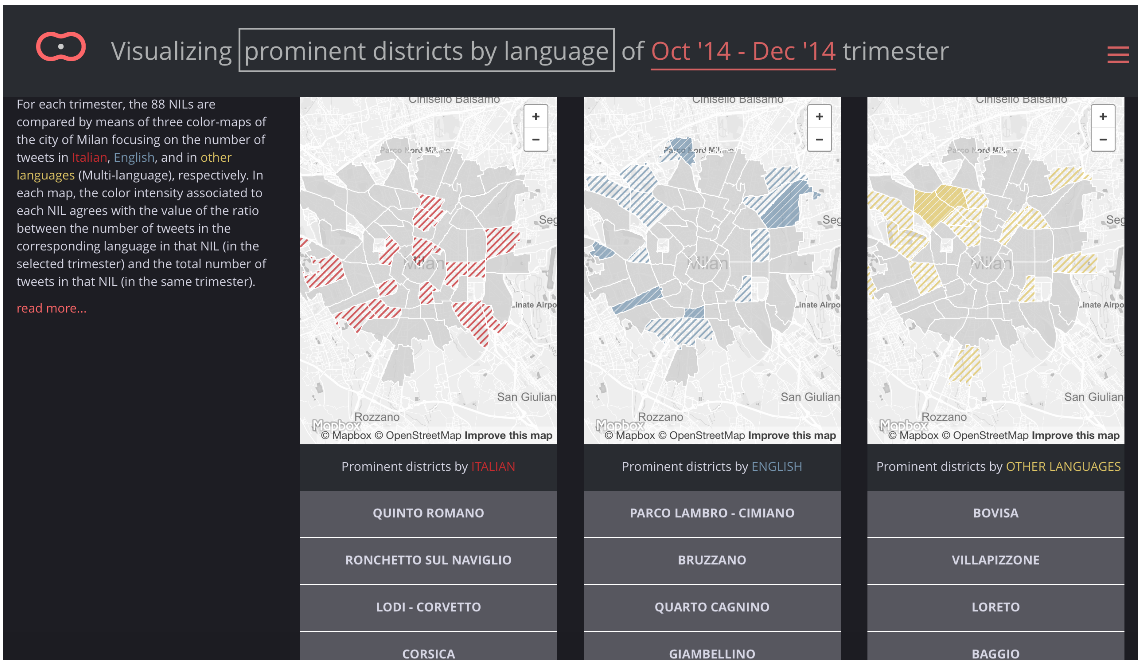
(a) Tweets written in Italian during a selected time period

(b) Other: the details about the language distribution in each NIL are shown

(c) By clicking on the NIL "Duomo" all the tweets in this zone are shown

Figure 3.4: Example of "Explore tweets" in the "Cities into cities" module

tweets in each NIL. For each trimester, three maps are presented in order to compare the 88 NILs: the user can select the quarter he prefers to get a zoomed view. NILs are coloured due to the predominance of Italian, English, or other languages. Precisely, “for each language dark-coloured NILs indicate NILs with an exceptional extreme excess of tweets in the corresponding language with respect to the other NILs in the same trimester; light-colored NILs indicate NILs with a moderate excess of tweets in the corresponding language with respect to the other NILs in the same trimester; gray-colored NILs indicate NILs that do not present any notable excess of tweets in the corresponding language with respect to the other NILs in the same trimester. Transparent NILs are instead the ones for which the overall counting of tweets does not allow reliable estimates of the language-to-total ratio in all trimesters.” as it is explicated in the page “Analysis methods”. In fact, as previously said, the “Analyse” section of any lens helps the user to understand the visualisation giving extra information about the scientific analysis made on the data. An example of the visualisation presented in this section is shown in Figure 3.5.


Visualizing prominent districts by language of Oct '14 - Dec '14 trimester

For each trimester, the 88 NILs are compared by means of three color-maps of the city of Milan focusing on the number of tweets in Italian, English, and in other languages (Multi-language), respectively. In each map, the color intensity associated to each NIL agrees with the value of the ratio between the number of tweets in the corresponding language in that NIL (in the selected trimester) and the total number of tweets in that NIL (in the same trimester).

read more...

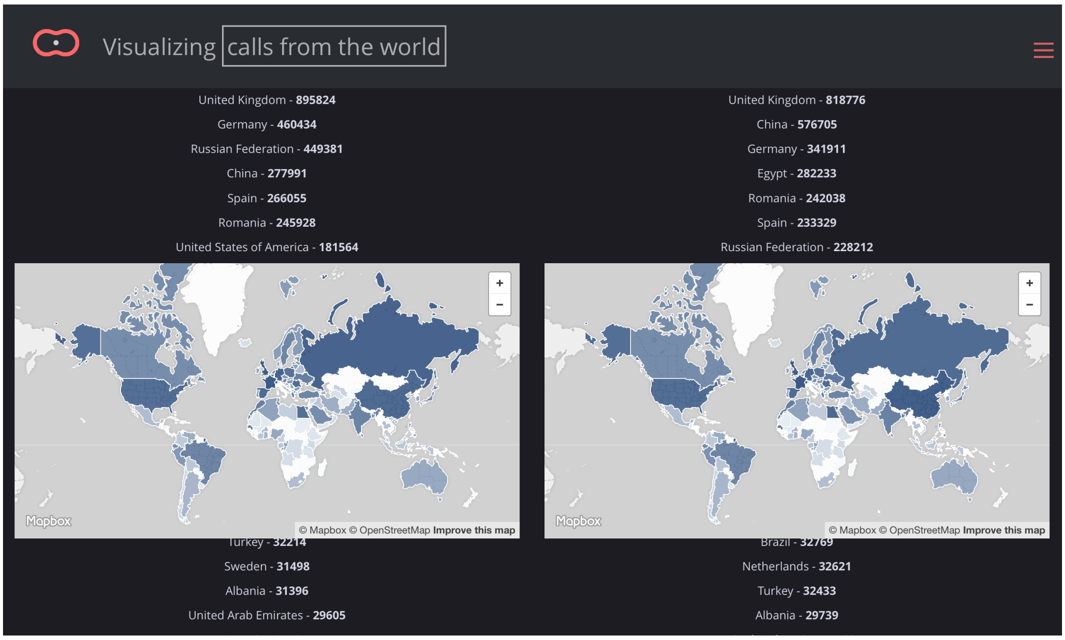
Prominent districts by ITALIAN	Prominent districts by ENGLISH	Prominent districts by OTHER LANGUAGES
QUINTO ROMANO	PARCO LAMBRO - CIMIANO	BOVISA
RONCHETTO SUL NAVIGLIO	BRUZZANO	VILLAPIZZONE
LODI - CORVETTO	QUARTO CAGNINO	LORETO
CORSICA	GIAMBELLINO	BAGGIO

Figure 3.5: An example of the “Analysed Tweets” in the “Cities into cities” section

3.2.3 City and the world

This is the section, as the title suggests, dedicated to find a link between the NILs of Milan and the countries of the world. In the Urbanscope project, this connection is made possible by observing and studying the phone calls that Milan makes and receives in a

selected month.
As one can guess, there are some interesting questions that arise talking about phone traffics. One of the first is for sure if there is some difference in ranking the countries according to the incoming and outgoing calls to/from the main city of the north of Italy. One can find an answer just looking at the exploring session called “Explore Calls”, that is the only one present for the moment. This is divided in two parts, structured in the following way: a list of nations ordered by the number of calls that make (left part) and that receives (right part) to/from Milan. To let the user play with the countries there are also two maps (one for each part), where the states are coloured with different shades of blue, according to the intensity of the phone traffic that characterised them. If one wants to select a particular country on the map, a sign will show up with the name and the number of calls of the state in one month.


Visualizing calls from the world
+
−
Mapbox
© Mapbox © OpenStreetMap **Improve this map**

United Kingdom - **895824**
Germany - **460434**
Russian Federation - **449381**
China - **277991**
Spain - **266055**
Romania - **245928**
United States of America - **181564**
Turkey - **32214**
Sweden - **31498**
Albania - **31396**
United Arab Emirates - **29605**

United Kingdom - **818776**
China - **576705**
Germany - **341911**
Egypt - **282233**
Romania - **242038**
Spain - **233329**
Russian Federation - **228212**
Brazil - **32769**
Netherlands - **32621**
Turkey - **32433**
Albania - **29739**

Figure 3.6: Map of the world coloured according to the incoming calls (left) and outgoing calls (right)

As one can notice from Figure 3.6, Italy is in white, not because, of course, it does not receive or make calls, but due to the fact that from the dataset it is not possible to distinguish if Milan is calling itself or another part of Italy.
The Urbanscope team is already working to create for this section also the part “Analyse”. Phone data are perfect to be treated in the modelling perspective of Network Analysis. The aim of this part, in fact, is to create a graph where the vertices are NILs and countries, which are connected by an edge if there is phone traffic between them. From

this kind of network it will be easy to recognise which are the parts of Milan more active in calling foreign states and, on the other hand, which are the countries of the world which are particularly linked to the Italian city.
Another question that will find an answer creating the “Analyse” section is if there is a distinction in calls between working days or festivities. Does it change something in the intensity of phone traffic? Does any NIL or country become more or less important during the weekends? The answer of these questions will be clear as soon as this new section will be released. Another step forward is trying to analyse also the behaviour of phone calls based on the type: business or not business. As soon as a new dataset is available, the Urbanscope team will be able to study also this kind of traffic and, moreover, the difference in sex and age range of the Milan citizens.
Also for this part, like for the other modules, a monthly horizon is given, so it will be also interesting to understand how the parts of Milan and the world change in time.

3.2.4 Top venues

The section concerning the most visited locations in Milan is called “Top venues” and, for the moment, it is only characterized by the “Explore” section named “Explore Venues”.

This third view of Urbanscope shows the dynamics of presence and preferences (check-ins) in time. Every venue has been categorised in the following groups: Event Arts and Entertainment, Food, Health and Fitness, Hotel, Monuments and Building, Nightlife Spot, Store. Therefore, it is possible to know which are the more attractive venues for entertainment or which monuments Foursquare users visit the most.

This time the visualisation is also based on a map but there is not any NILs division. Precisely, the top 20 venues in the selected month are shown and compared with the top 20 of the previous and following month. Every location presented in the ranking is coloured according to its number of check-ins: a darker hue means an elevated number of check-ins. Morover, if a venue is in the top 20 of consecutive months, then a connection is established in order to highlight the evolution of the venue itself.

As usual, the user can interact with the view using different filters: it is possible to select the main month and the group in which all the locations have been categorised (All, Event Arts and Entertainment, Food, Health and Fitness, Hotel, Monuments and Building, Nightlife Spot, Store). In order to make the interface more user-friendly, all the selections made by the users are recapped in a sentence in natural language form (e.g. “Visualizing Monuments & Buildings top 20 venues for 2014-10-01”).

One can also easily identify the position of any venue in the top 20 by clicking on its name: it will be displayed on a map in the form of dot. Changing point of view, it is always possible to identify the name of a venue represented on the map by clicking on the corresponding dot. Figure 3.7 shows two examples of visualization for the module just explained.

For the moment, the “Analyse” section is not available.

3.2.5 City magnets

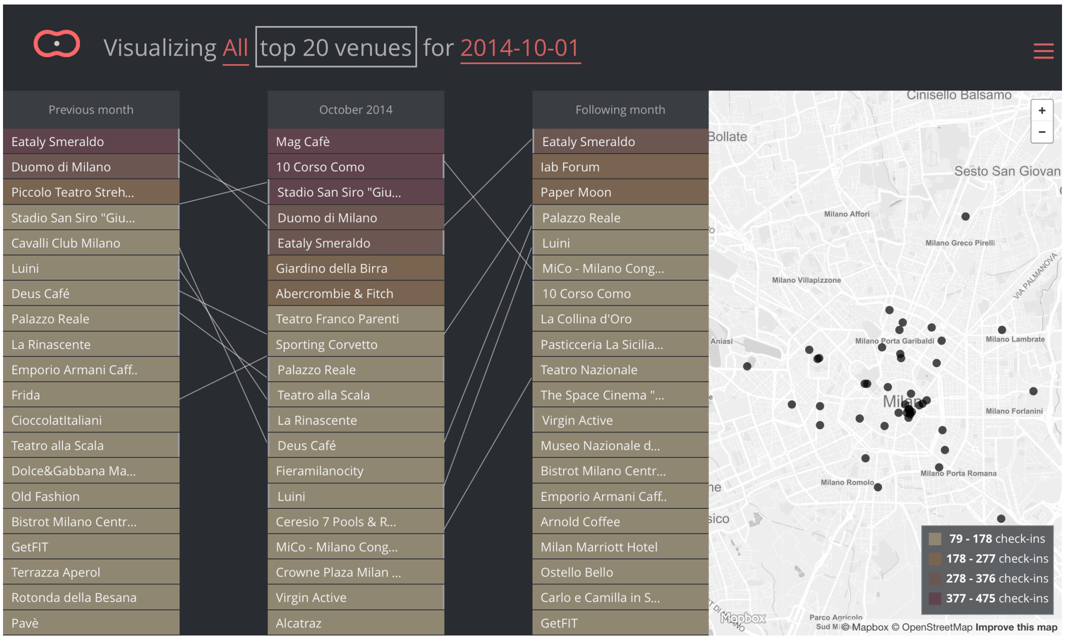
The last module that now characterises the Urbanscope project is called “City magnets”. This visualization is based on the same data of the module “Top venues”. The data are taken from the digital platform of Foursquare that allows people to check-in, i.e. share their position with friends. By mapping all the check-ins it is possible to reveal which are the attractors in Milan where the physical and the digital layers overlap. Moreover, due to the fact that every “check-inable” venue is categorized by Foursquare itself, one can also understand which venue category is the most fashionable. It is possible to discover which NILs are popular and interesting for the digital community.

For the moment, the lens is only characterized by the “Explore” section named “Explore Checkins”. The visualisation here presented is really similar to the one in the “Explore Tweets” section. By clicking on the "Explore Checkins" button, one can visualise all the check-ins within the city of Milan. The visualization is grounded on a map whose NILs are apparently invisible. The user can interact with the view using different filters: it is possible to select the NIL, the time window and the group in which all the check-ins have been categorised. In fact, as seen in “Top venues” lens, all the check-ins have been grouped, on the base of the venue they are referring to, in the following groups: Event Arts and Entertainment, Food, Health and Fitness, Hotel, Monuments and Building, Nightlife Spot, Store. It is important to notice how this categorisation differs from the one proposed by Foursquare: the Urbanscope’s grouping identifies only 7 groups while the original one identifies more than 300 groups.

As usual, in order to make the interface more user-friendly, all the selections made by the users are recapped in a sentence in natural language form (e.g. “Visualizing Monuments & Buildings check-ins from 2014-08-01 to 2016-03-31”). To select NILs, instead, it is necessary to interact with the left window in which all the NILs are listed with their characteristic number of check-ins, i.e. the number of persons that checked their presence. thus, one can easily identify NILs by clicking on the name shown in the left of the visualisation and access to all the checked venues geo-referenced in that zone during the chosen months.

To show a magnetic zone in the city it has been decided to use a sort of red cloud. Zone with an high number of check-ins are identified by a darker red cloud. Figure 3.8 explains with some examples what it has been explicated by words.

At the moment, there is no “Analyse” section, in fact, the main task of this thesis

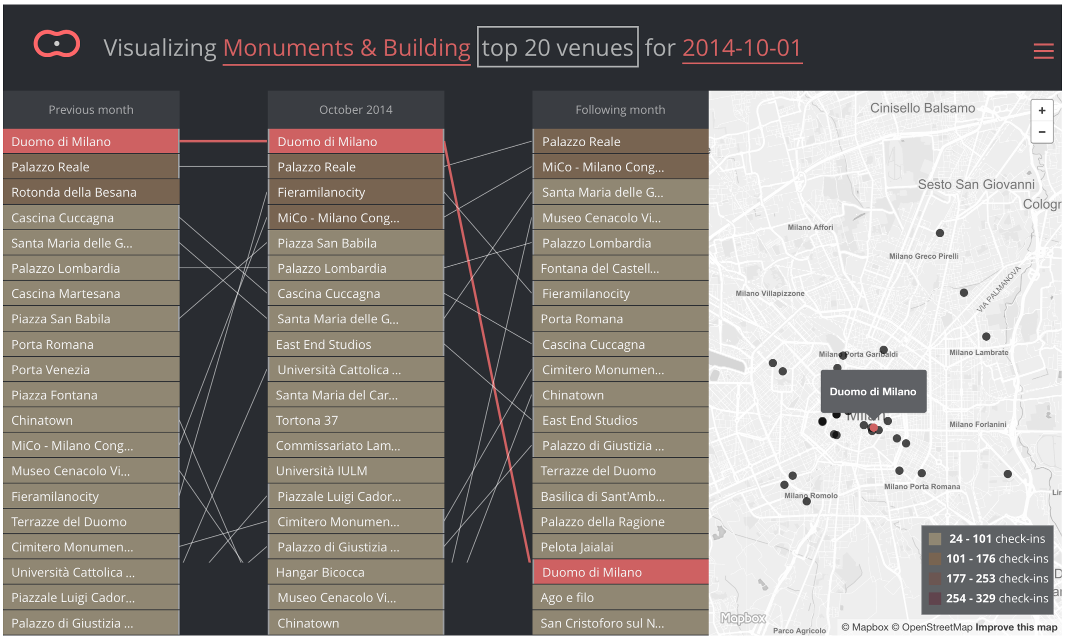


Visualizing All top 20 venues for 2014-10-01

Previous month	October 2014	Following month
Eataly Smeraldo	Mag Cafè	Eataly Smeraldo
Duomo di Milano	10 Corso Como	lab Forum
Piccolo Teatro Streh...	Stadio San Siro "Giu...	Paper Moon
Stadio San Siro "Giu...	Duomo di Milano	Palazzo Reale
Cavalli Club Milano	Eataly Smeraldo	Luini
Luini	Giardino della Birra	MiCo - Milano Cong...
Deus Café	Abercrombie & Fitch	10 Corso Como
Palazzo Reale	Teatro Franco Parenti	La Collina d'Oro
La Rinascente	Sporting Corvetto	Pasticceria La Sicilia...
Emporio Armani Caff..	Palazzo Reale	Teatro Nazionale
Frida	Teatro alla Scala	The Space Cinema "...
CioccolatItaliani	La Rinascente	Virgin Active
Teatro alla Scala	Deus Café	Museo Nazionale d...
Dolce&Gabbana Ma...	Fieramilanocity	Bistrot Milano Centr...
Old Fashion	Luini	Emporio Armani Caff..
Bistrot Milano Centr...	Ceresio 7 Pools & R...	Arnold Coffee
GetFIT	MiCo - Milano Cong...	Milan Marriott Hotel
Terrazza Aperol	Crowne Plaza Milan ...	Ostello Bello
Rotonda della Besana	Virgin Active	Carlo e Camilla in S...
Pavè	Alcatraz	GetFIT

▨ 79 - 178 check-ins
▨ 178 - 277 check-ins
▨ 278 - 376 check-ins
▨ 377 - 475 check-ins

(a) "All" top 20 venues



Visualizing Monuments & Building top 20 venues for 2014-10-01

Previous month	October 2014	Following month
Duomo di Milano	Duomo di Milano	Palazzo Reale
Palazzo Reale	Palazzo Reale	MiCo - Milano Cong...
Rotonda della Besana	Fieramilanocity	Santa Maria delle G...
Cascina Cuccagna	MiCo - Milano Cong...	Museo Cenacolo Vi...
Santa Maria delle G...	Piazza San Babila	Palazzo Lombardia
Palazzo Lombardia	Palazzo Lombardia	Fontana del Castell...
Cascina Martesana	Cascina Cuccagna	Fieramilanocity
Piazza San Babila	Santa Maria delle G...	Porta Romana
Porta Romana	East End Studios	Cascina Cuccagna
Porta Venezia	Università Cattolica ...	Cimitero Monumen...
Piazza Fontana	Santa Maria del Car...	Chinatown
Chinatown	Tortona 37	East End Studios
MiCo - Milano Cong...	Commissariato Lam...	Palazzo di Giustizia ...
Museo Cenacolo Vi...	Università IULM	Terrazze del Duomo
Fieramilanocity	Piazzale Luigi Cador...	Basilica di Sant'Amb...
Terrazze del Duomo	Cimitero Monumen...	Palazzo della Ragione
Cimitero Monumen...	Palazzo di Giustizia ...	Pelota Jaialai
Università Cattolica ...	Hangar Bicocca	Duomo di Milano
Piazzale Luigi Cador...	Museo Cenacolo Vi...	Ago e filo
Palazzo di Giustizia ...	Chinatown	San Cristoforo sul N...

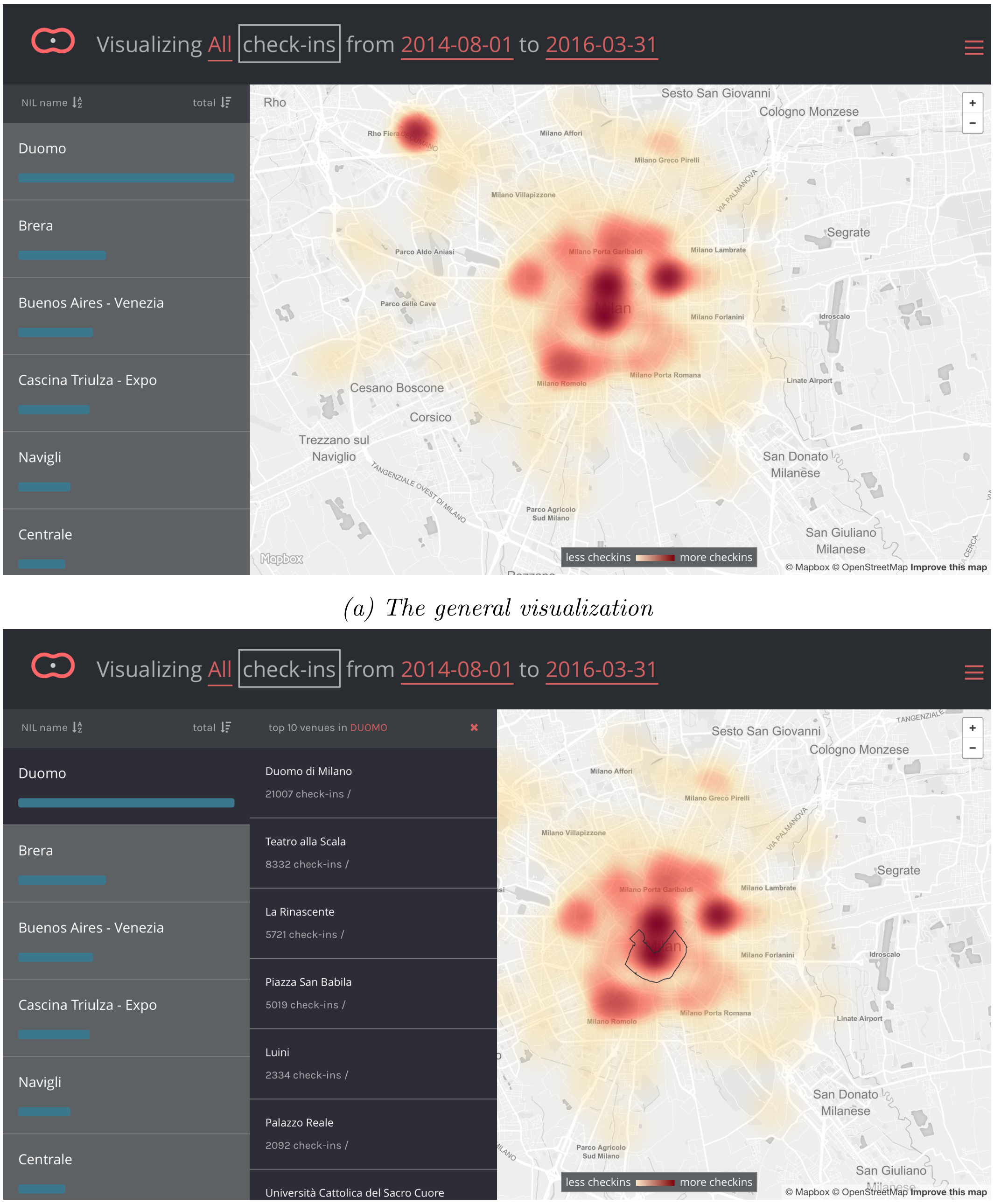
▨ 24 - 101 check-ins
▨ 101 - 176 check-ins
▨ 177 - 253 check-ins
▨ 254 - 329 check-ins

(b) "Monuments & Buildings" top 20 venues with "Duomo di Milano" highlighted

Figure 3.7: Example of "Explore Venues" in the "Top venues" module

is to create a brand new, interesting analysis for the “City Magnets” lens. Therefore, it has been decided to use the Cheng and Church’s Biclustering algorithm in order to find biclusters: groups of NILs considered similar for particular groups of Foursquare’s categories. Moreover, the application of Biclustering techniques to this kind of data is a innovative.

In the next chapters everything will be deeply explicated. The process of description will start from how the dataset is composed to the final design layout of the “Analyse” section of the “City magnets” module.



Visualizing All check-ins from 2014-08-01 to 2016-03-31

NIL name | total

Duomo

Brera

Buenos Aires - Venezia

Cascina Triulza - Expo

Navigli

Centrale

(a) The general visualization

Visualizing All check-ins from 2014-08-01 to 2016-03-31

NIL name | total

Duomo

Brera

Buenos Aires - Venezia

Cascina Triulza - Expo

Navigli

Centrale

top 10 venues in DUOMO

Duomo di Milano
21007 check-ins /

Teatro alla Scala
8332 check-ins /

La Rinascente
5721 check-ins /

Piazza San Babila
5019 check-ins /

Luini
2334 check-ins /

Palazzo Reale
2092 check-ins /

Università Cattolica del Sacro Cuore

(b) The visualization detailed for the NIL called "Duomo"

Figure 3.8: Example of "Explore Checkins" in the "City magnets" module

Chapter 4

Preprocessing and analysis

We are not to attempt to hack off parts like a clumsy butcher.

Plato

In this chapter the reader will get acquainted with everything considered fundamental to perform the Cheng and Church's Biclustering algorithm. However, before applying the method described item by item in Chapter 2, it is necessary to understand the raw data and to process them in order to have the typical matricial structure. Moreover, due to the high number of venue categories proposed by the digital platform of Foursquare it has been decided to select only the most important ones and to group them in clusters. Therefore, in Section 4.1 the source of the data, i.e. the Foursquare digital platform, will be introduced from a functional and historical point of view.

In Section 4.2 the dataset is explained item by item and then processed in order to have a more convenient data structure called array 3D.

Now that everything has been set up, the reader will receive some general information about the Clustering procedure. Thanks to a vectorial clustering approach, the dataset will be modified by cutting down all the categories resulting not important to describe the urban dynamic. In addition, different kind of procedures will be shown and performed in order to study both NILs and categories. Therefore, a polished managerial point of view will be fundamental to interpret new data and their resulting groups.

Every result will be shown and commented.

4.1 Foursquare as a source of data

This section aims at describing Foursquare, the digital platform which the data used in the analysis are founded on. Therefore, to understand the data, it is necessary to comprehend how Foursquare works and, in general, the history of this web and mobile phone application.

Precisely, Foursquare is a local search and discovery service mobile app which provides search results for its users. By taking into account the places one goes, the things he likes or recommends and other users' trusted advices, Foursquare provides recommendations for new locations and commercial activities. The service was created in late 2008 and launched in 2009 by Dennis Crowley and Naveen Selvadurai, in order to find a new way to make people use their mobile phones to interact with the environment. Using some terms previously introduced in the context of Urbanscope, Crowly and Selvadurai wanted to create a direct connection between digital and material-urban layers.

Until late July 2014, Foursquare featured a social networking system that enabled a user to share their location with friends, via the "check in" - a user would manually tell the application when they were at a particular location using a mobile website, text messaging, or a device-specific application by selecting from a list of venues the application locates nearby. Every venue is categorised in groups. Italian Restaurant, Opera House, Sushi Bar, Brewery and Baseball Stadium are only some of the groups considered.

Figure 4.1: Foursquare logo

The checking-in ability is the core element of our dataset and analysis. In fact, by studying the number of check-ins it is possible to discover many things like, for examples, the city magnets, which are locations highly attractive, or the most attractive venue categories. As said before, all the efforts have been principally directed in finding biclusters. However in May 2014, the company launched Swarm, a companion app to Foursquare, that reimagined the social networking and location sharing aspects of the service as a separate application. On August 7, 2014 the company launched Foursquare 8.0, the completely new version of the service which finally removed the check-in and location sharing entirely, to focus on local search. Therefore, in order to share your position through Foursquare, now it is mandatory to connect the digital platform with Swarm. This new necessity, fortunately, does not impact in any way on the collection of data.

Now that Foursquare has been introduced, it is possible to pass to a more analytical description of the data.

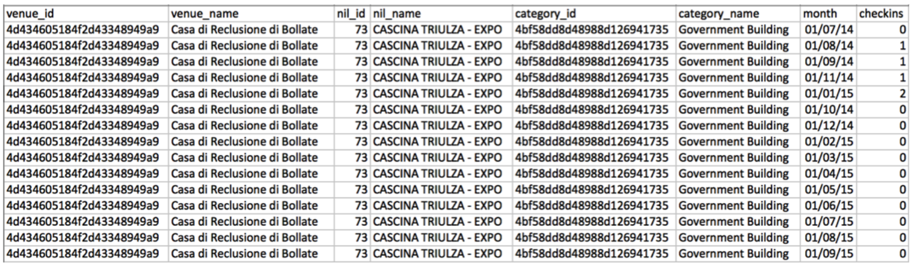
4.2 The data

The dataset on which the analysis is based is finally introduced. It can be described as a dataframe composed by 301770 rows or observations and 8 columns, also called features or variables.

The variables which describe the dataset are:

- venue_id - the Foursquare code used to identify the venue;
- venue_name - the name of the venue as proposed by Foursquare;
- nil_id - the number chosen by the municipality of Milan to identify the NIL;
- nil_name - the name chosen by the municipality of Milan to identify the NIL;
- category_id - the Foursquare code used to identify the venue category;
- category_name - the name of the venue category as proposed by Foursquare;
- month - the month taken in analysis from the first day of the month to the last;
- checkins - the number of checkins concerning the venue and the month;

Figure 4.2 shows the first rows of the dataset.



venue_id	venue_name	nil_id	nil_name	category_id	category_name	month	checkins
4d434605184f2d43348949a9	Casa di Reclusione di Bollate	73	CASCINA TRIULZA - EXPO	4bf58dd8d48988d126941735	Government Building	01/07/14	0
4d434605184f2d43348949a9	Casa di Reclusione di Bollate	73	CASCINA TRIULZA - EXPO	4bf58dd8d48988d126941735	Government Building	01/08/14	1
4d434605184f2d43348949a9	Casa di Reclusione di Bollate	73	CASCINA TRIULZA - EXPO	4bf58dd8d48988d126941735	Government Building	01/09/14	1
4d434605184f2d43348949a9	Casa di Reclusione di Bollate	73	CASCINA TRIULZA - EXPO	4bf58dd8d48988d126941735	Government Building	01/11/14	1
4d434605184f2d43348949a9	Casa di Reclusione di Bollate	73	CASCINA TRIULZA - EXPO	4bf58dd8d48988d126941735	Government Building	01/01/15	2
4d434605184f2d43348949a9	Casa di Reclusione di Bollate	73	CASCINA TRIULZA - EXPO	4bf58dd8d48988d126941735	Government Building	01/10/14	0
4d434605184f2d43348949a9	Casa di Reclusione di Bollate	73	CASCINA TRIULZA - EXPO	4bf58dd8d48988d126941735	Government Building	01/12/14	0
4d434605184f2d43348949a9	Casa di Reclusione di Bollate	73	CASCINA TRIULZA - EXPO	4bf58dd8d48988d126941735	Government Building	01/02/15	0
4d434605184f2d43348949a9	Casa di Reclusione di Bollate	73	CASCINA TRIULZA - EXPO	4bf58dd8d48988d126941735	Government Building	01/03/15	0
4d434605184f2d43348949a9	Casa di Reclusione di Bollate	73	CASCINA TRIULZA - EXPO	4bf58dd8d48988d126941735	Government Building	01/04/15	0
4d434605184f2d43348949a9	Casa di Reclusione di Bollate	73	CASCINA TRIULZA - EXPO	4bf58dd8d48988d126941735	Government Building	01/05/15	0
4d434605184f2d43348949a9	Casa di Reclusione di Bollate	73	CASCINA TRIULZA - EXPO	4bf58dd8d48988d126941735	Government Building	01/06/15	0
4d434605184f2d43348949a9	Casa di Reclusione di Bollate	73	CASCINA TRIULZA - EXPO	4bf58dd8d48988d126941735	Government Building	01/07/15	0
4d434605184f2d43348949a9	Casa di Reclusione di Bollate	73	CASCINA TRIULZA - EXPO	4bf58dd8d48988d126941735	Government Building	01/08/15	0
4d434605184f2d43348949a9	Casa di Reclusione di Bollate	73	CASCINA TRIULZA - EXPO	4bf58dd8d48988d126941735	Government Building	01/09/15	0

Figure 4.2: The first rows of the dataset referred to the venue "Casa di reclusione di Bollate"

One can notice that among the eight features, six come directly from the Foursquare database whereas two, nil_id and nil_name, refer to the urban structure of Milan. As said in Chapter 3, Milan has been divided in 88 NILs described by the features nil_id and nil_name. However not every NIL is in the data. In fact the dataset considers 87 NILs because the 40th NIL, called Ronchetto delle Rane, has never been checked by any user.

Focusing on the venues, one can easily understand that, as for the case of the NILs, venue_id and venue_name simply give the same information: the identification of the venue. In the dataset analysed, 18439 different venues have been considered. Venues, in effect, are the main elements that characterise the “Top venue” lens in the Urbanscope project.

Instead, the number of categories taken into account is 274. Therefore there are 274 different category_name and a same number of category_id because, as usual, they just repeat the same information. The set of categories is really variegated and it stretched from “Accessories Store” to “Yoga Studio”.

As the name is suggesting, the month which the row is referring to is shown in the month feature. The dataset covers exactly 15 months: from July 2014 to September 2015. Therefore the analysis here proposed is able to detect all the changes due to EXPO2015, the important exposition which took place from May 2015 to October 2015.

The core information is the number of check-ins and it is contained in the checkins variable. It is important to notice that each couple of venue and month correspond to a different row and to a different number of check-ins. Therefore, for every single venue fifteen different rows with their own number of check-ins are available.

At this point, one can notice that the dataframe presents variables highly redundant or useless for the analysis here proposed. Precisely, nil_name and nil_id, as said before, are two different and interchangeable ways to identify NILs. The situation is the same for category_name and category_nil. Instead, regarding the venues, all the information contained in venue_name and venue_id are not necessary for the analysis here presented. In fact, this thesis has no will to study venues as done in the “Top venues” lens. For these reasons it has been decided to modify the original dataset by cutting down some variables considered redundant or not interesting for the analysis. Thus, the final dataframe is composed by 301770 rows and 4 columns. The variables are:

- nil_name
- category_name
- month
- checkins

However, even if the data have been simplified, it is still necessary to perform some preprocessing operations, explicitly shown in Section 4.2.1.

4.2.1 Data preprocessing

The new simplified dataset needs to be processed in order to start Clustering and Biclustering. In fact, in order to perform different methods of grouping, it is mandatory to have all the data in a matricial structure. To solve this problem it has been decided to reorder the data in 15 matrices A_K with $K = 1, \dots, 15$, one for every month, with 87 rows and 274 columns. Every row is a different NIL while every column is one single category. The element a_{ij}^K, which is the intersection between the i-th row and the j-th column of A_K, the matrix corresponding to the K-th month, is the number of check-ins in venues belonging to the j-th category in the i-th NIL.

In Table 4.1 a portion of one of the data matrix just explained is shown.

	$\cdots$	Vietnamese Restaurant	Volleyball Court	Whisky Bar	Wine Bar	$\cdots$
ADRIANO	$\cdots$	0	0	0	0	$\cdots$
AFFORI	$\cdots$	0	0	1	0	$\cdots$
BAGGIO	$\cdots$	1	0	0	2	$\cdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

Table 4.1: A portion of the dataset represented in a matricial structure

All the 15 matrices each referring to a single month are put together in a new structure called Array 3D. The array 3D is a three dimensional structure that can be obtained by overlapping 2D matrices. In our case, the obtained array 3D is composed by 15 layers and each layer is an A_K. The reader can imagine it as a book of fifteen pages where each page describes one month. Figure 4.3 gives an example of an array 3D which counts three layers.

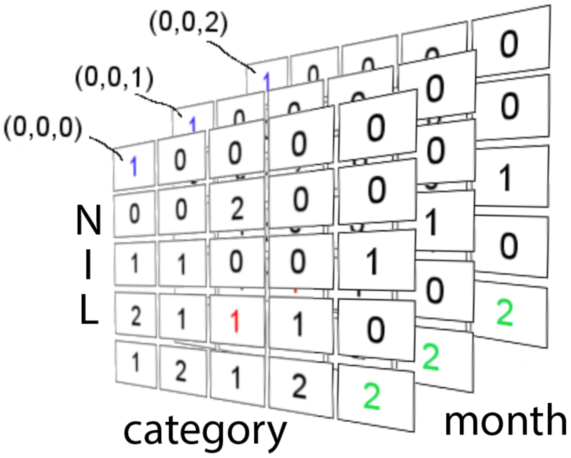


Figure 4.3: An array 3D composed by three layers

The interpretation of array 3D as an array of matrices, however, is not the only possible interpretation. In fact, it is possible to see the structure as a single matrix with vectorial elements. Therefore, the array 3D here presented can be viewed as a matrix with 87 rows and 274 columns where each element a_{ij} is a vector containing 15 values, precisely, the single numbers of check-ins in each month.

4.3 Clustering the data

The main task of this section is to use different Clustering approaches to perform a grouping of the data. One, in fact, can decide to cluster the NILs or, by transposing the data matrix, to cluster the categories but, in both the cases, it is possible to follow more than one way. The routes taken in this thesis are three:

- a monthly approach - the clustering procedure is performed on each layer (month) of the array 3D in order to have a monthly description;
- a vectorial approach - the clustering procedure is performed directly on the array 3D viewed as a single matrix with vectorial elements;
- a summed approach - the clustering procedure is performed on a single matrix resulting from the sum of all the single layers (months) composing the array 3D;

All the three approaches have been performed both on NILs and categories and they will be compared and described item by item in the following sections. However there exists some common elements that can be introduced right now. One of this element is the use hierarchical algorithms to perform Clustering. The hierarchical Clustering is a method of cluster analysis which seeks to build a hierarchy of clusters. Strategies for hierarchical Clustering generally fall into two types: agglomerative or divisive. In general, the merges and splits are determined in a greedy manner and the results are usually presented in dendrograms. From the study of the resulting dendrograms it is possible to visually decide the number of clusters.

However, agglomerative clustering requires a measure of dissimilarity between sets of observations to work. This is achieved by using an appropriate metric and a linkage criterion which specifies the dissimilarity of sets as a function of the pairwise distances of observations in the sets. Therefore, the choice of a correct metric is fundamental to define the shape of the clusters. In the case here studied, the Euclidean's distance has been used. It is expressed in the following way:

$$\parallel \mathbf{a} - \mathbf{b} \parallel_2 = \sqrt{\sum_i (a_i - b_i)^2}$$

where **a** and **b** are two general vectors and a_i and b_i are their i-th elements.

Instead, the linkage criterion determines the distance between sets of observations as a function of the pairwise distances between observations. The most common linkage criteria are the complete-linkage clustering criterion, the single-linkage clustering criterion and the average-linkage clustering criterion.

The complete-linkage clustering criterion defines the "shortest distance" between clusters as the distance between those two elements (one in each cluster) that are farthest away from each other. The distance between cluster sets A and B can be described by the following expression:

$$D(A, B) = \max(d(a, b) : a \in A, b \in B)$$

where $d(a, b)$ denotes the distance between the two farthest elements a and b.

The single-linkage clustering criterion, instead, defines the "shortest distance" between clusters as the distance between those two elements (one in each cluster) that are closest from each other. The distance between clusters A and B is described by the expression

$$D(A, B) = \min(d(x, y) : x \in A, y \in B)$$

where A and B are any two sets of elements considered as clusters, and $d(x, y)$ denotes the distance between the two closest elements x and y.

Instead, the average-linkage clustering criterion defines the "shortest distance" between any two clusters A and B, each of size $|A|$ and $|B|$, as the average of all distances $d(a, b)$ between pairs of objects a in A and b in B, that is, the mean distance between elements of each cluster:

$$D(A, B) = \frac{1}{|A| \cdot |B|} \sum_{a \in A} \sum_{b \in B} d(a, b)$$

The three different linkages just explained are described in Figure 4.4.

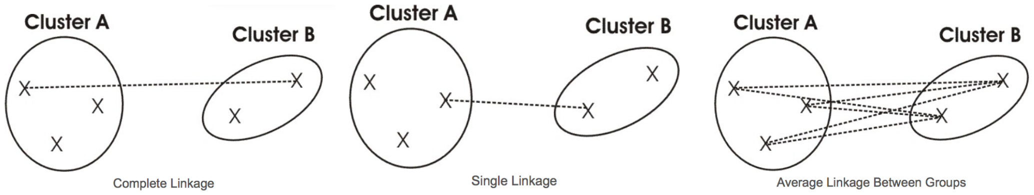


Figure 4.4: The three different main linkage opportunities

Unfortunately, the three criteria just introduced produce, both for NILs and categories Clustering, insufficient result in terms of dendrograms. In fact, the results are characterized by really short branches. Moreover, dendrograms tend to group the data by sorting them by the intensity of the number of check-ins. This is due to the high skewness of the dataset caused by the elevated difference between top and worst rated NILs-categories.

Figure 4.5, Figure 4.6 and Figure 4.7 show what previously described. It is important to remark that only examples for NILs are presented because of the difficulty to distinctly visualize the 274 categories.

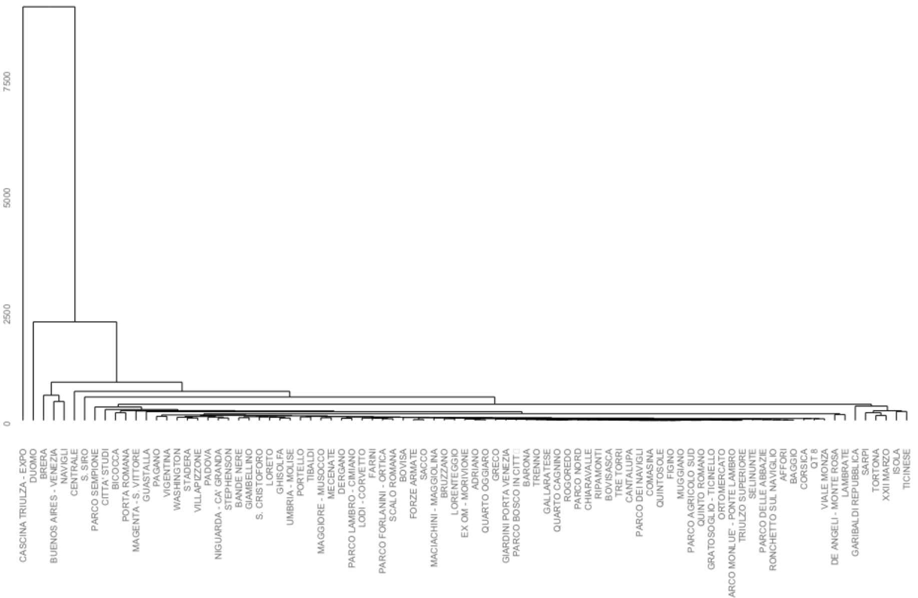


Figure 4.5: Complete-linkage dendrogram for the NILs in September 2015

In order to avoid that too many single elements are clusters in their own, a new linkage criterion has to be selected. The right choice ended up to be the Ward's minimum variance method in term of Lance-Williams update formula. The Lance–Williams algorithms are an infinite family of agglomerative hierarchical clustering algorithms which are represented by a recursive formula for updating cluster distances at each step, i.e. each time a pair of clusters is merged. At each step, it is necessary to optimize the objective function in order to find the optimal pair of clusters to merge. The recursive formula simplifies finding the optimal pair. The curious reader is invited to deepen his knowledge about this method by reading Hartigan, 1972. In our case the Ward method results in dendrograms whose branches are long and well defined. Moreover, all the possible groups count several elements and the single-element clusters are few.

In Section 4.4 and in Section 4.5 the clustering results both for NILs and categories will be displayed. However it is necessary to find a way in order to reduce the high number of categories.

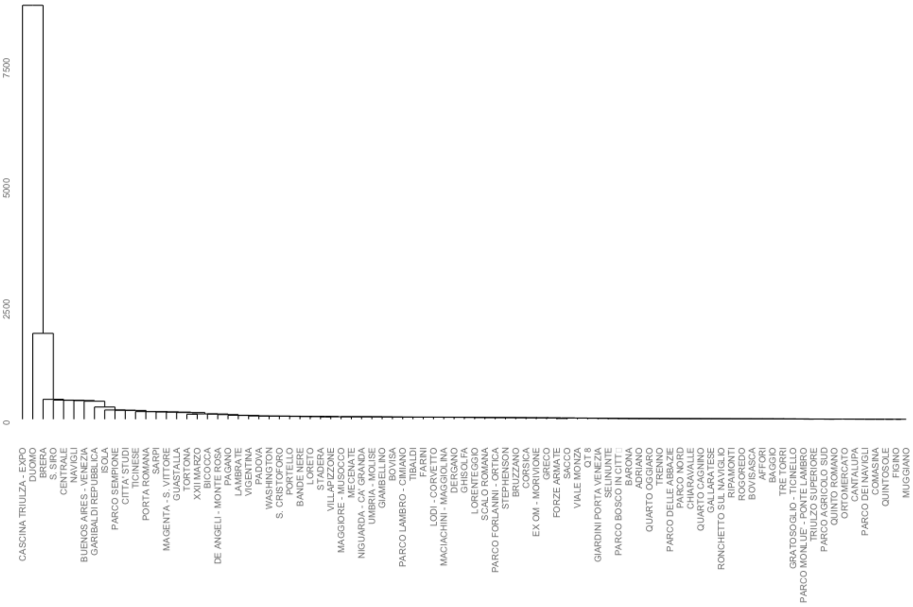


Figure 4.6: Single-linkage dendrogram for the NILs in September 2015

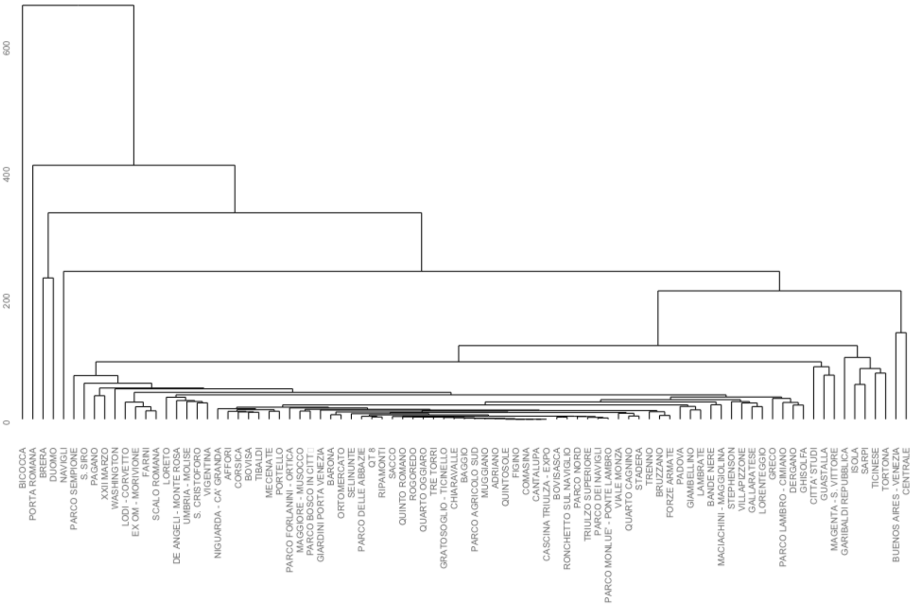


Figure 4.7: Average-linkage dendrogram for the NILs in December 2015

4.4 Clustering Categories

This section aims at describing everything about the Clustering procedures used to group the categories composing the city of Milan. In this way it is possible to change the elevated number of categories in order to reduce the dimension of the dataset and the noise. In fact all the categories that are similarly frequented will be clustered together.

As previously said, the method is based on a hierarchical algorithm with Euclidean's distance and Ward's method and it returns a dendrogram which can be studied to extract the clusters. Clusters are decided by cutting the dendrogram. Unfortunately, differently from the case of NILs, it is not possible to represent clusters in a cohesive way and, as said before, a dimension reduction is absolutely mandatory. In fact it is quite difficult to visualise 274 categories at the same time.Therefore, it is intelligent to suppose that one can delete all the clusters composed by categories with a number of check-ins near zero because they are not popular among Foursquare users.

Considering this fact, the vectorial clustering approach will be presented at first. Then the dataset will be modified and analysed month by month using a monthly approach. After decreasing the number of categories, a descriptive analysis will be carried out.

4.4.1 Vectorial approach to reduce the number of categories

The vectorial approach is used to reduce the number of categories in order to perform a descriptive analysis and Biclustering. This method is based on the view of the array 3D as a matrix whose elements are vectors composed by 15 values, one for each month. One can notice that this view is a summary of all the data. Any Clustering algorithm here performed generates general categories clusters which take into account the 87 NILs and the 15 months. The clusters, therefore, optimally summarise the categories situation for all the time period analysed.

However, having a full description of the whole dataset means not knowing the dynamic of Milan that only a month by month approach can guarantee. In addition, it is necessary to transform the array 3D in a matrix. In fact, in order to be able to use hierarchical methods, the array 3D is not a possible starting point: it has to be modified. The modification here proposed was inspired by the colour-based segmentation, an image vision process of partitioning a digital image into multiple similar regions with respect to colors.

In fact, the color-based segmentation consists in clustering an image according to colours. The colour information is structured in an array 3D composed in the following way: three layers of colours, red, green and blue, in the form of matrix. Therefore, to perform the Clustering procedure it is fundamental to transform the structure in a summarising matrix. It is the same procedure that it has been used in this thesis. The only differences are the number of layers, three instead of fifteen, and the fact that every layer do not present 87 NILs as rows and 274 categories as columns but a same number of rows and columns indicating pixels position.

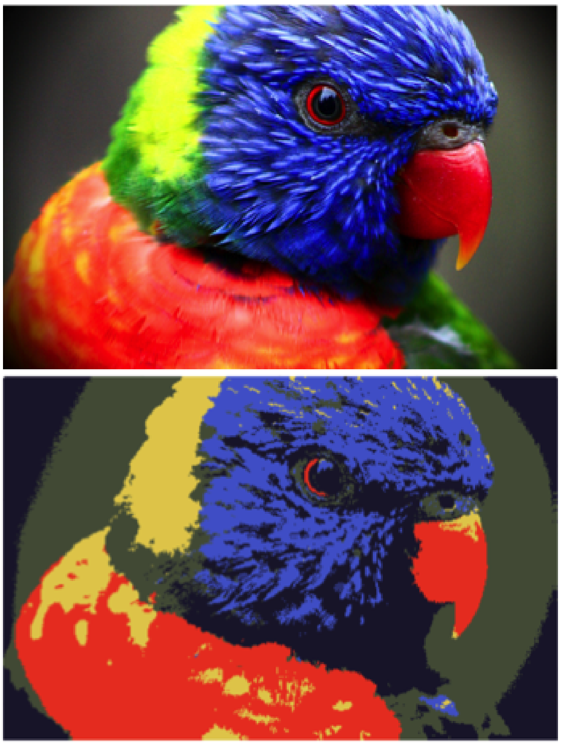


Figure 4.8: A birdy example of color-based segmentation

Figure 4.9 shows the resulting dendrogram of the hierarchical Clustering. One can easily notice how it is difficult to visualise all the labels due to the high numerosity. Therefore, it has been decided to select only the cluster of categories whose number of check-ins is distant from zero. This decision is based on the supposition that all the elements grouped in clusters with very low profile are not important to define the dynamic of the city because they present a very few amount of check-ins.

The selected categories are the ones belonging to the red rectangle. They are 45: Event space, University, Ice Cream Shop, Monument/Landmark, Clothing Store , Art Gallery, Sandwich Place, Breakfast Spot, Burger Joint, Boutique, Gym, Japanese Restaurant, Pub, Bar, Restaurant, Government Building, Lounge, Church, Soccer Stadium, Italian Restaurant, Cocktail, Bar, Department Store, Bakery, Brewery, Chinese Restaurant, Neighbourhood, Supermarket, Sushi Restaurant, Bookstore, Bistro, Theater, Seafood Restaurant, Brasilian Restaurant, Wine Bar, Convention Center, Gym / Fitness Center, Plaza, Food & Drink Shop, Nightclub, Hotel, Café, Pizza Place, Multiplex, Art Museum and Opera House.

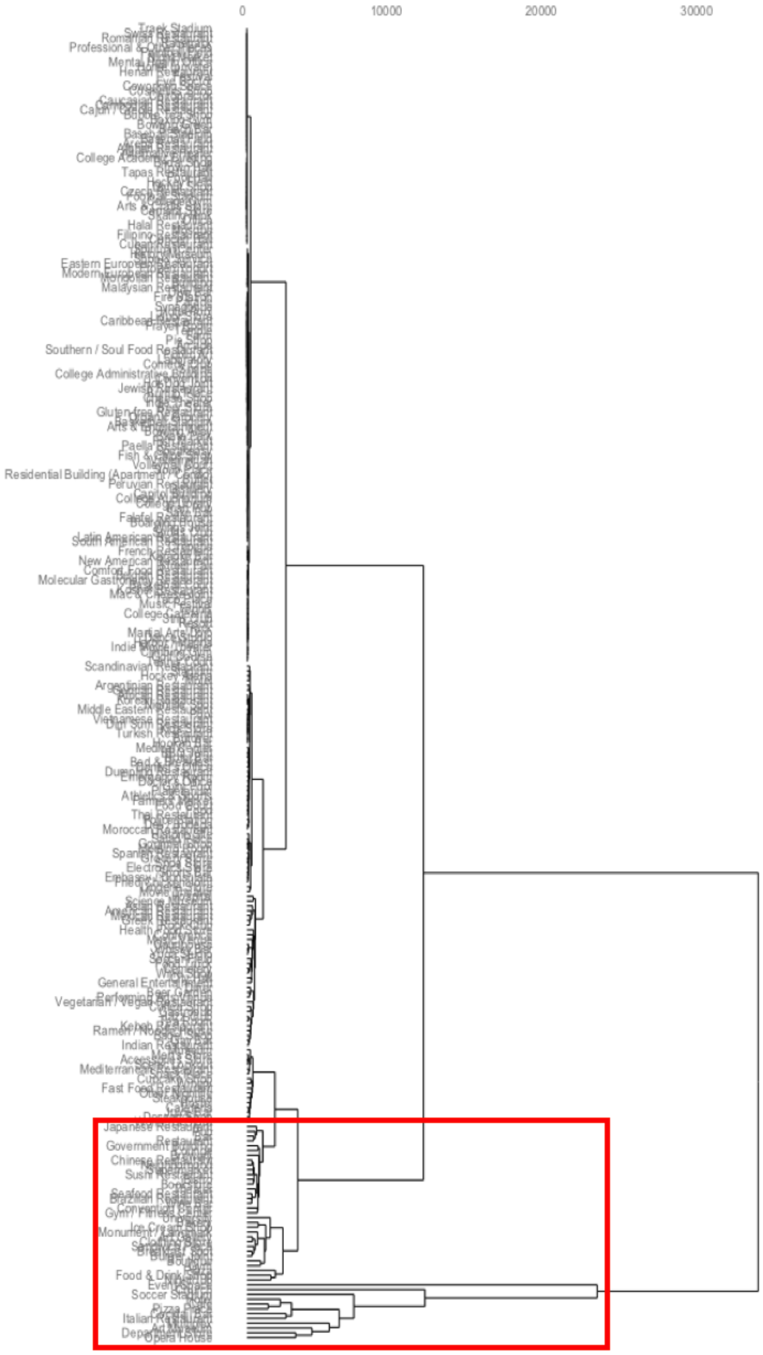


Figure 4.9: The dendrogram resulting from the 274 *categories Clustering*

At this point the dataset has been modified. All the categories which do not belong to the 45 listed above are discarded. The new data has the same three dimensional structure of the array 3D but every layer now counts 87 NILs and 45 selected categories.

The new dataset thus obtained is easier to handle and visualise thanks to the reduced number of categories. Therefore, now it is interesting to cluster the remaining categories in order to ease the Biclustering views introduced in Chapter 5. As usual, a vectorial clustering approach has been used. The resulting dendrogram is shown in Figure 4.10.

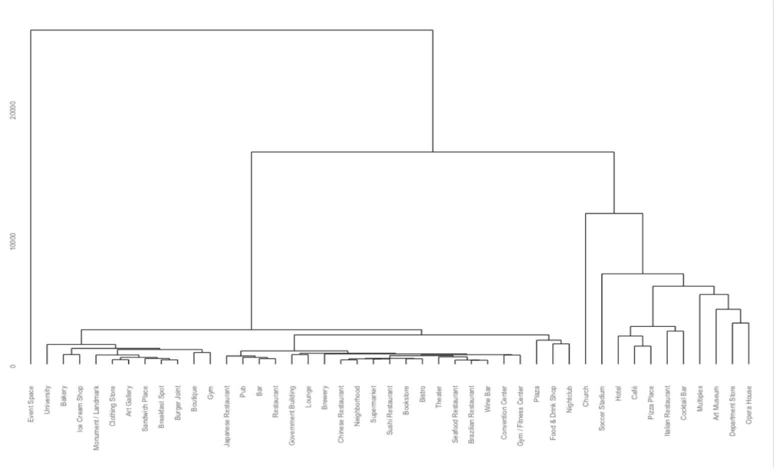


Figure 4.10: The dendrogram resulting from the 45 *categories Clustering*

It has been decided to cut the dendrogram in 6 clusters:

- Event Space;
- University, Bakery, Ice Cream Shop, Monument / Landmark, Clothing Store, Art Gallery, Sandwich Place, Burger Spot, Boutique, Gym, Japanese Restaurant, Pub, Bar, Restaurant, Government Building, Lounge, Brewery, Chinese Restaurant, Neighbourhood, Supermarket, Sushi Restaurant, Bookstore, Bistro, Theater, Seafood Restaurant, Brazilian Restaurant, Wine Bar, Convention Center, Gym / Fitness Center, Plaza, Food & Drink Shop, Nightclub;
- Church;
- Soccer Stadium;
- Hotel, Cafè, Pizza Place, Italian Restaurant, Cocktail Bar;
- Multiplex, Art Museum, Department Store, Opera House;

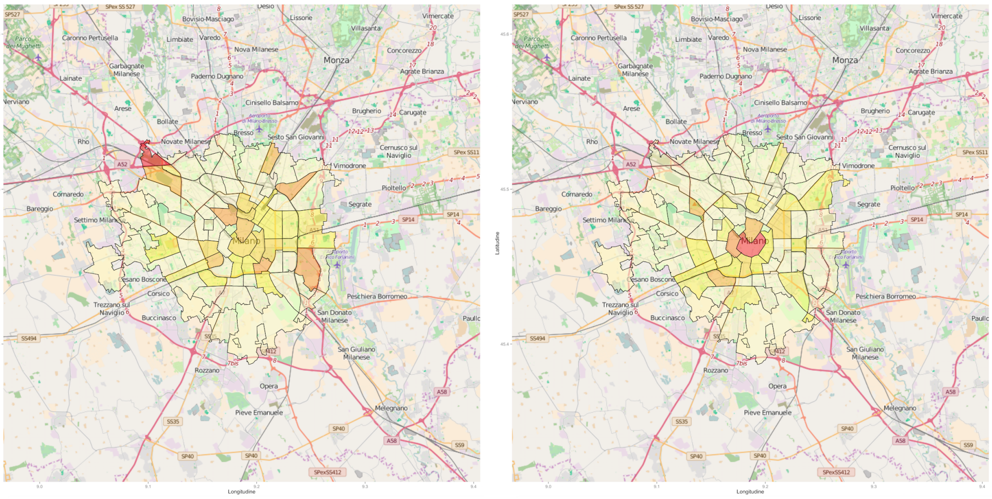
These groups can be a very useful tool to summarise and explain Biclustering results. Therefore, it has been deliberated to consider them with a managerial approach in order to find a reason under the grouping. This analysis led us to manually modify the resulting clusters and name them in the following way:

- Events: Event Space;
- Student Life: University, Bakery, Ice Cream Shop, Monument / Landmark, Clothing Store, Art Gallery, Sandwich Place, Burger Spot, Boutique, Gym;
- Life Style: Japanese Restaurant, Pub, Bar, Restaurant, Government Bulding, Lounge, Brewery, Chinese Restaurant, Neighbourhood, Supermarket, Sushi Restaurant, Bookstore, Bistro, Theater, Seafood Restaurant, Brazilian Restaurant, Wine Bar, Convention Center, Gym / Fitness Center, Plaza, Food & Drink Shop, Nightclub;
- Church: Church;
- Mass Entertainment: Soccer Stadium, Multiplex;
- Tourist Life: Hotel, Cafè, Pizza Place, Italian Restaurant, Cocktail Bar, Art Museum, Department Store, Opera House;

Precisely, the groups called Events and Church are the only two which remained unvaried: Events is composed only by Event Space while Church is composed only by the Church category. The importance of these two clusters will be analysed later but it is principally connected with EXPO2015 and the huge amount of tourists that everyday also takes selfies with the Dome of Milan. One first hint about this fact is given by the heat maps presented in Figure 4.11. The reader will surely notice how the darker NILs, therefore the most representative ones, are respectively Cascina Triulza - EXPO and Duomo.

The biggest cluster has been split into two group: Student Life and Life Style. The first one is characterised by categories, like University, Bakery and Burger Spot, typical referred to students or, generally, young persons. The second one presents principally restaurants, pubs and stores and, for these reasons, it is called Life Style. It is possible to see that the Life Style cluster covers a greater number of NILs than the Student Life one which, instead, is limited to zones popular among younger visitors. Figure 4.12 shows the difference just explained using heat maps.

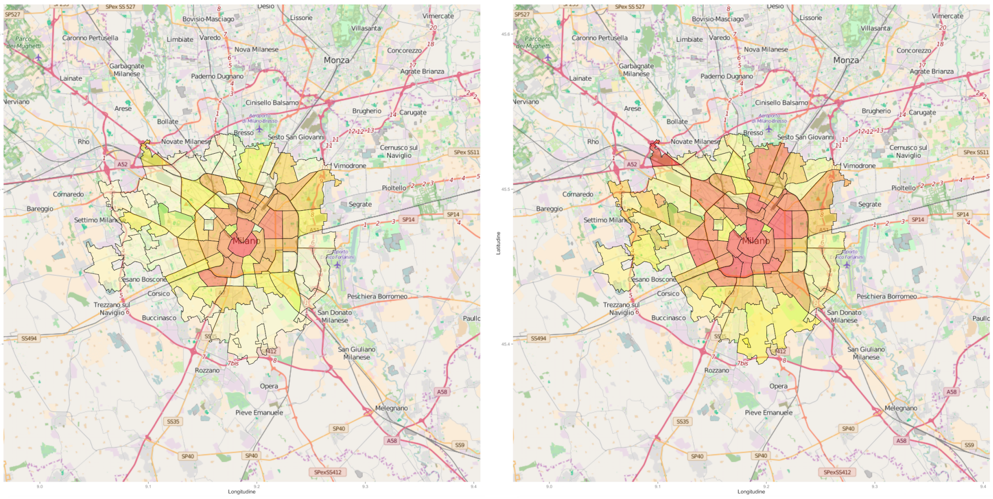
The last two clusters are named Mass Entertainment and Tourist Life. The first one corresponds to the previous group only composed by Soccer Stadium with the addition of the Multiplex category.



(a) Events heat map

(b) Church heat map

Figure 4.11: Events and Church heat maps



(a) Student Life heat map

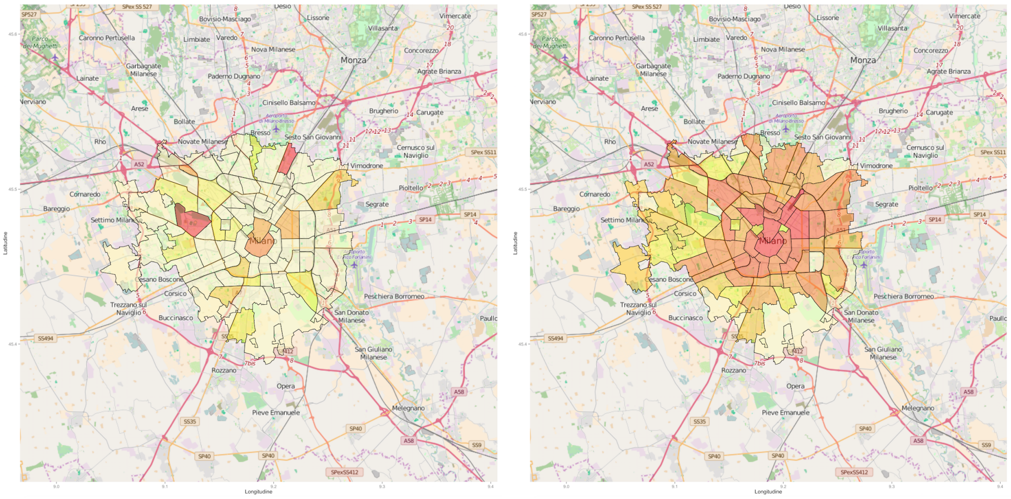
(b) Life Style heat map

Figure 4.12: Student Life and Life Style heat maps

As the reader will learn in the NILs analysis, these two categories are really strong in two precise zones of Milan: San Siro and Bicocca. In fact these two NILs host two important venues: the San Siro soccer stadium and the multiplex UCI Cinemas Bicocca.

Instead, it is possible to notice how the Tourist Life cluster is created by adding together two old clusters (without Multiplex). It has been decided to merge them together because they both present categories very popular in the center of Milan as Figure 4.13 shows.

Thanks to the new categorisation of the remaining 45 categories, it is possible to



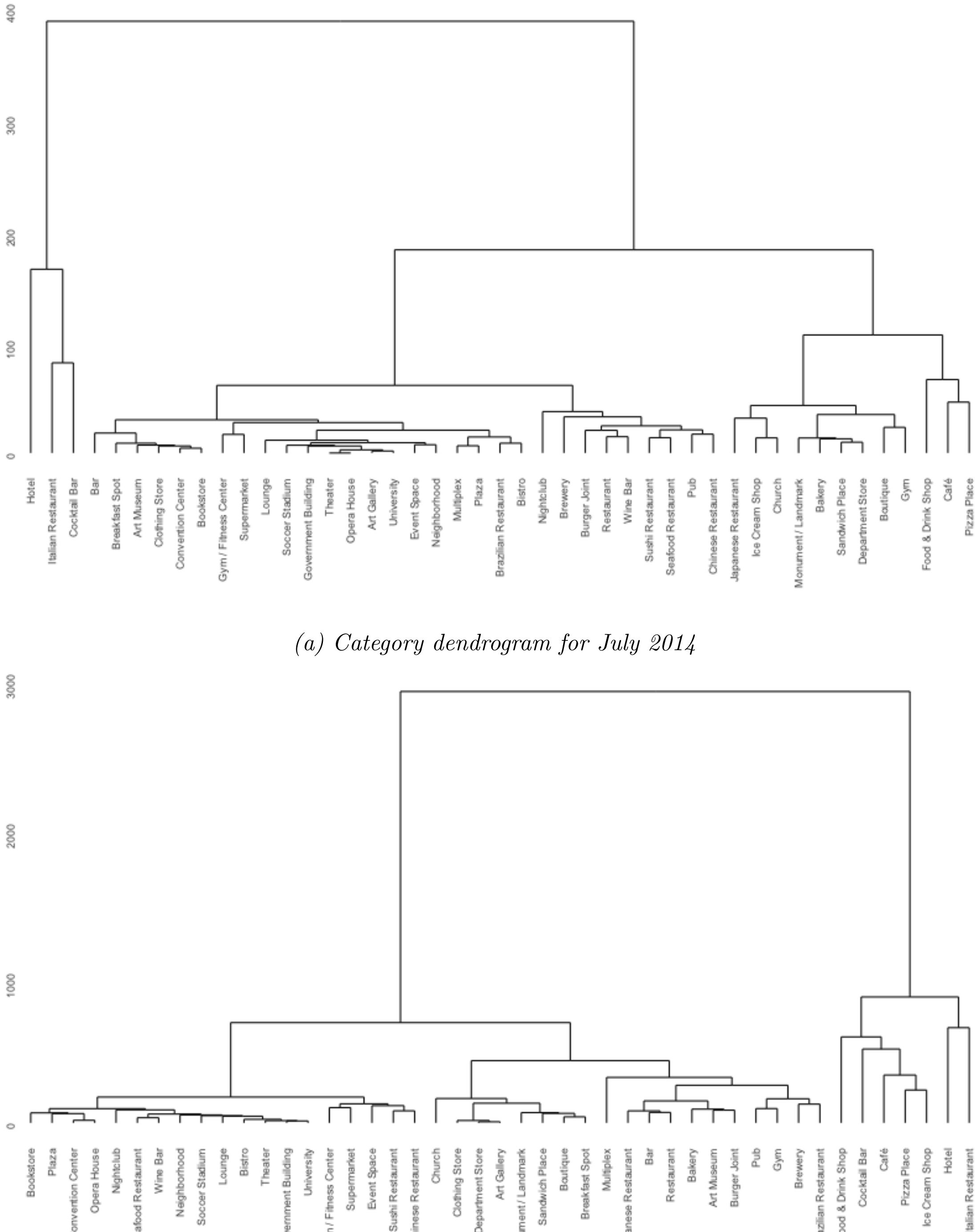
(a) Mass Entertainment heat map *(b) Tourist Life heat map*

Figure 4.13: Mass Entertainment and Tourist Life heat maps

present more polished bicluster views that will surely help the comprehension. In conclusion, the reader will be grateful for this mathematical and managerial grouping: it is a promise.

4.4.2 Describe categories dynamics using the monthly approach

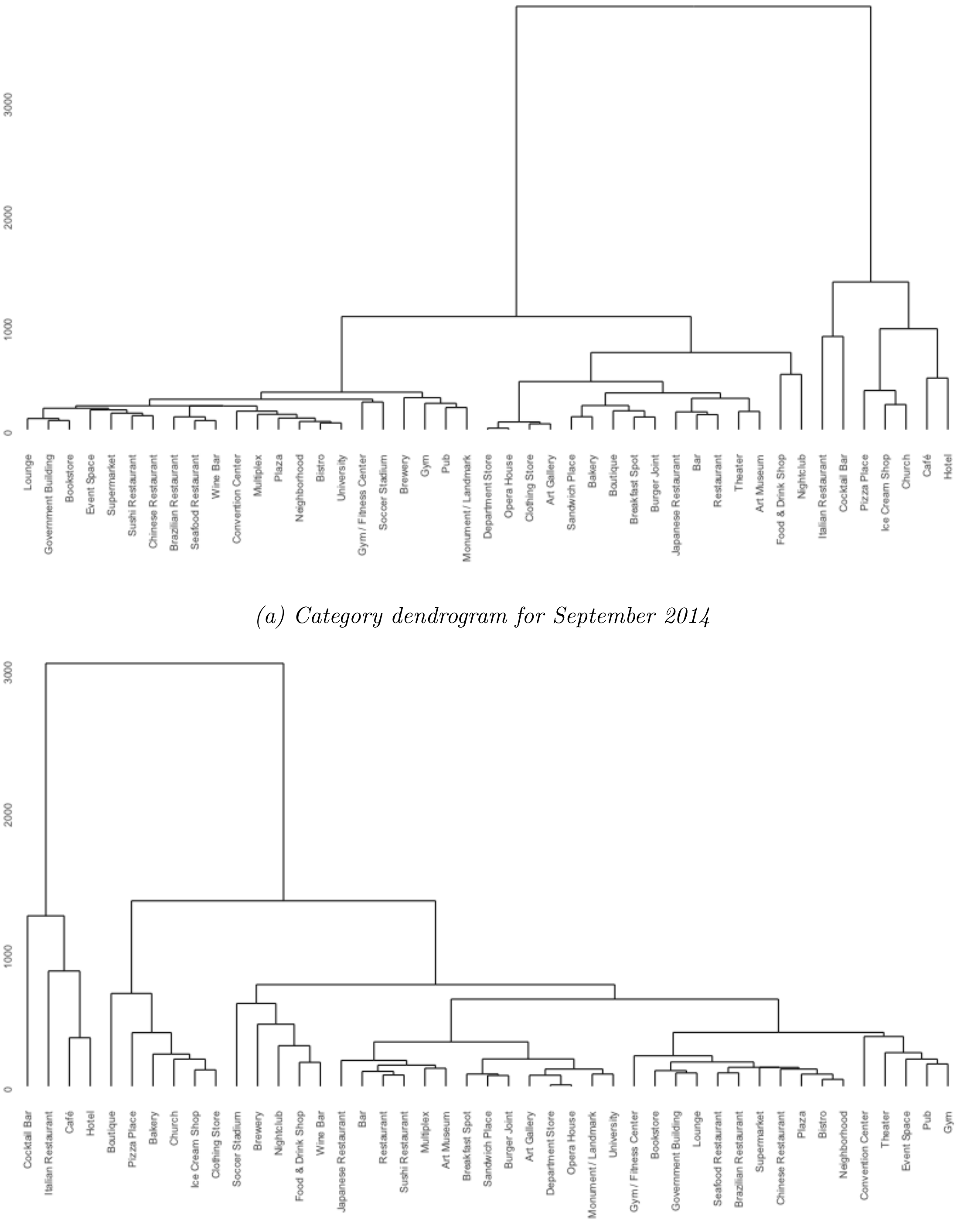
The first and most natural way to describe the 45 categories is given by the monthly approach of grouping. The Clustering procedure is performed on every single month, corresponding to every singular layer of the array 3D. The task is to understand how the interest of Foursquare users in terms of categories change over time. Therefore the task is not to cluster categories but to define their dynamic using clustering. For this reason, it has been decided not to cut the resulting dendrograms but to use them as source of information. Next figures present all the 15 dendrograms. The intelligent reader can easily notice how it is difficult to understand all the monthly changes even if with only 45 categories. However some patterns are quite evident. Generally, Hotel, Café, Italian Restaurant, Pizza Place and Cocktail Bar are near from each other and divided from the rest. Sometimes single categories stand out from the totality. It is the case of Multiplex in December 2015, March 2015 and April 2016. Also Event Space, thanks to the popularity of EXPO2015, is the top category in May 2015 (ex aequo with Opera House), July 2015, August 2015 and September 2015. Therefore June is the only summer month having Church as primal category instead of Event Space.



(a) Category dendrogram for July 2014

(b) Category dendrogram for August 2014

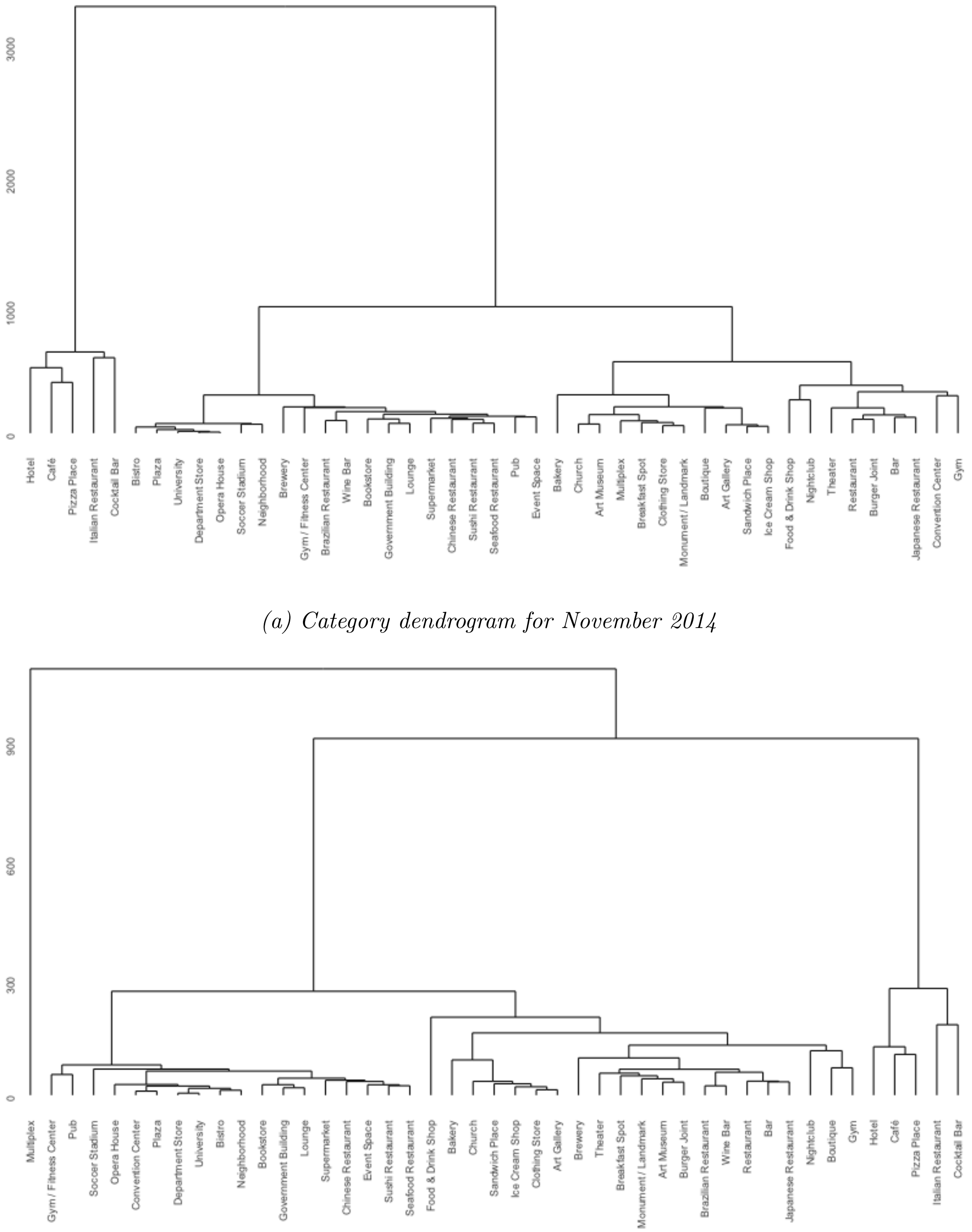
Figure 4.14: Category dendrogram July and August 2014



(a) Category dendrogram for September 2014

(b) Category dendrogram for October 2014

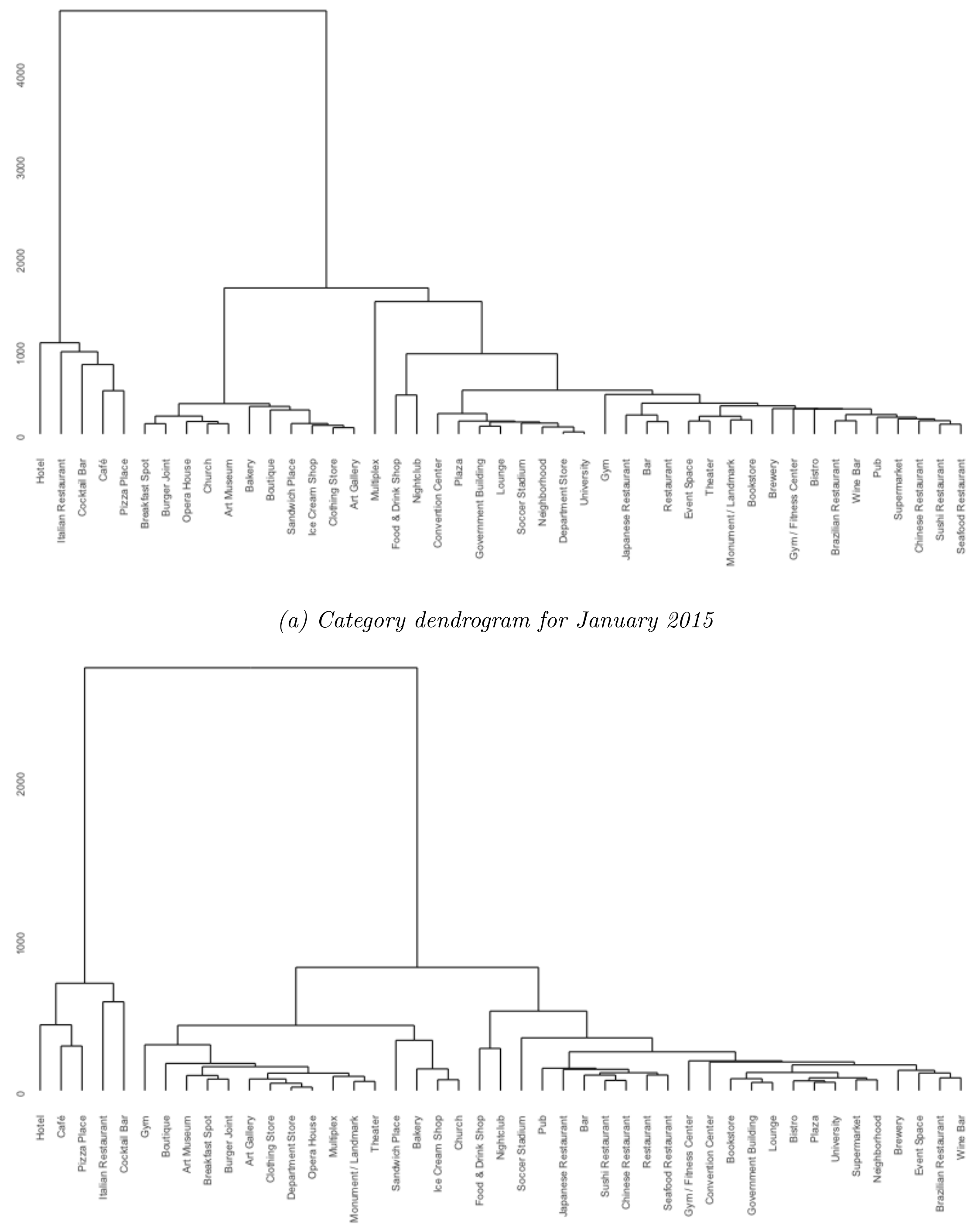
Figure 4.15: Category dendrogram September and October 2014



(a) Category dendrogram for November 2014

(b) Category dendrogram for December 2014

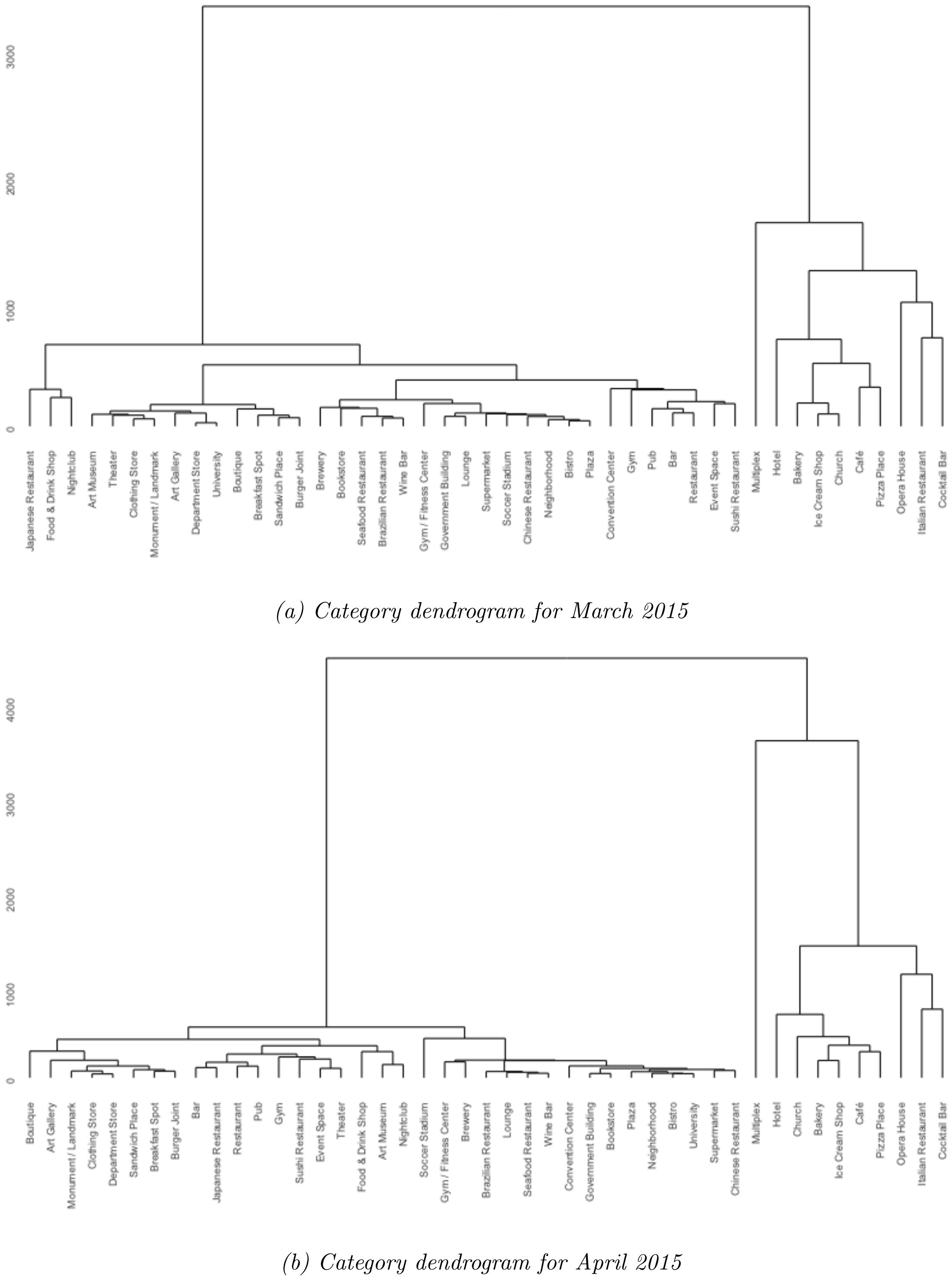
Figure 4.16: Category dendrogram November and December 2014



(a) Category dendrogram for January 2015

(b) Category dendrogram for February 2015

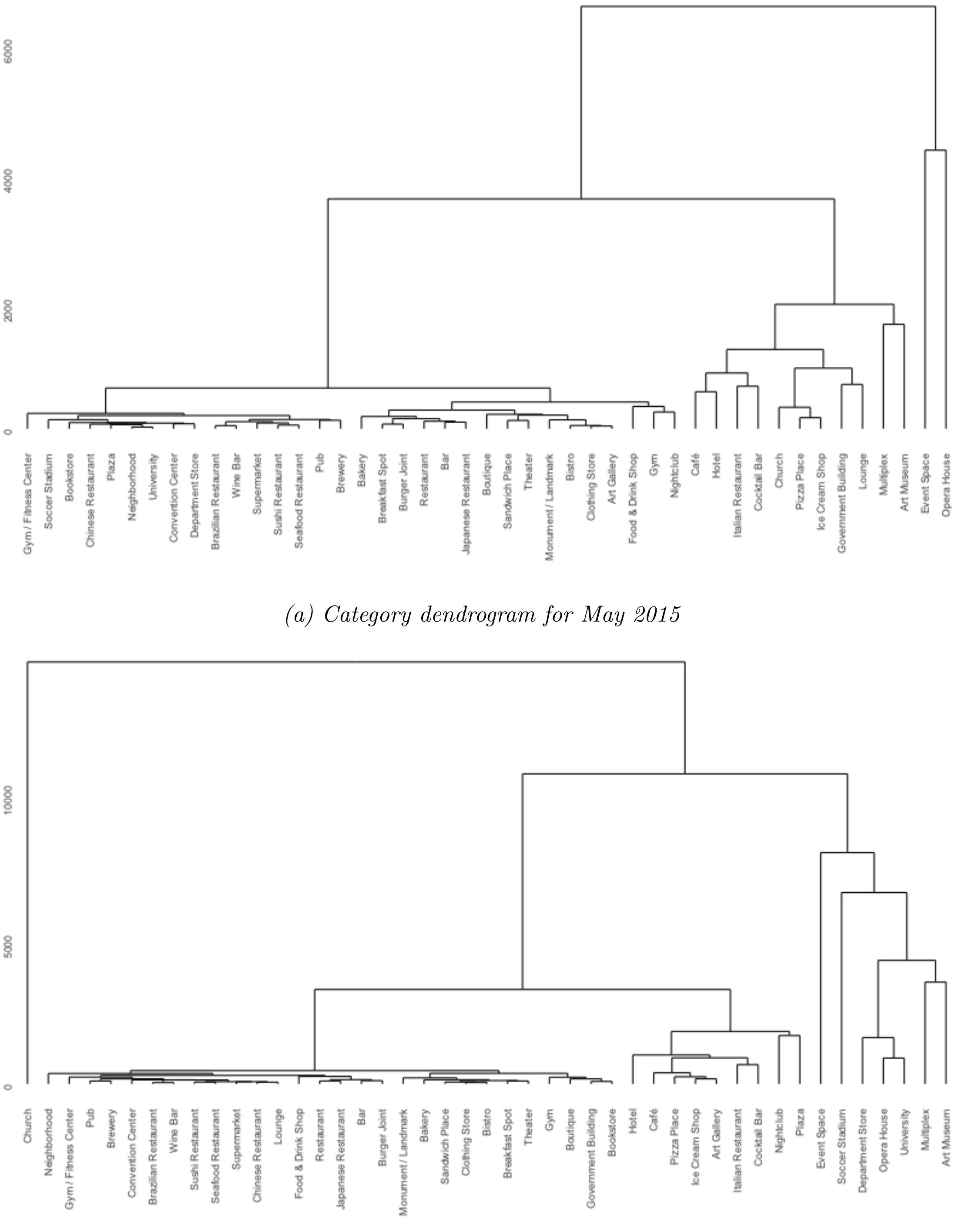
Figure 4.17: Category dendrogram January and February 2015



(a) Category dendrogram for March 2015

(b) Category dendrogram for April 2015

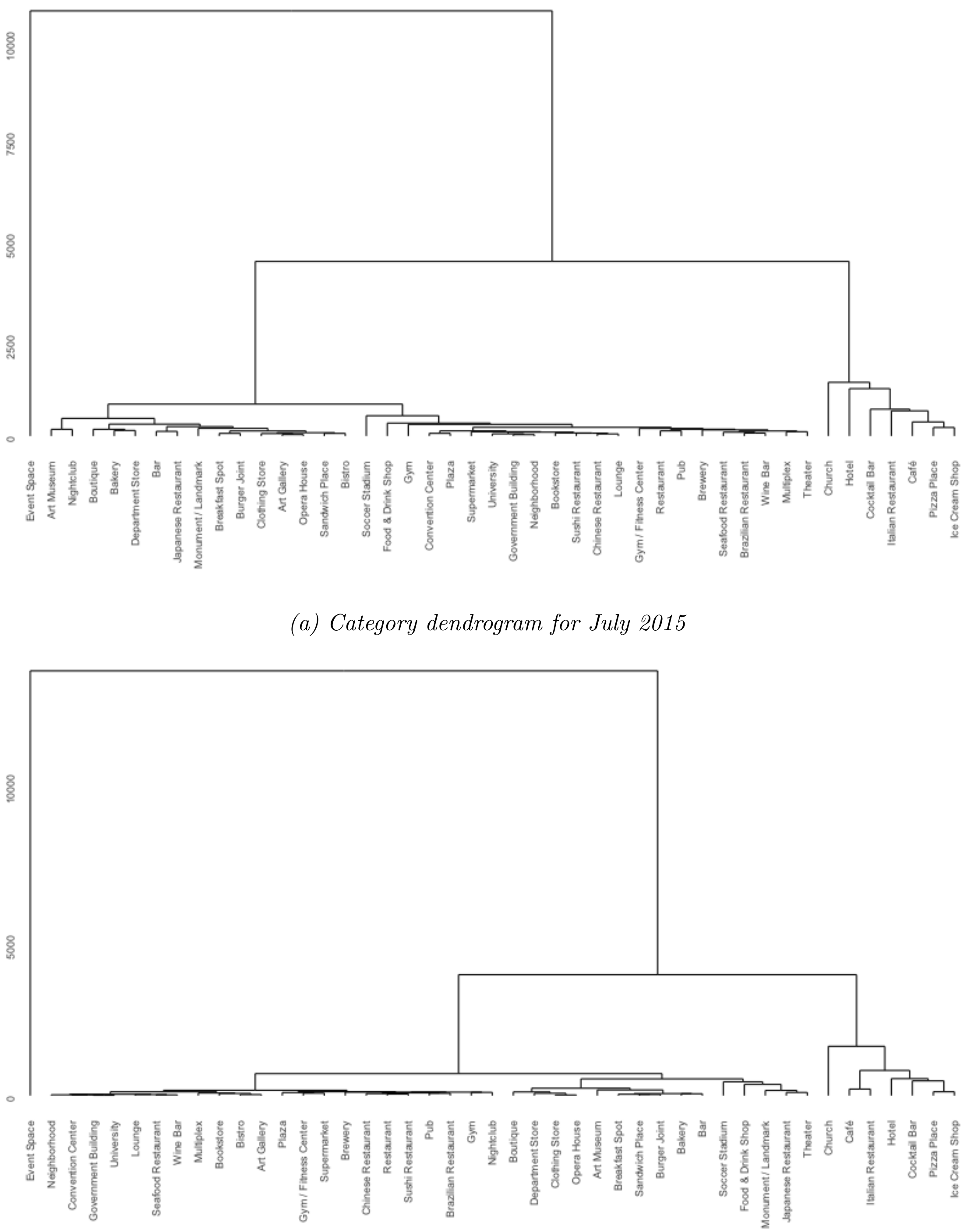
Figure 4.18: Category dendrogram March and April 2015



(a) Category dendrogram for May 2015

(b) Category dendrogram for June 2015

Figure 4.19: Category dendrogram May and June 2015



(a) Category dendrogram for July 2015

(b) Category dendrogram for August 2015

Figure 4.20: Category dendrogram July and August 2014

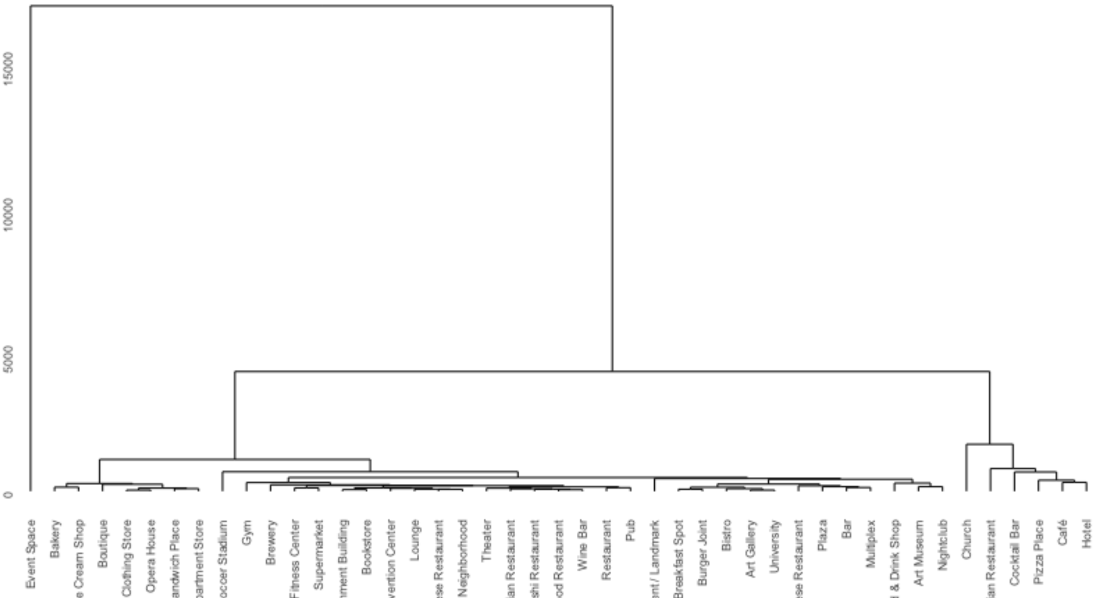


Figure 4.21: Category dendrogram for September 2015

4.5 Clustering the NILs

This section aims at describing everything about the Clustering procedures used to group the NILs composing the city of Milan. In this way, one will be able to understand the similarity patterns that characterises the zones of Milan according to the Foursquare users. In fact, those NILs that are similarly frequented will be clustered together. At this point it is important to remark that every NIL is taken in its own integrity. Thus, all the 45 categories are considered at the same time.

As previously said, the procedure is based on a hierarchical algorithm with Euclidean's distance and Ward's method and it returns a dendrogram which can be studied to extract the clusters. In the case of NILs, thanks to their geo-referenced nature, it has been decided to plot the resulting clusters on the map of Milan: NILs that belongs to the same group will be coloured with the same hue. This double representation is a luxury if compared with the overpopulated dendrograms presented in the categories section. It is, in fact, instantaneous to understand and so pleasant for the reader's eyes.

In order to take into account every minutia, it has been decided to perform and display all the three approaches previously introduced: the monthly approach, the vectorial approach and the summed approach. In this way the reader will have a description as comprehensive as possible.

It is important to remark that the Clustering procedures here presented are based on the scalar number of check-ins. This is the most natural thing to do, considering how data interpretation is grounded in reality. However many other approaches are possible. One is a compositional one that considers every NILs as composed by different percentages of categories. For example, Duomo is characterized by a 20% of Restaurant, 30% of Church and so on. Unfortunately, due to the sparsity of the data matrix, this kind of approach, even if reasonable, is really problematic. It is also possible to consider the categories grouped in the six clusters in Section 4.4 in order to identify how the center of Milan remains clustered in an unvaried way through time.

4.5.1 Monthly approach

The first and most natural way to describe the differences among the zones of Milan is to use a monthly description of NILs. The Clustering procedure is performed on every single month, corresponding to a singular layer of the array 3D. The task is to understand how Milan and its NILs, after the grouping, change with time. Therefore, it has been decided to show both dendrograms and maps. However, due to the fact that it is impossible to show the maps referring to every choice of dendrogram cut, it has been chosen in order to identify six clusters for every month. On the other hand, the choice of 6 clusters is a nice trade-off between months with more reasonable groups and those with less groups. In order to enjoy this colourful subsection the reader is invited to compare the results with the categories analysis.

In these figures it is possible to notice that a great amount of NILs are grouped together in a single big cluster, usually the black one that, starting from May 2015, changes its colour in red. It is a cluster that principally identifies the peripheral area of Milan and it counts the majority of NILs. On the contrary, the center of the city is characterized by many clusters. However, one can notice that some NILs usually appears as cluster of their own, i.e. a cluster made only by a single element. This is the case of the evergreen NILs of Duomo (in every month December excluded), Brera (in Figures 4.23-4.26) and Navigli (in Figures 4.25-4.27 and in Figure4.29) but also Bicocca and San Siro. These two last NILs, even if outside the center of Milan, do not belong to the peripheral cluster thanks to the presence of two attractive venues: the Multiplex UCI Cinemas Bicocca for Bicocca and the San Siro stadium for San Siro. Therefore, Bicocca stands out as a cluster of its own in December 2014, January 2015, March 2015, April 2015 and May 2015, thanks to the cinematographic season and, probably, to some discounts proposed to whoever checked himself in the cinema. Let us remark that the NIL of Bicocca is also characterized by the University of Bicocca which, however, is not as attractive as the multiplex. A similar analysis can be made for San Siro: it stands out as a cluster of its own in June 2015 and September 2015, thanks to many important concerts (Tiziano Ferro and Jovanotti for instance) which took place in the stadium.

Let us remark that the evolution of the city is deeply connected with the events it hosts. Everyone will love to visit a NIL which proposes attractive categories. For this reason the categories description overlaps with the one done on NILs. The major example is, for sure, the NIL of Cascina Triulza - Expo, famous to be the set of EXPO2015. Therefore, starting from May 2015 to September 2015, it defines a cluster of its own, being one of the most visited zone of the whole city. This is an extraordinary case of how an event can modify a particular zone profile. In fact, Cascina Triulza - Expo was one of the NILs with the lowest number of clusters until the beginning of the globally renowned exposition.

It is important to notice that the dynamic of Milan is also interwoven with seasonality. Parco Sempione, for instance, is one of the most visited NILs in May 2015 thanks to high temperatures, concerts and “EXPO in Città” events. However, it is possible to notice that Parco Sempione, differently from the other parks, is not usually considered a less attractive neighbourhood because of its nightclubs.

Generally, the dynamic of the city is really variegated and it is hard to identify a stationary group, even if it is evident how the center of Milan is more attractive than the rest. However, using the six main category clusters it is possible to verify how some NILs are constantly grouped into the same clusters.

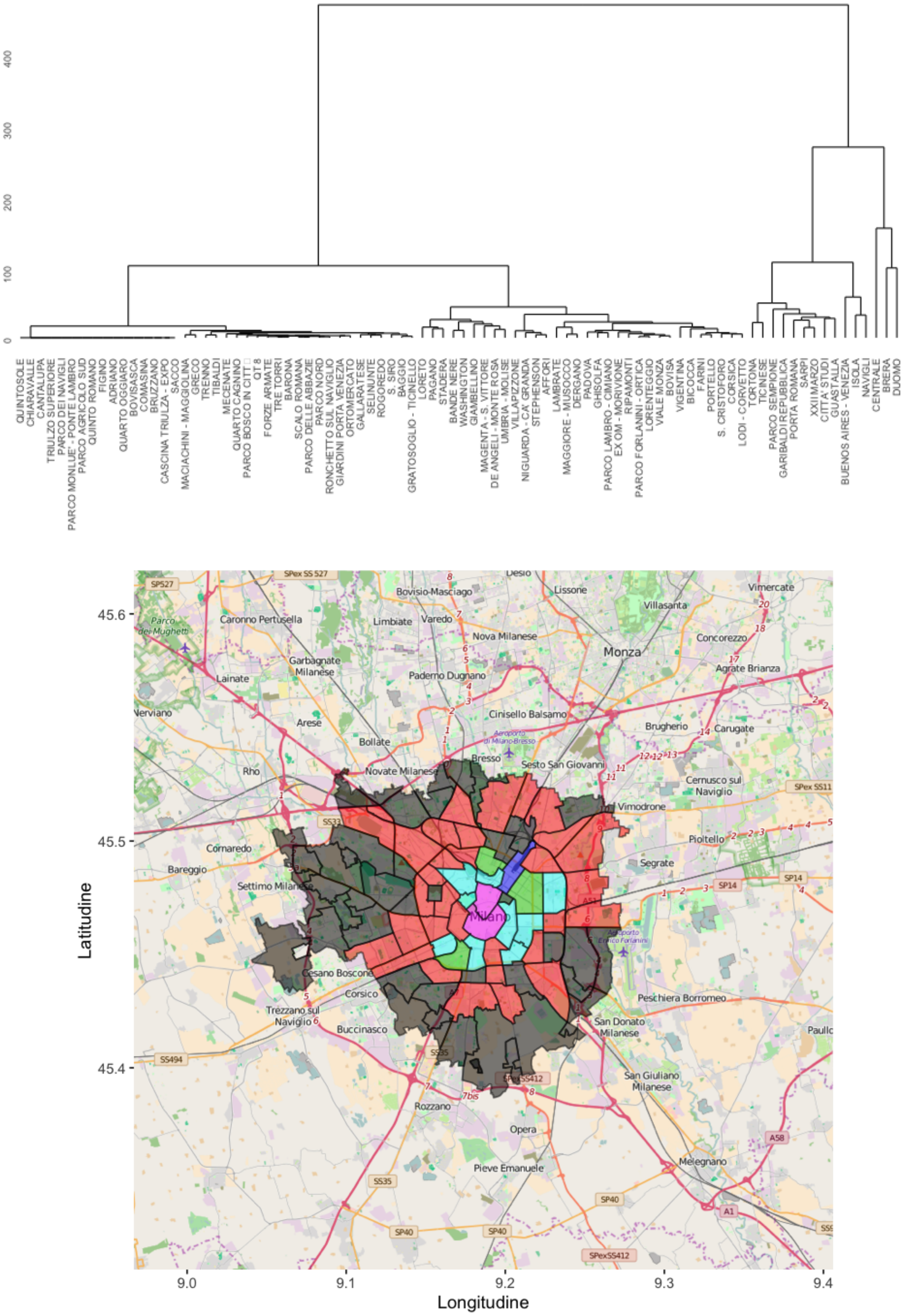


Figure 4.22: NILs clustering for the month of July 2014

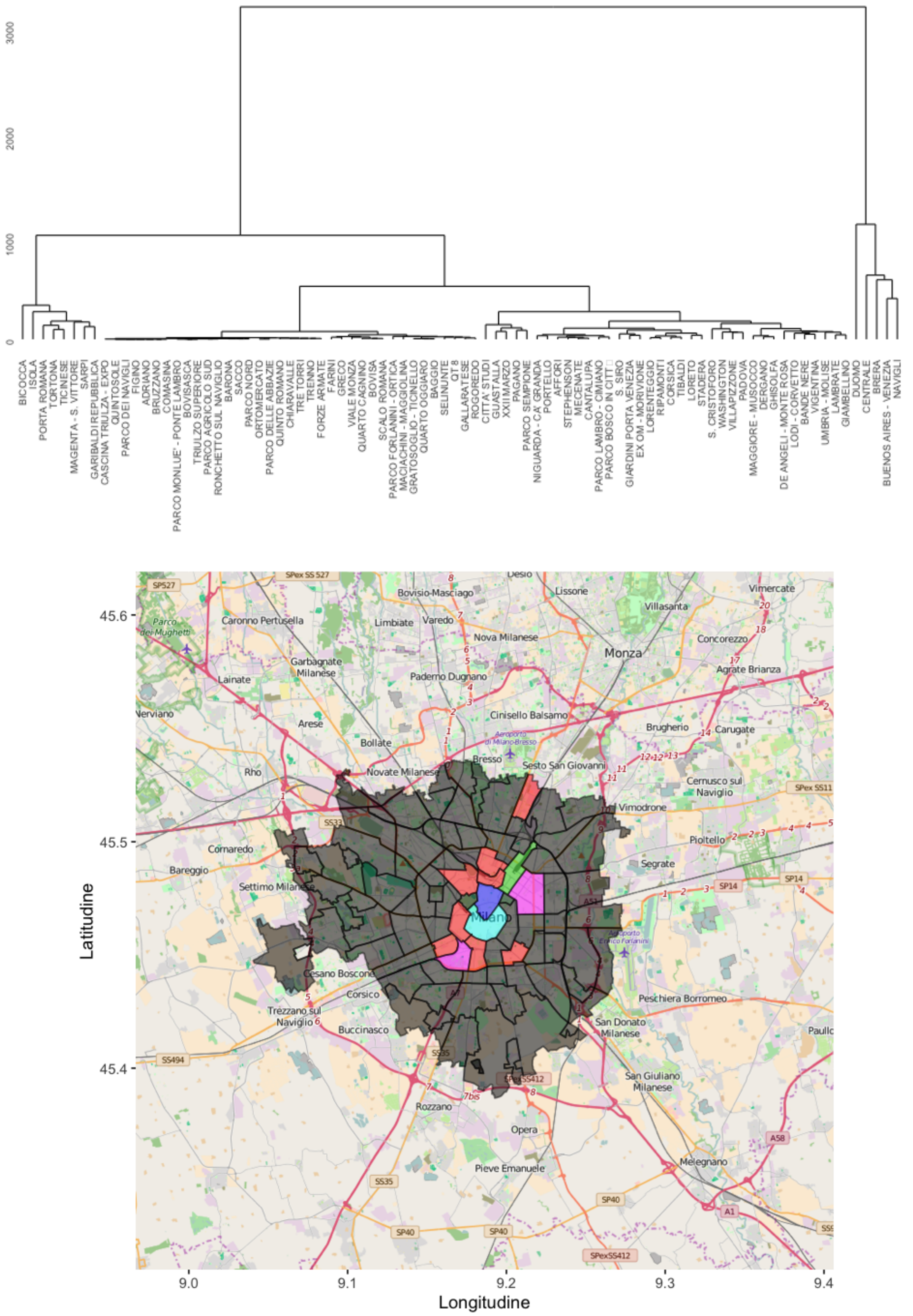


Figure 4.23: NILs clustering for the month of August 2014

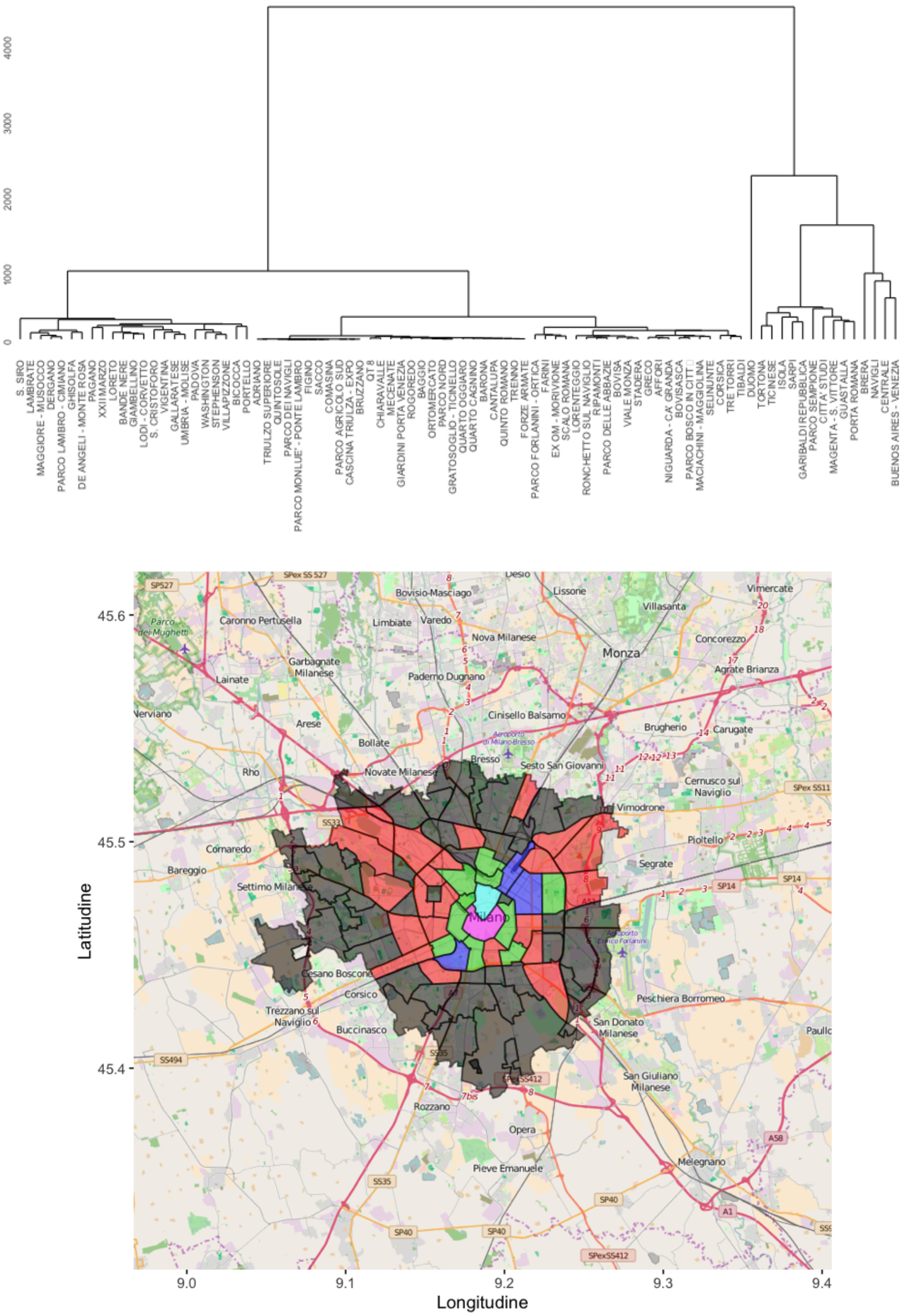


Figure 4.24: NILs clustering for the month of September 2014

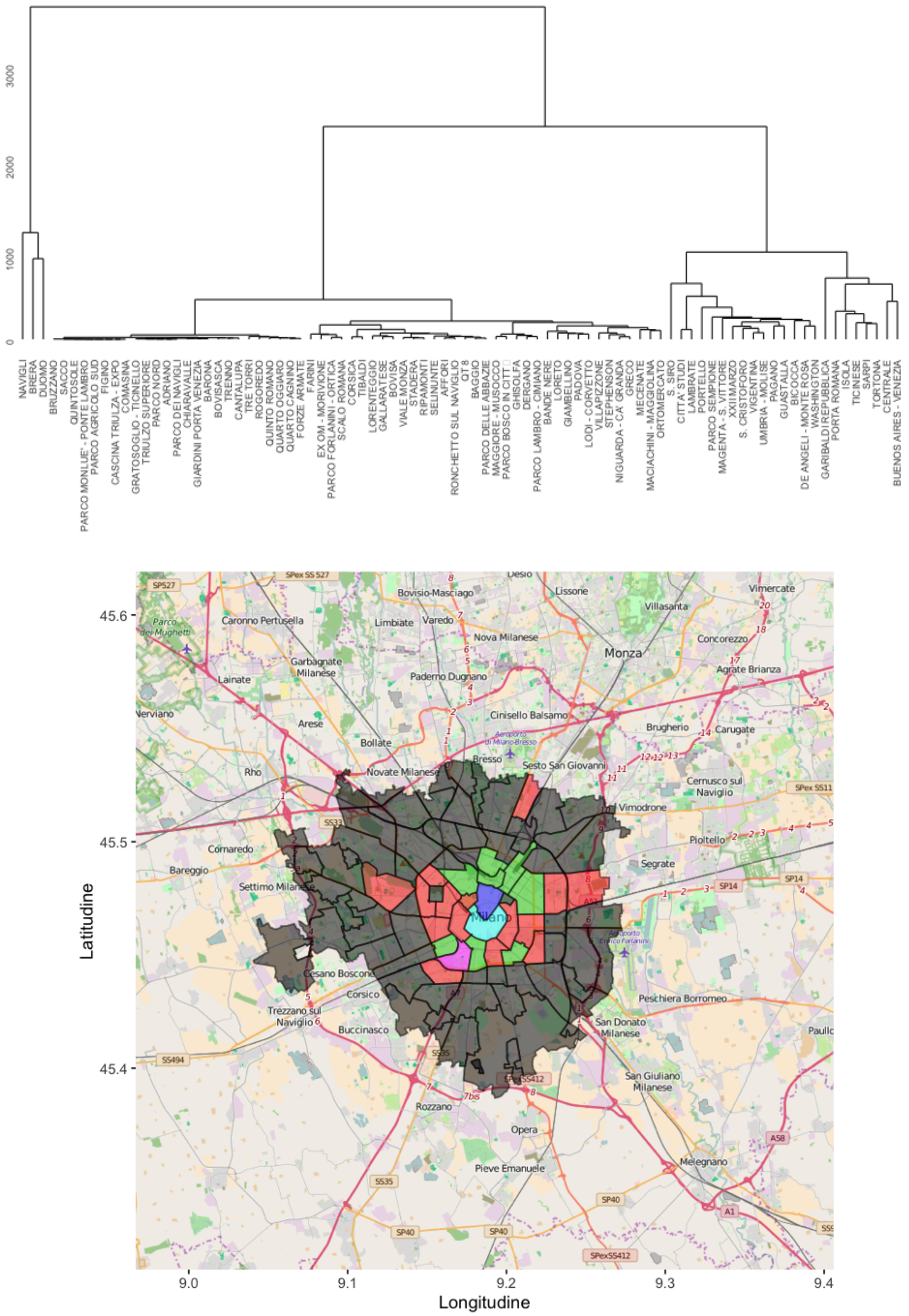


Figure 4.25: NILs clustering for the month of October 2014

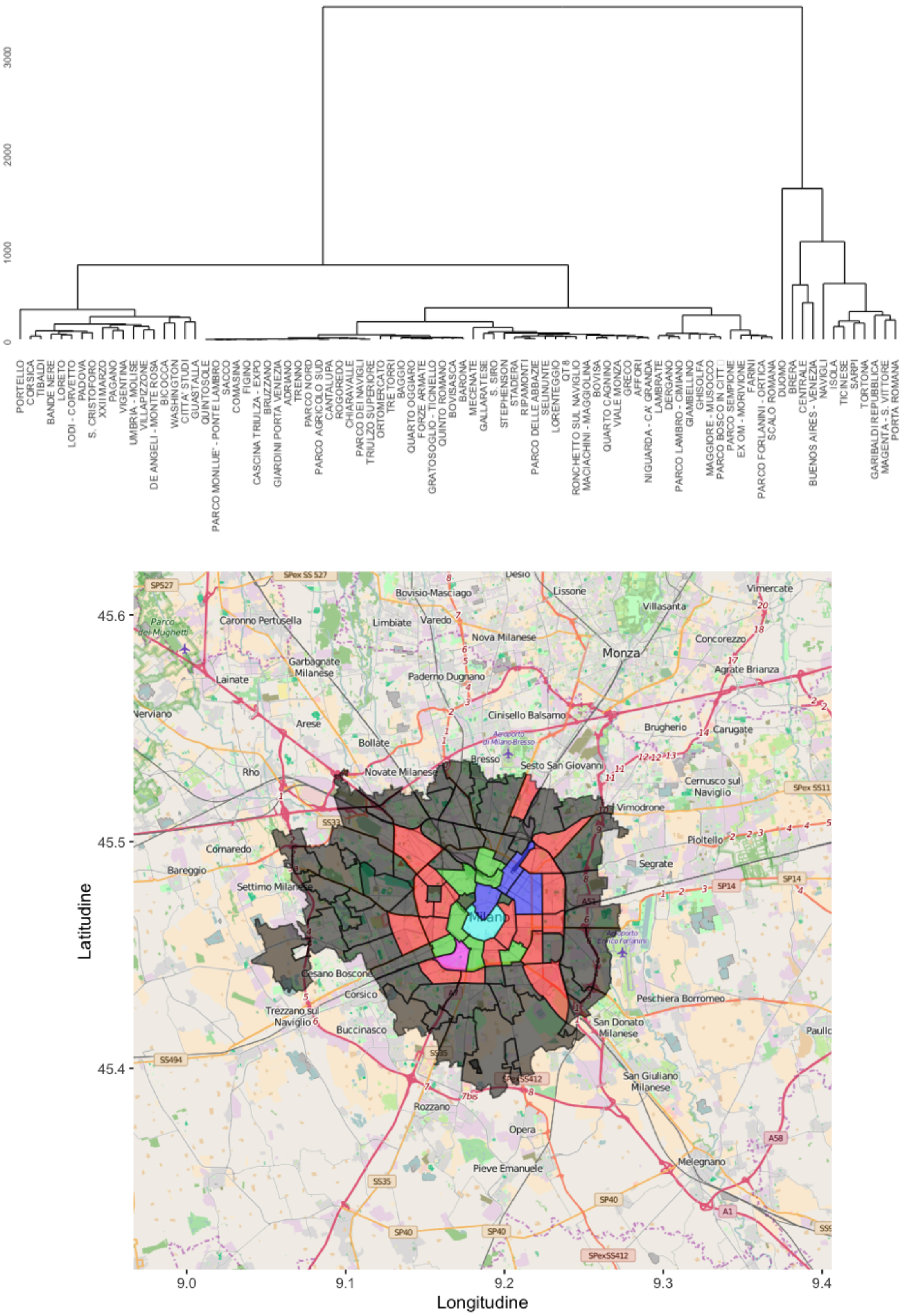


Figure 4.26: NILs clustering for the month of November 2014

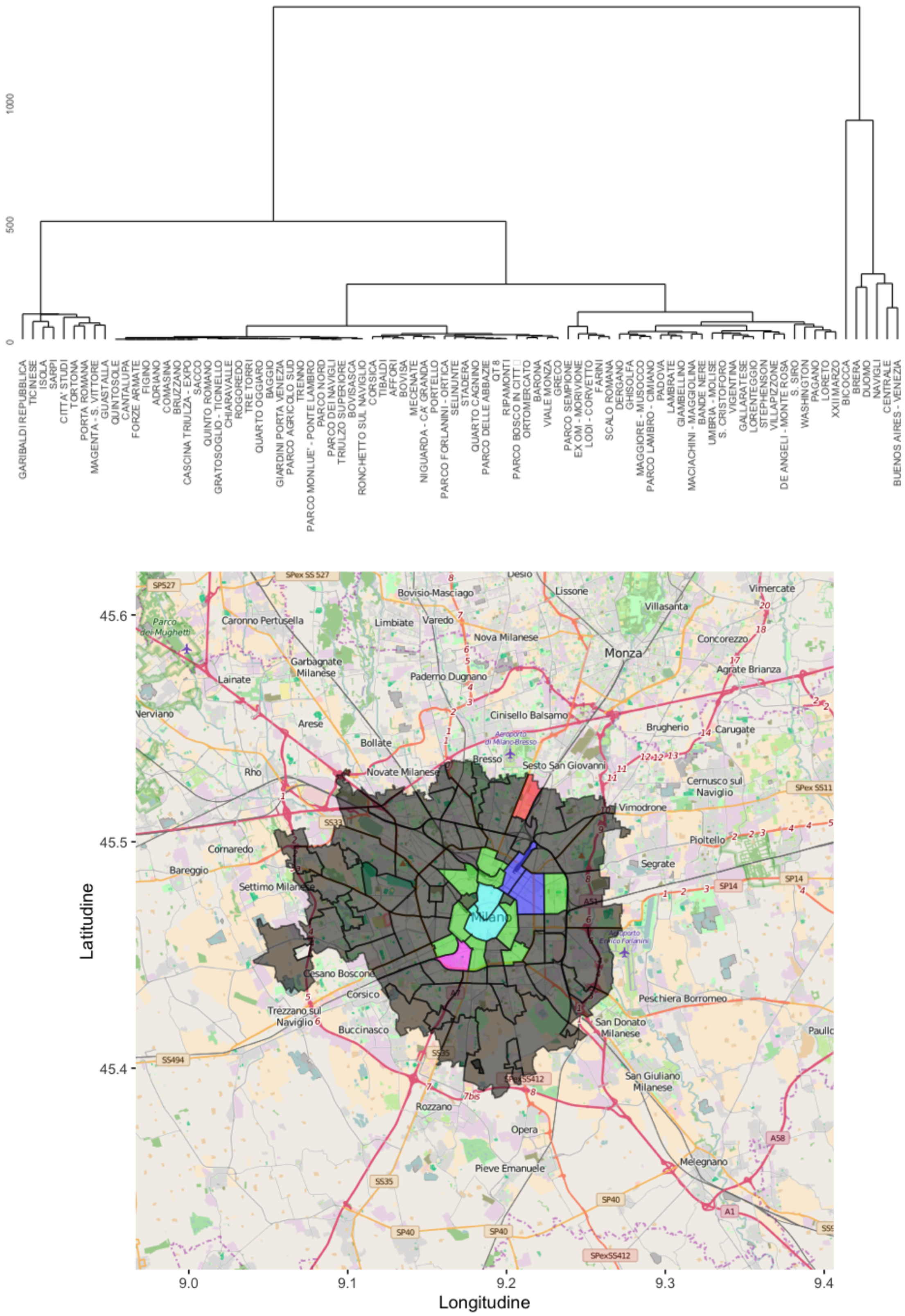


Figure 4.27: NILs clustering for the month of December 2014

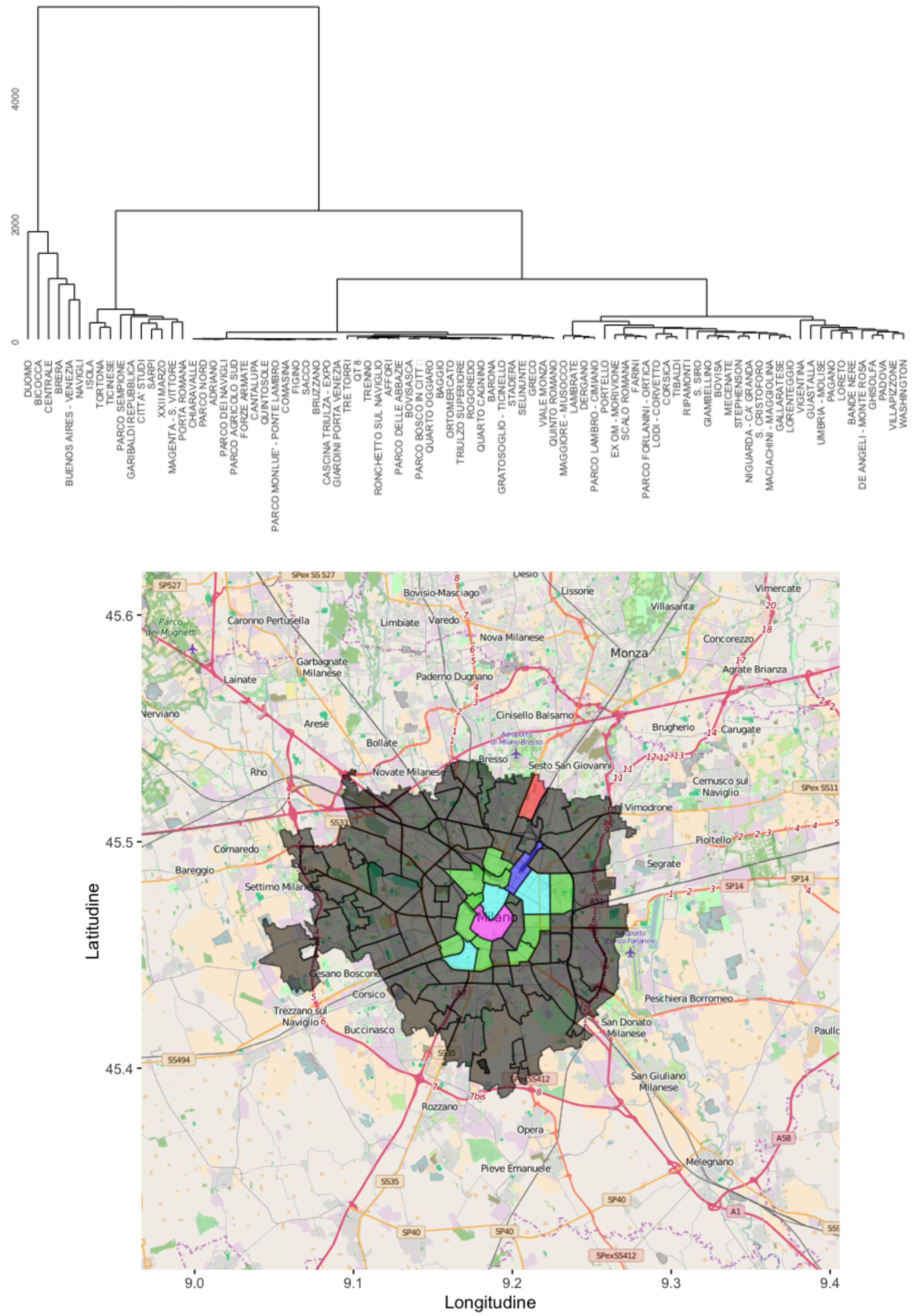


Figure 4.28: NILs clustering for the month of January 2015

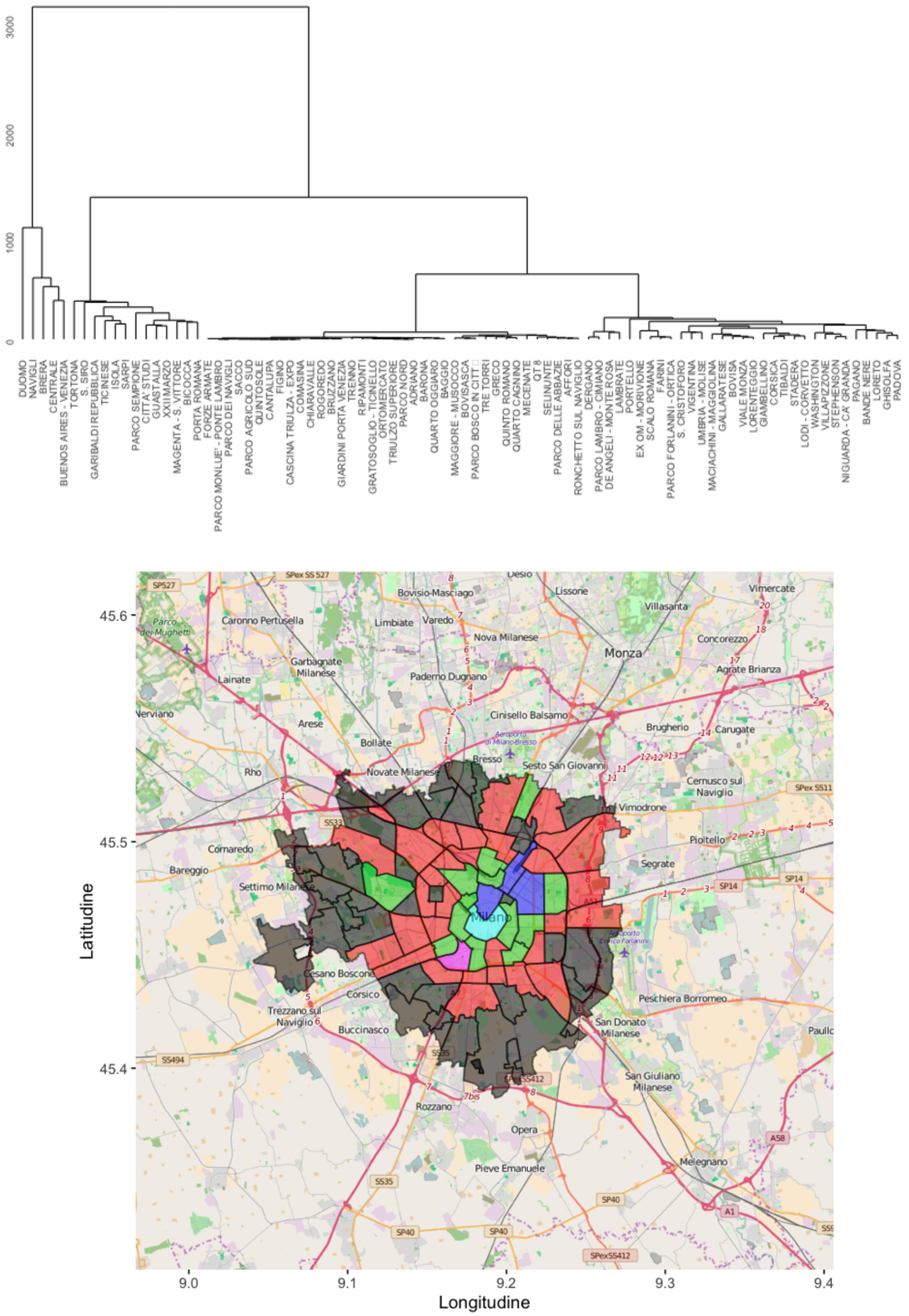


Figure 4.29: NILs clustering for the month of February 2015

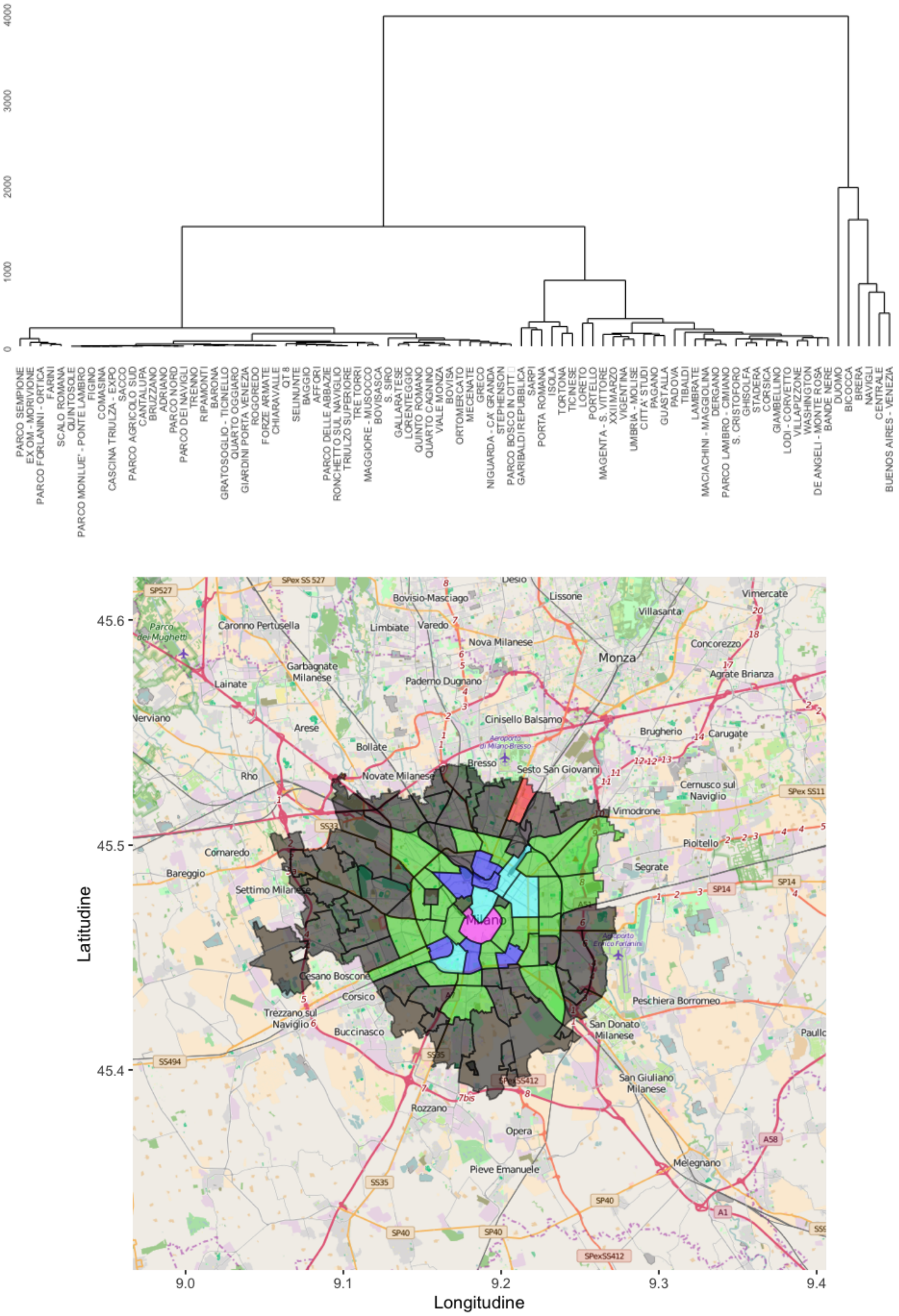


Figure 4.30: NILs clustering for the month of March 2015

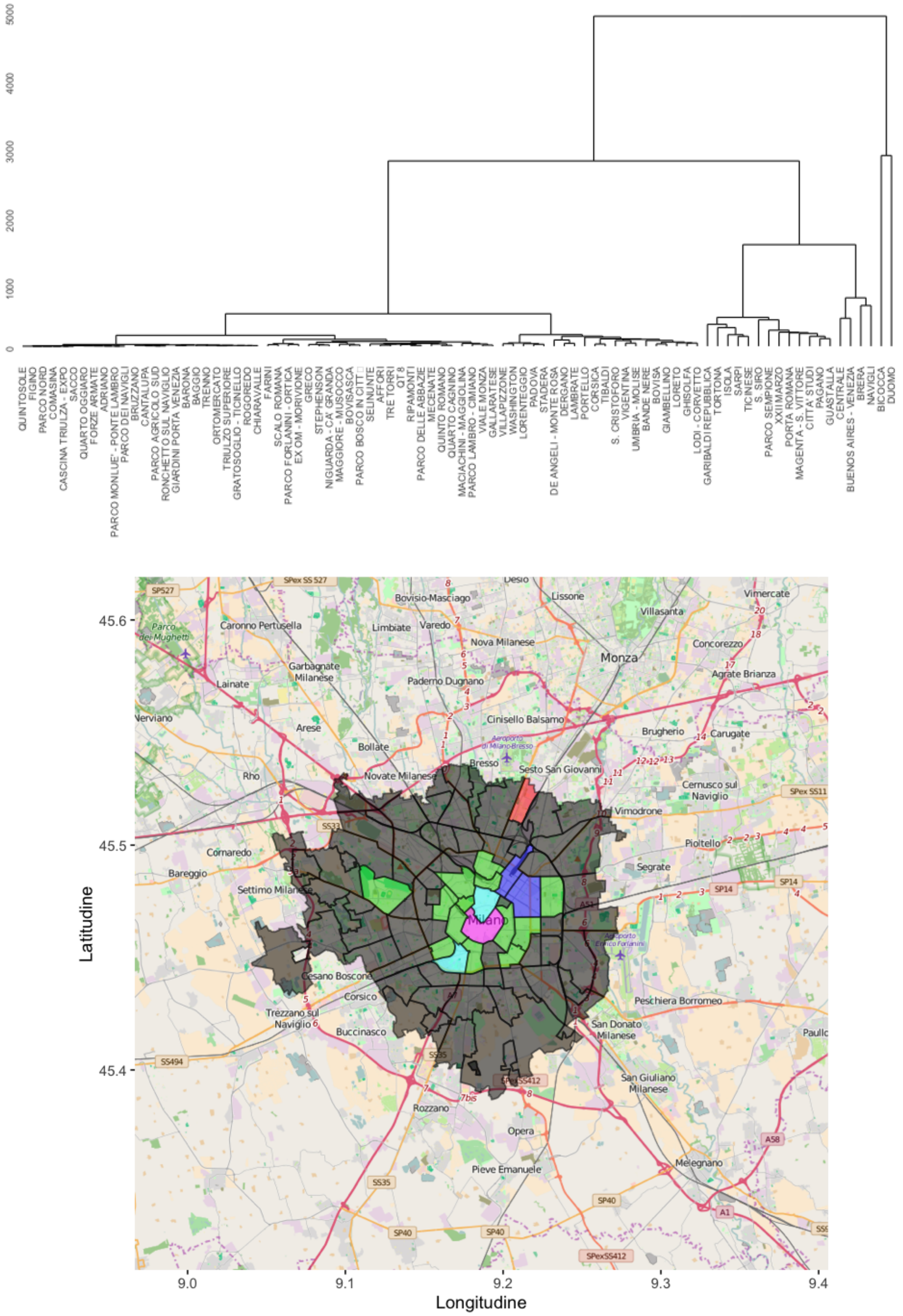


Figure 4.31: NILs clustering for the month of April 2015

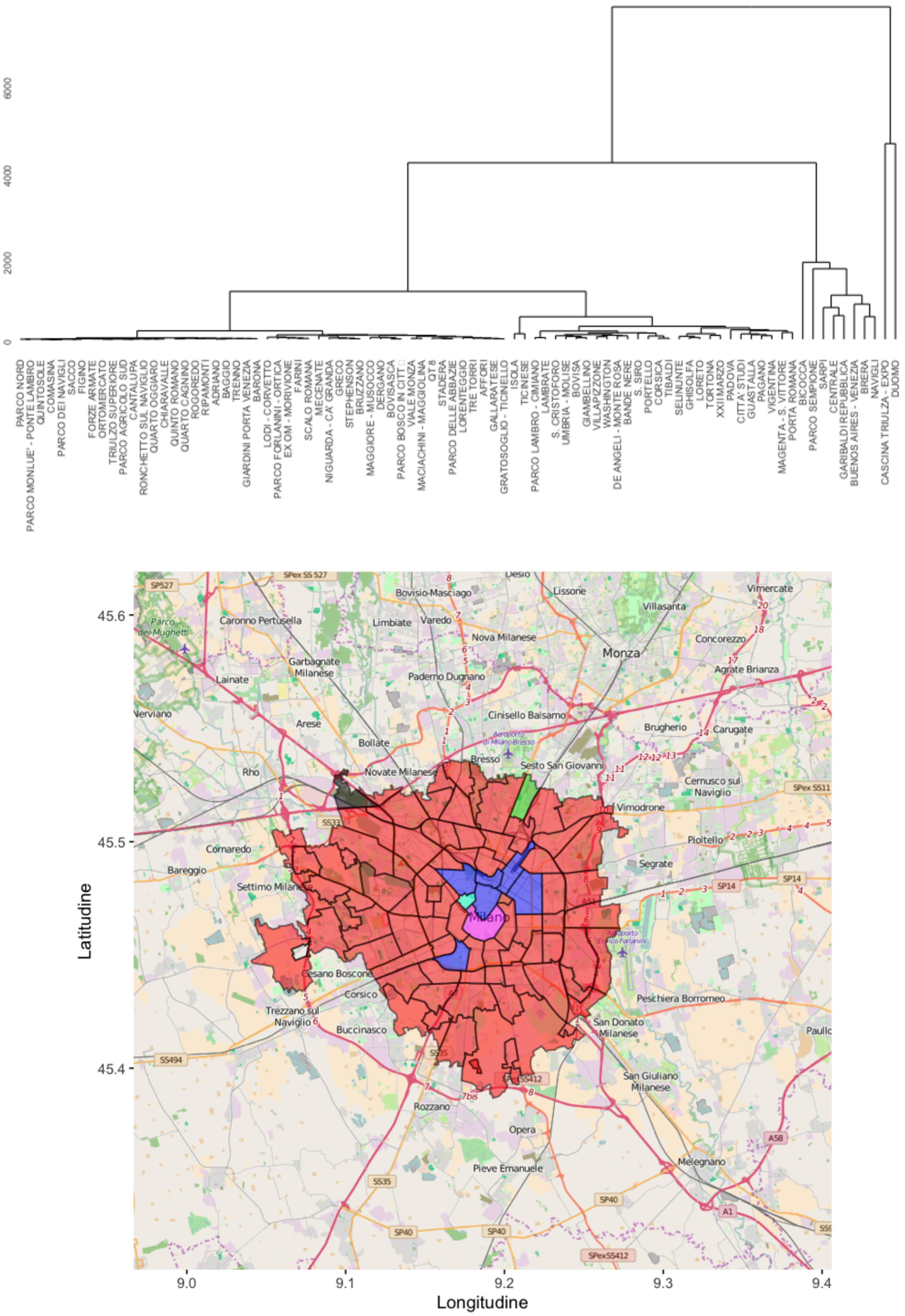


Figure 4.32: NILs clustering for the month of May 2015

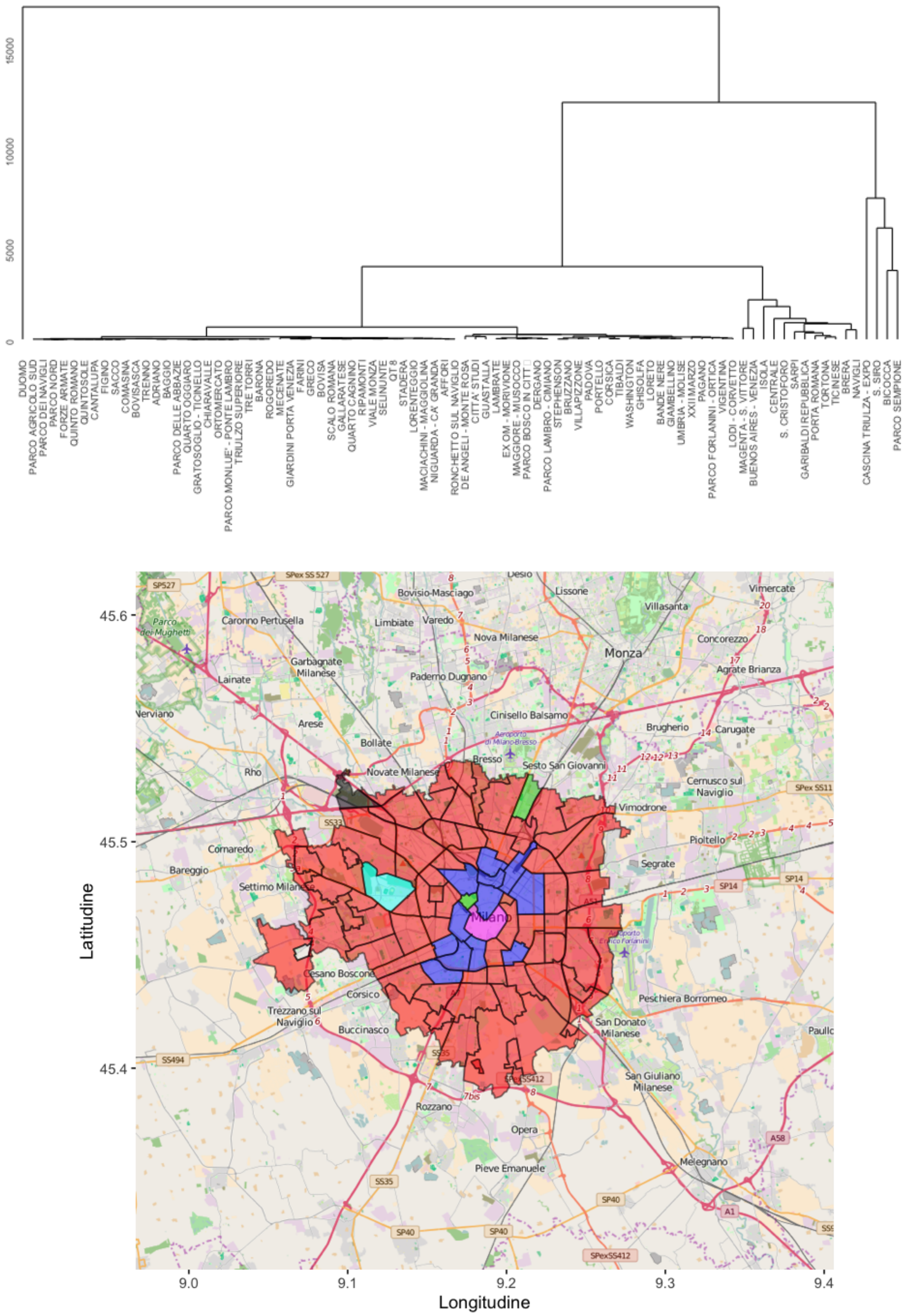


Figure 4.33: NILs clustering for the month of June 2015

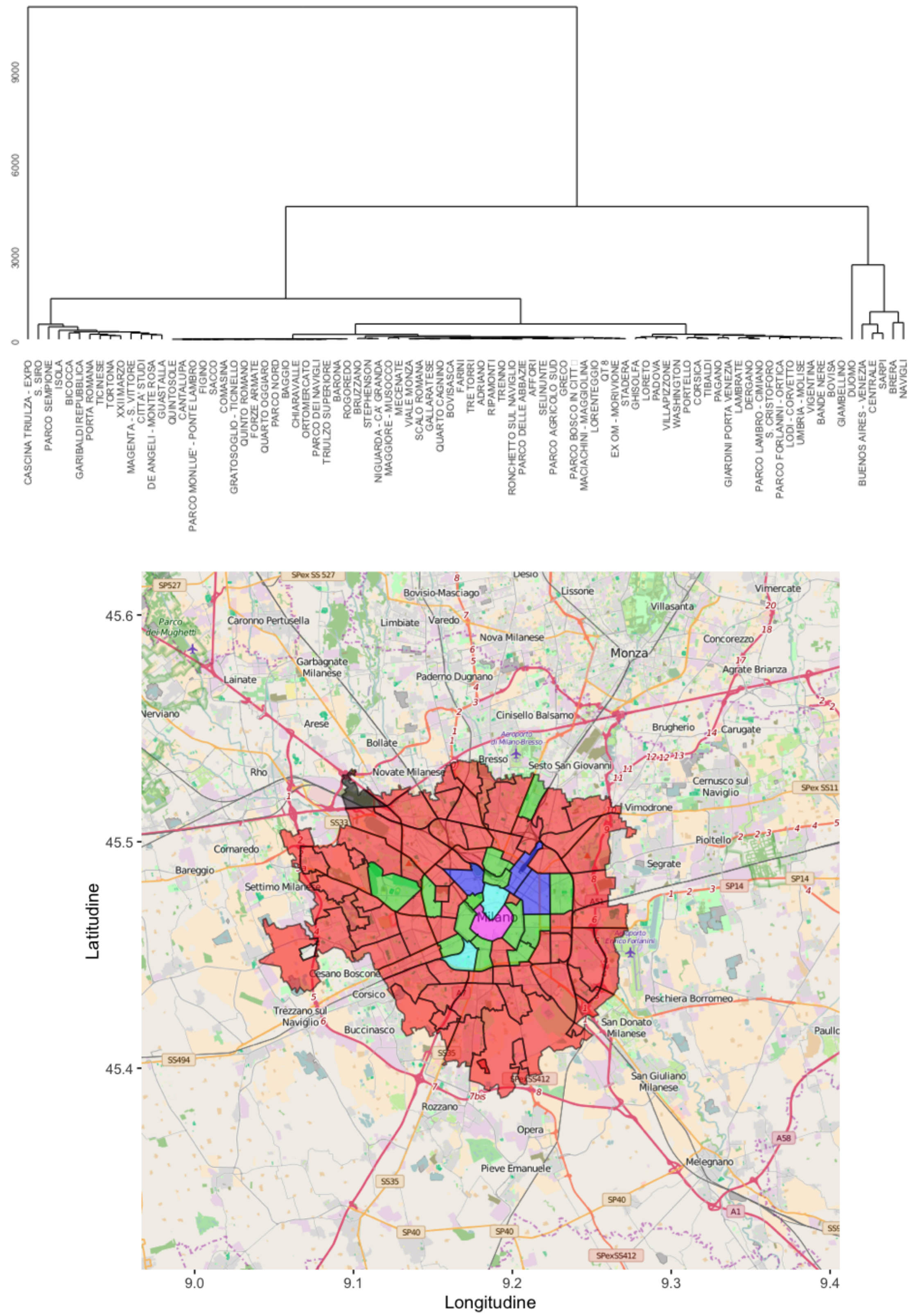


Figure 4.34: NILs clustering for the month of July 2015

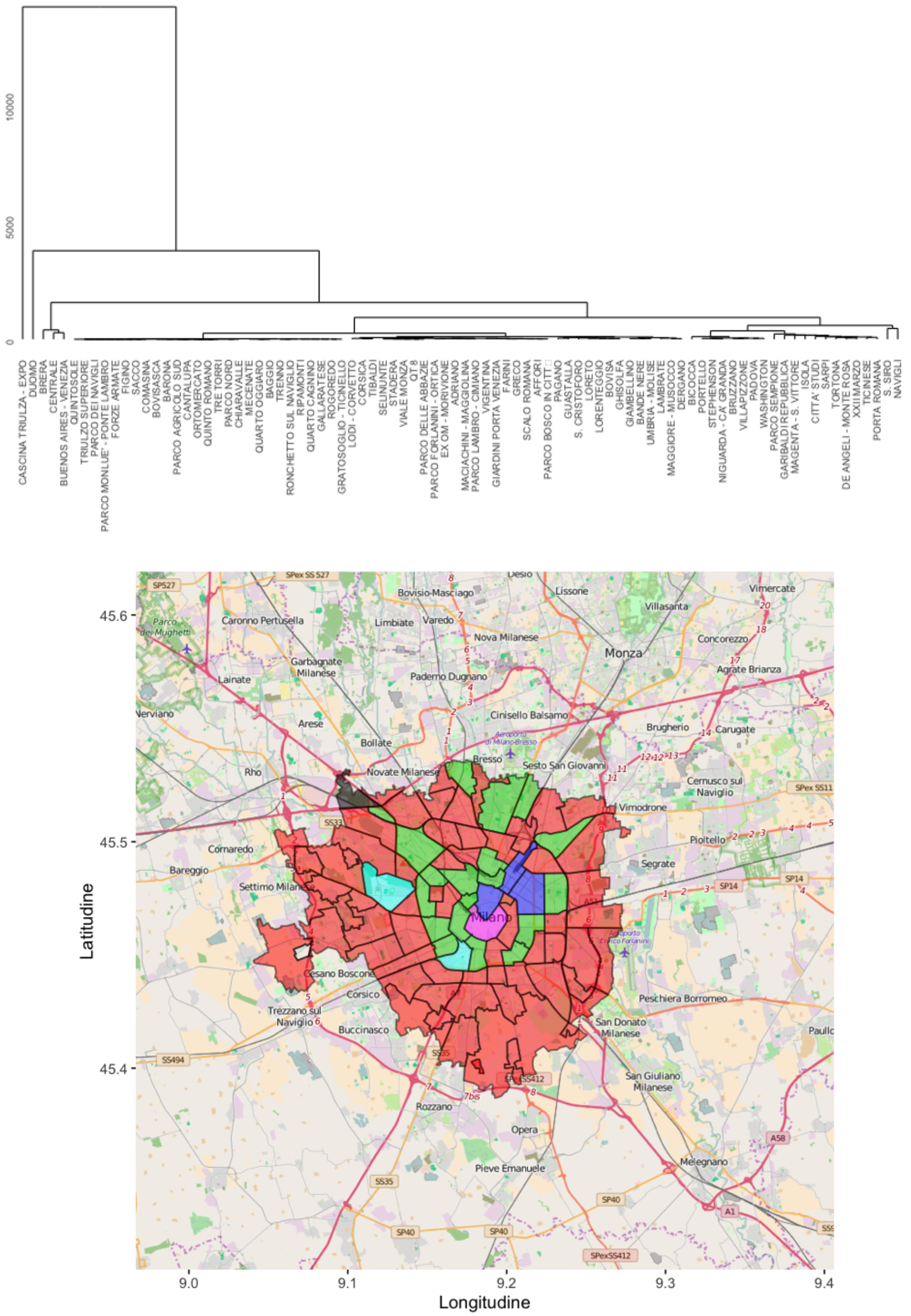


Figure 4.35: NILs clustering for the month of August 2015

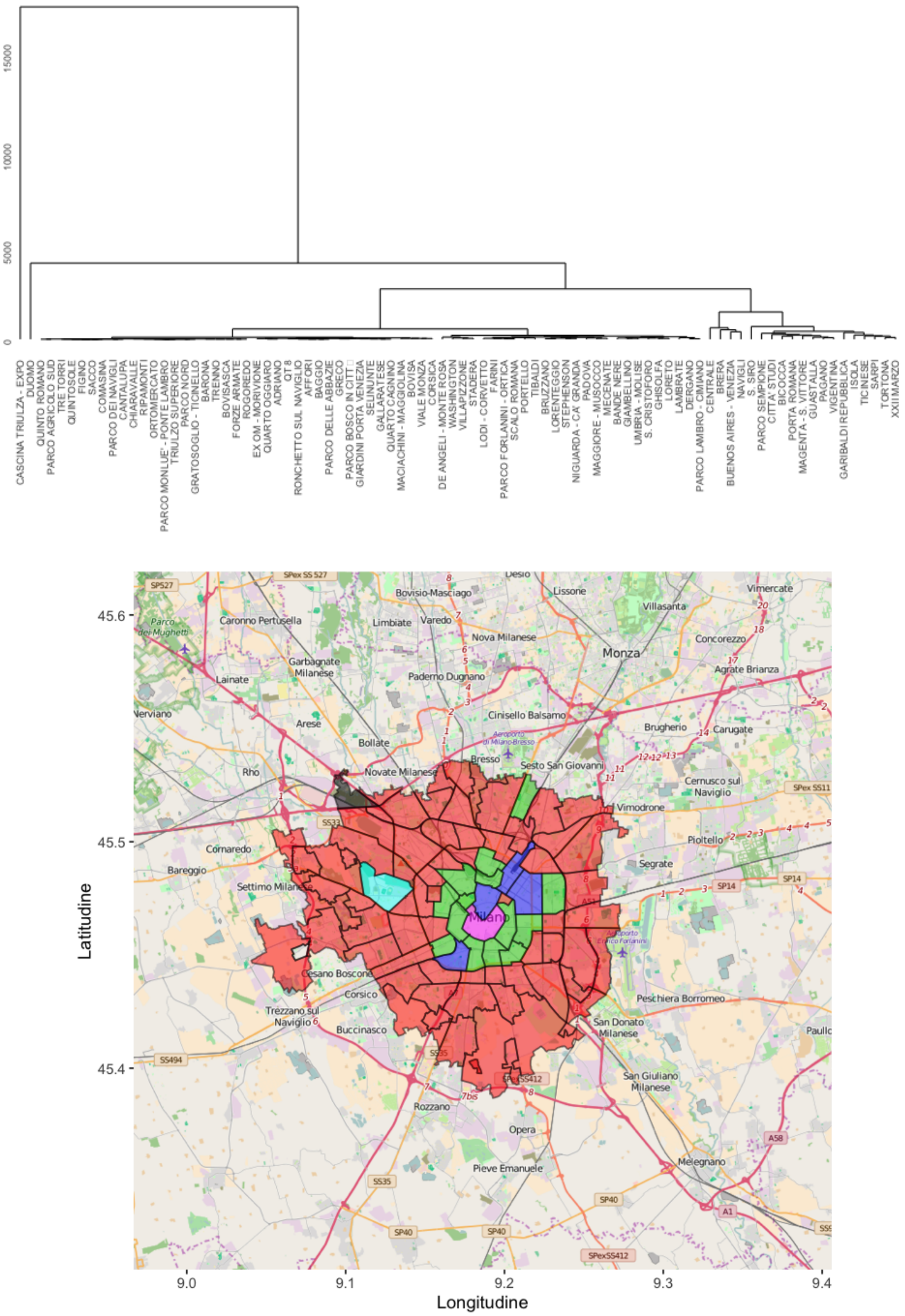


Figure 4.36: NILs clustering for the month of September 2015

4.5.2 Vectorial approach

The second approach is the vectorial one. It is based on the view of the array 3D as a matrix whose elements are vectors composed by 15 values, one for each month as described in Section 4.4.1. One can notice that this view is a summary of all the data. Any Clustering algorithm here performed will generate general clusters which take into account the 45 selected categories and all the 15 months. The resulting clusters, therefore, optimally summarise all the time period analysed.

However, having a full description of the whole dataset means not knowing the dynamic of Milan that only a month by month approach can guarantee. In addition, it is necessary to transform the array 3D in a matrix in the way explained in Section 4.4.1.

Figure 4.38 shows the resulting dendrogram and map of the hierarchical clustering. It has been decided to identify 8 clusters. In fact, it seems to be the best number in order to show details without granularity and it agrees with other approaches.

It is easy to notice that the majority of the map is red. It corresponds to a peripheral or residential area which is not particular attractive for Foursquare users. Instead, generally, the center of Milan is characterised by the presence of many clusters. As expected, these central NILs, for example Duomo, Brera, Buenos Aires - Venezia, Navigli and Porta Romana, are in general the vibrant core of the city and they always attract citizens and tourists. However there are also some suburban NILs standing out from the red cluster. It is the case of San Siro, Bicocca and Cascina Triulza - Expo, three zones which present important venues or events, as explained in Section 4.5.1.

The clusters are the following ones:

- Grey: Duomo;
- Light Blue: Navigli, Brera, Buenos Aires - Venezia, Centrale
- Blue: Isola, Magenta - S. Vittore, Garibaldi Repubblica, Sarpi, Porta Romana, Ticinese, Tortona;
- Yellow: Parco Sempione;
- Green: Bicocca;
- Magenta: San Siro;
- Black: Cascina Triulza - Expo
- Red: Parco Monlué - Ponte Lambro, Parco dei Navigli, Comasina, Muggiano, Figino, Quintosole, Chiaravalle, Cantalupa, Parco Agricolo Sud, Quarto Cagnino, Quinto Romano, Quarto Oggiaro, Rogoredo, Adriano, Parco Nord, Baggio, Gratosoglio

- Ticinello, Triulzo Superiore, Barona, Forze Armate, Sacco, Tre Torri, Trenno, Bovisasca, Parco Bosco in Città, Ortomercato, QT8, Affori, Parco delle Abbazie, Ronchetto sul Naviglio, Giardini Porta Venezia, Bruzzano, Greco, De Angeli - Monte Rosa, Lambrate, Ghisolfa, Parco Lambro - Ciminiano, Dergano, Maggiore - Musocco, Villapizzone, Niguarda - Ca' Granda, Stephenson, Padova, Washington, Loreto, Corsica, Tibaldi, Lodi - Corvetto, Bande Nere, Giambellino, Mecenate, Ex Om - Morivion, Parco Forlanini - Ortica, Farini, Scalo Romana, Lorenteggio, Maciachini - Maggiolina, Selinunte, Stadera, Ripamonti, Bovisa, Gallaratese, Viale Monza, Guastalla, Città Studi, XXII Marzo, Pagano, Umbria - Molise, Vigentina, Portello, S. Cristoforo.

NILs standing out as clusters of their own are characterized by events that modify the usual profile. Figure 4.37 can help the interpretation. It shows the number of check-ins for every single NIL in every single month. It is important to remark that it is not a cumulative information.

The NIL Duomo is the most visited zone in the whole city of Milan and, thus, it creates a cluster all by itself. It is nice to remark how it became more fashionable thanks to EXPO2015.

The light blue and blue NILs are quite visited during the whole year. They did not receive the impact of EXPO2015 and remained identical through time showing similar profiles with positive peaks in January and June 2015.

The cluster composed by Bicocca, the green one, presents low profile for all the months with the exception of December 2014, May and June 2015. As already explained in Section 4.5.1, the top venue in Bicocca is the Multiplex UCI Cinemas Bicocca and, thus, its film scheduling and discount politics could have had an impact on the number of check-ins. However it is mandatory to remark the presence of the University of Bicocca.

Same story for the clusters of San Siro and Parco Sempione, coloured in magenta and yellow, respectively. These two NILs get popular particular during Summer (with peak in June) thanks to the presence of concerts and en plein air events.

The red NILs compose the suburban cluster and are characterised by a number of check-ins which is around zero. As said before, these are parks or residential areas that do not attract Foursquare users or visitors.

The black cluster counts the only NIL of Cascina Triulza - EXPO. It is one of the most visited NIL in Milan starting from May 2015, the inaugural month of EXPO2015. It has a growing profile thanks to the exposition and, therefore, it is easy to imagine that without this important event it would have been clustered in the red group.

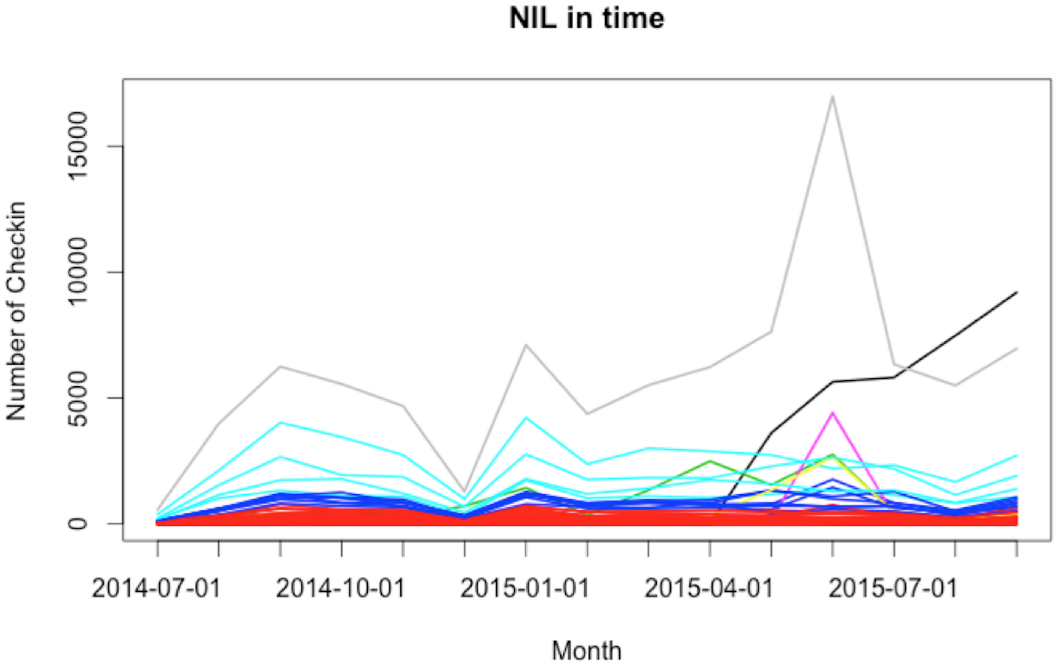


Figure 4.37: The profile of each NIL coloured by the cluster they belong to. Each line corresponds to a NIL

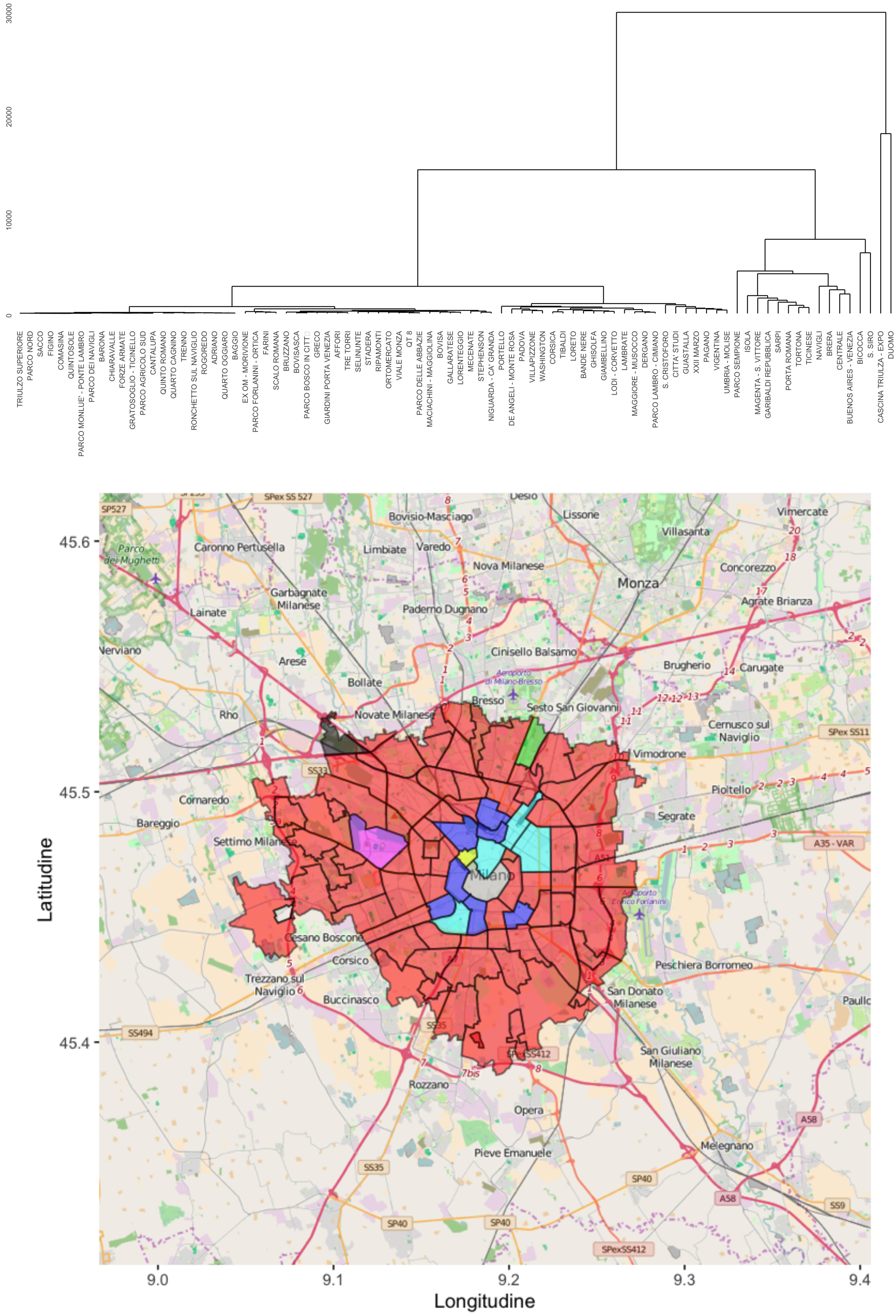


Figure 4.38: The 8 clusters obtained using a vectorial approach

4.5.3 Summed approach

The third approach here proposed is called Summed approach. It is performed on a single matrix resulting from the sum of all the layers (months) composing the array 3D. For this reason, it is said to be “summed”. As one can notice, differently from the vectorial approach, it does not represent a real situation. In fact, there is an important hypothesis that must be considered: the matrix, which the Clustering procedure operates on, describes the city of Milan as if all the check-ins, which happened in fifteen different months, took place in one. Therefore all the results here presented do not have to be taken as a way to describe reality but as a useful tool to verify results.

Figure 4.39 shows dendrogram and map. It has been chosen to cut the dendrogram in order to find 8 different clusters to compare with the 8 ones found with the vectorial approach.

It is clearly evident that many clusters previously described reappear. In general, there are identical patterns coming back. It is the case of Duomo, Bicocca, San Siro and Cascina Triulza - Expo as clusters of their own but it is also the case of Brera, Centrale, Buenos Aires - Venezia and Navigli grouped again together.

The main difference between the vectorial and summed approach is that, in the second view, the suburban cluster has been split in two: the red group and the green one. One can also notice that Parco Sempione is not a cluster of its own but it is grouped together with Garibaldi Repubblica, Isola, Ticinese, Tortona, Porta Romana and Sarpi.

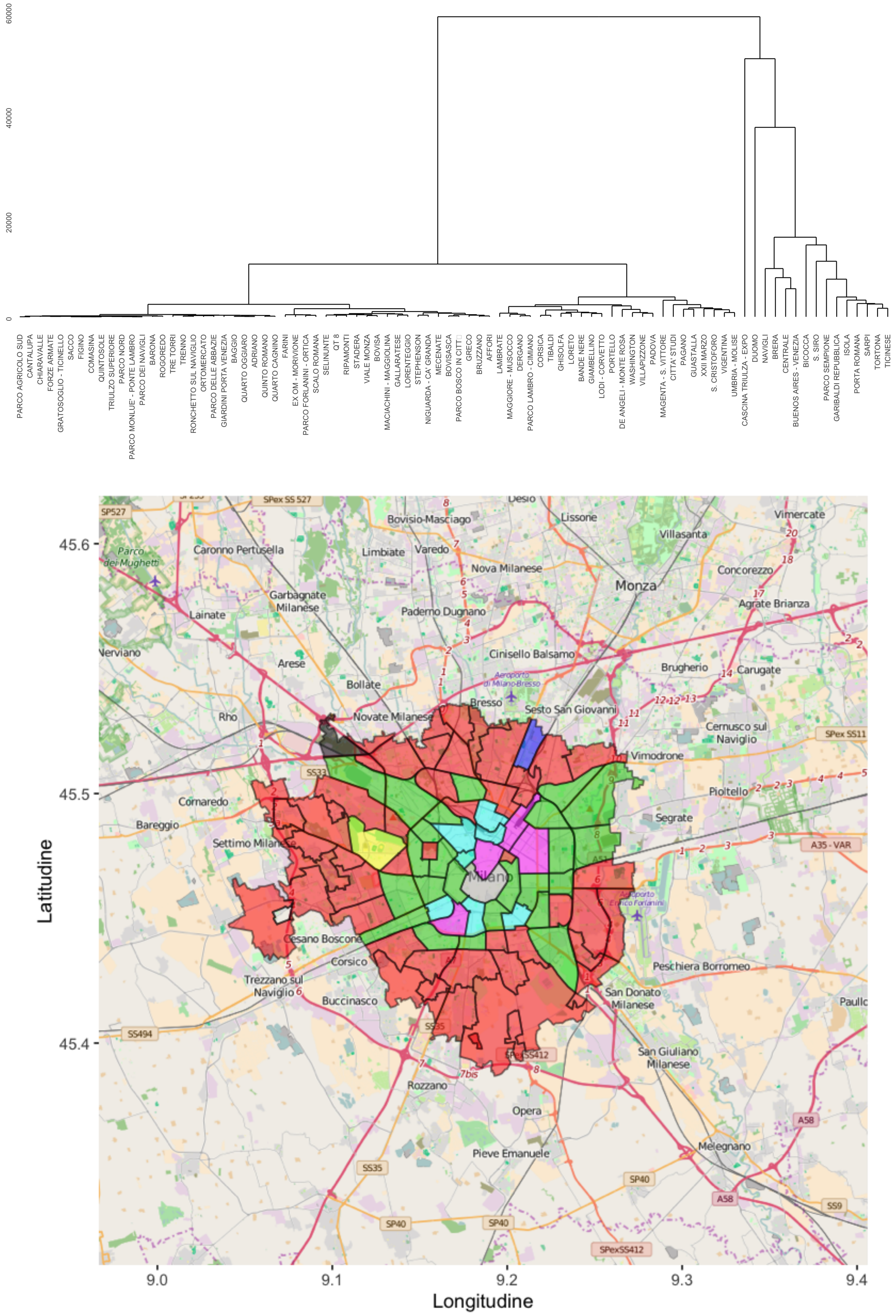


Figure 4.39: The 8 clusters obtained using a summed approach

Chapter 5

Creating the lens

Everything is illuminated.

Jonathan Safran Foer

In this last chapter the reader will understand how notions and procedures previously introduced lead to the completion of this thesis: the creation of the “Analyse” section in the Urbanscope “City magnets” module. Even if, for the moment, the problem has been merely analysed from a statistical point of view, many other knowledges are necessary to build a lens. Therefore, in Section 5.1 it has been decided to remark the importance of all the elements composing the Urbanscope project. Section 5.2 concludes the analytical trial, started in Chapter 1, by performing the Chung and Church’s algorithm on our data and showing the results. The charts created will be the basis of the “Analyse” section of “City magnets” . Due to the fact that Urbanscope is an educational project for everyone’s use, it is still important to create appealing, easy-to-read graphs. The synthesis and the visualisation of biclusters in order to ease the comprehension is therefore a vital issue that will be analysed in Section 5.3. Therefore, after many managerial and graphical hypotheses, the final view, expressly designed for the project, will be finally presented.

5.1 A two-faced Janus

The Urbanscope team is composed by researchers with competencies in Computer Engineering, Management Engineering, Mathematics and Communication and Information design. Every figure is fundamental: the computer engineer is the one who can download the raw data from the digital world, the management engineer is the figure that identifies stakeholders, captains the work and circumscribes the problem, the mathematical engineer gets involved in the data analysis and, last but not least, designers can help by representing data and reordering them in a more comprehensible way. All these people, coming from different backgrounds, contribute to the difficult mission of creating and divulging Milan real-time statistics and analysis. This is a two-sided task: implementation and visualisation. In fact, like the greek god of Janus, depicted as having two faces because he can see indoor and outdoor at the same time, Urbanscope project tries to watch inside the digital world referring to Milan to express all the gained information outside, to the whole web.

The first side of this bifaced Janus is more typical to engineers rather than designers. In fact, it is necessary to download and inspect the data in order to find interesting patterns. However, even if fascinating, it is always difficult to make the analysis without any goal to reach. Management engineers define the purpose of the research. They set the directions that computer and mathematical engineers follow. In fact, according to the final aim, data is downloaded, treated, integrated and processed in a particular way. After that, designers and mathematical engineers can start exploring the data. The exploration can be very informative and it can enrich and guide the creation of the "Explore" section. For the "Analyse" section mathematical engineers are the main actors. In fact, a deep knowledge of mathematical and statistical tools are usually mandatory in order to make a deep and coherent inspection.

On the contrary, the second side of the problem concerns explication and visualisation. Precisely, once the analysis is ready, one has to explain it and show it to the world. In fact, the digital platform of Urbanscope does not have a defined user base because it wants to reach whoever is interest in the evolution of the city. Therefore, the analysis proposed and represented in the website has to be informative and understandable to everyone, even if without any technical knowledge. The results, sometimes obtained by really complicated algorithm, need to be explained in a simple manner but without trivializing the mathematical complexity. This is a very difficult task that requires strong representative and communication skills characterising designers. Also a managerial approach is requested in order to coordinate all the pieces of the procedure and to give interpretation to the results.

Therefore, the two faces of the project demand to mix different branches of knowledge together in order to have the most self-explanatory and homogeneous representation of Milan evolution.

In the next sections the reader will follow all the steps that lead to the creation of the "Analyse" section of the "City magnets" lens. Firstly, the statistical point of view characterised by the previously introduced Biclustering algorithm is presented. In a second instance, the resulting biclusters are commented from a different point of view that will allow a better and simpler representation. This is the management engineering field. Lastly, everything is studied and designed in order to express the obtained results without any inconveniences.

5.2 The application of Cheng and Church's Biclustering algorithms

The creation of the lens starts with the definition of the final goal. In fact, it is important to define the purpose of the analysis, namely the pattern to discover. In our case, for the "City magnets" module, it has been decided to look for the relation between places, identified by the NILs, and the categories proposed by Foursquare. Precisely the goal is really punctual: find which NILs are similar referring to some category, i.e. identify biclusters. It has been decided to find biclusters for each of the 15 months in order to complete the analysis by making comparisons through time. Therefore, it is necessary to apply a Biclustering algorithm. The attentive reader knows from Chapter 1 that there are many algorithms which can be performed. However, the large variety of alternatives is reduced to those algorithms resulting in biclusters with constant or coherent values. In fact, as already said, the analysis aims at finding submatrices with a constant or coherent number of check-ins. Every method has its own pros and cons and it has been decided to utilise the procedure proposed by Cheng and Church. As already explained in Chapter 1 and in Chapter 2, the algorithm aims at finding biclusters with coherent values as a subset of rows and a subset of columns, whose values are given by the following expression:

$$a_{ij} = \mu + \alpha_i + \beta_j \tag{5.1}$$

where μ is the typical value within the bicluster and α_i and β_j are two adjustments, respectively for the rows $i \in I$ and the columns $j \in J$. Thus, in the case considered in this thesis, α_i refers to NILs whereas β_j to categories. Values, therefore, are assumed on an underlying additive model that is part of the one used in two-factor analysis of variance model. This fact allows to explain the Cheng and Church's procedure using the ANOVA framework. Precisely, the mean squared residue score

$$H(I,J) = \frac{1}{\mid I \mid\mid J \mid} \sum_{i \in I, j \in J} (a_{ij} - a_{iJ} - a_{Ij} + a_{IJ})^2 \tag{5.2}$$

where $a_{iJ} = \frac{1}{|J|}\sum_{j \in J} a_{ij}$ is the NIL means, $a_{Ij} = \frac{1}{|I|}\sum_{i \in I} a_{ij}$ is the category means and $a_{IJ} = \frac{1}{|I||J|}\sum_{j \in J, i \in I} a_{ij} = \frac{1}{|I|}\sum_{i \in I} a_{iJ} = \frac{1}{|J|}\sum_{j \in J} a_{Ij}$ is the mean of the matrix, corresponds to the error term of the variance of a two-factor ANOVA response variable. The Cheng and Church's Biclustering method uses this expression to measure the coherence of a set of NILs under a set of categories and to qualify biclusters. The lower is the score the better is the biclusters. Thus, the perfect bicluster has $H(I,J) = 0$ which means a variance arising from the error term equals to $a_{ij} = a_{IJ} + (a_{iJ} - a_{IJ}) + (a_{Ij} - a_{IJ})$ for $i \in I$ and $j \in J$.

JANUARY 2016	Hotel	Pizza Place	Italian Restaurant	Cocktail Bar
Brera	316	225	587	335
Duomo	568	571	1074	525

Table 5.1: An example of bicluster in the form of table

By applying Cheng and Church's Biclustering method, one can notice that the number of resulting biclusters varies month by month but it never gets too big. In addition, at least one bicluster is found for each month. In Section 5.2.1 all the details and the results will be shown and commented.

5.2.1 The resulting biclusters

As previously described, the algorithm that has been used is the one proposed by Cheng and Church. Precisely it has been applied separately on all the 15 layers of the original array 3D in order to find biclusters for each month. Many possible representations of the dataset are available depending on the information one wants to show. If one is interested on how the NILs are grouped together, the best available view is guaranteed by a coloured map: similar neighbourhoods are characterized by the same color as seen for the clusters of NILs. Otherwise, if one prefers to visualise an insight of the categories referring to a particular bicluster, a barplot visualisation is the ideal solution. In fact, it presents all the 45 categories and if one of them is in the subset of categories that characterises the bicluster then it is placed side by side to a bar. Therefore all the categories belonging to the bicluster have a bar next to them. The length of the bar refers to the average number of check-ins calculated for the specific category in all the NILs in the bicluster. In addition, it is important to remark that if some category belonging to the bicluster have an average number of check-ins of 0 then it presents a fictional bar with length equal to zero. Therefore, the reader needs to pay attention to the difference between categories characterising the bicluster with an average of zero, and categories which are outside the bicluster and they do not have any bar. In order to ease the comparison between biclusters and population, it has been decided to show also the average number of check-ins for each category in all the NILs as a red line.

However it is quite difficult to represent the whole situation of a bicluster in a cohesive way. One can show all the informations in a submatricial way, as done in Table 5.1, but the best visualisation remains the juxtaposition of the two graphs discussed before. Therefore, it has been deliberated to show the biclusters for each of the 15 month using both the map for the NILs and the barplot for the categories. Unfortunately the two representations are not perfectly connected one to another because of the lack of personalisation tools. Therefore, it is necessary to remark how the labels of the barplot refer to biclusters represented in the map. In fact, in the barplot every bicluster is named using a capital letter, i.e. “A”, “B”, “C” and so on. Instead, in the map, a visual grouping based on matching colours has been used. The two visualisation are connected in the following way:

- A - black;
- B - red;
- C - green;
- D - blue;
- E - light blue;
- F - purple;

It is trivial to understand that these two representations, maps and barplots, need to be joined in a better way, in order to give all the information instantaneously and to solve all the matching issues between labels and colours. For the moment, all the biclusters referring to each month are shown in Figures 5.1-5.15.

Firstly, let us remark how in every month there are at least three biclusters with a maximum of six biclusters in Figure 5.9, Figure 5.10 and in Figure 5.13. In addition, it is possible to notice that there is always a big bicluster coloured in black characterised by categories with an average number of check-ins around zero and then inferior to the average in the population. This is due to the fact that the algorithm looks for the biggest submatrices that minimize the mean squared residue score. Therefore, all the peripheral areas which are not very attractive and have a low number of check-ins are usually grouped together for many empty categories. In addition, it is natural that sometimes the resulting biclusters are trivial. For example, this is the case of Figure 5.4 that presents a similarity between Duomo, Guastalla and Buenos Aires - Venezia because they all do not have check-ins in categories such as Soccer Stadium which is only present in San Siro. Therefore, the Biclustering methodology can return biclusters having NILs grouped together for the categories they all do not have: it finds similarity in the empty categories.

In Figure 5.2, Figure 5.5, Figure 5.7, Figure 5.8, Figure 5.11 and Figure 5.15 it is possible to notice how the NIL Duomo is not coloured because it does not belong to any bicluster thanks to its high difference from the rest of the city. Instead, when it is grouped with other NILs such as Brera or Buenos Aires - Porta Venezia, the referring categories are usually over the population mean in the month.

One can also notice how month by month the description made using biclusters changes in accord to the events that characterised Milan. For example, the NILs of Cascina Triulza - EXPO, always grouped with peripheral NILs, because of categories under the population mean, from May 2015 starts to belong to other biclusters thanks to EXPO2015.

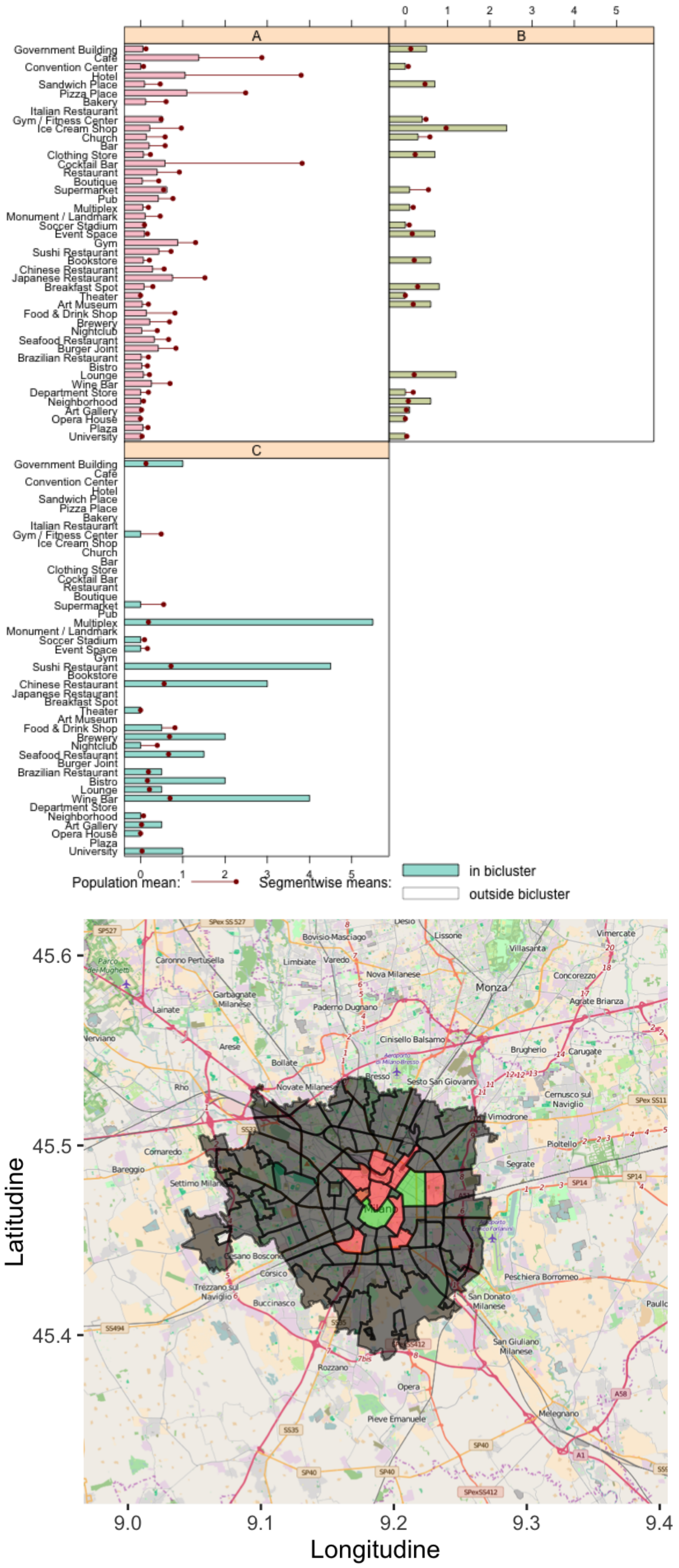


Figure 5.1: Biclusters for the month of July 2014

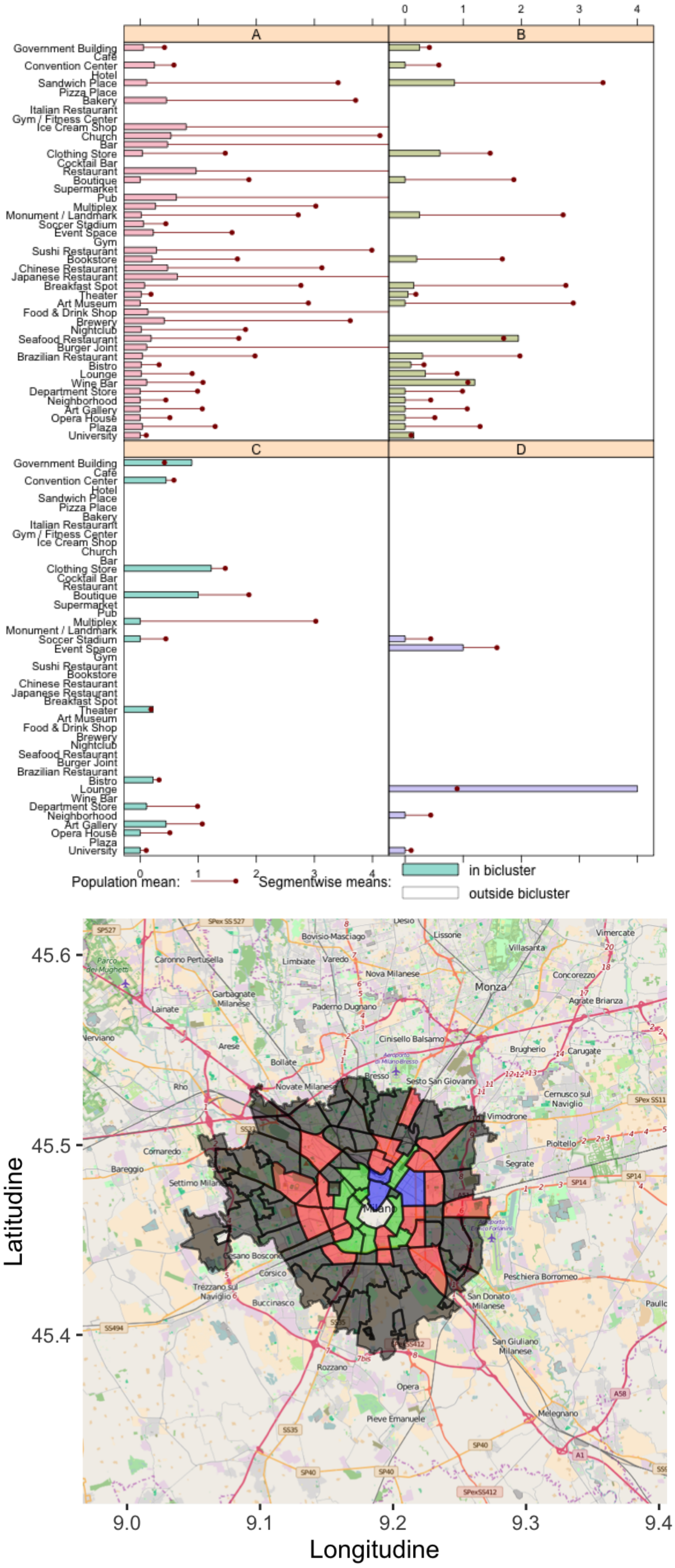


Figure 5.2: Biclusters for the month of August 2014

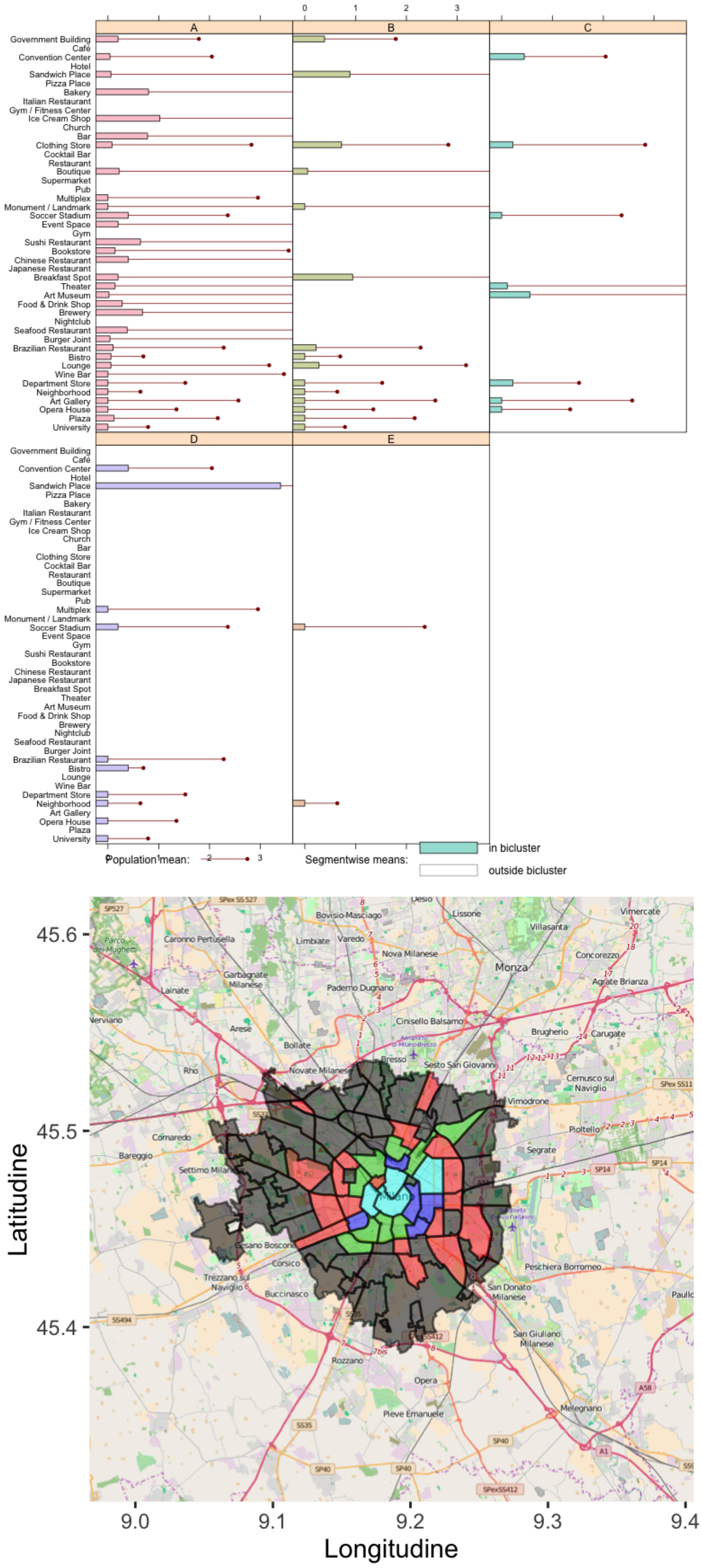


Figure 5.3: Biclusters for the month of September 2014

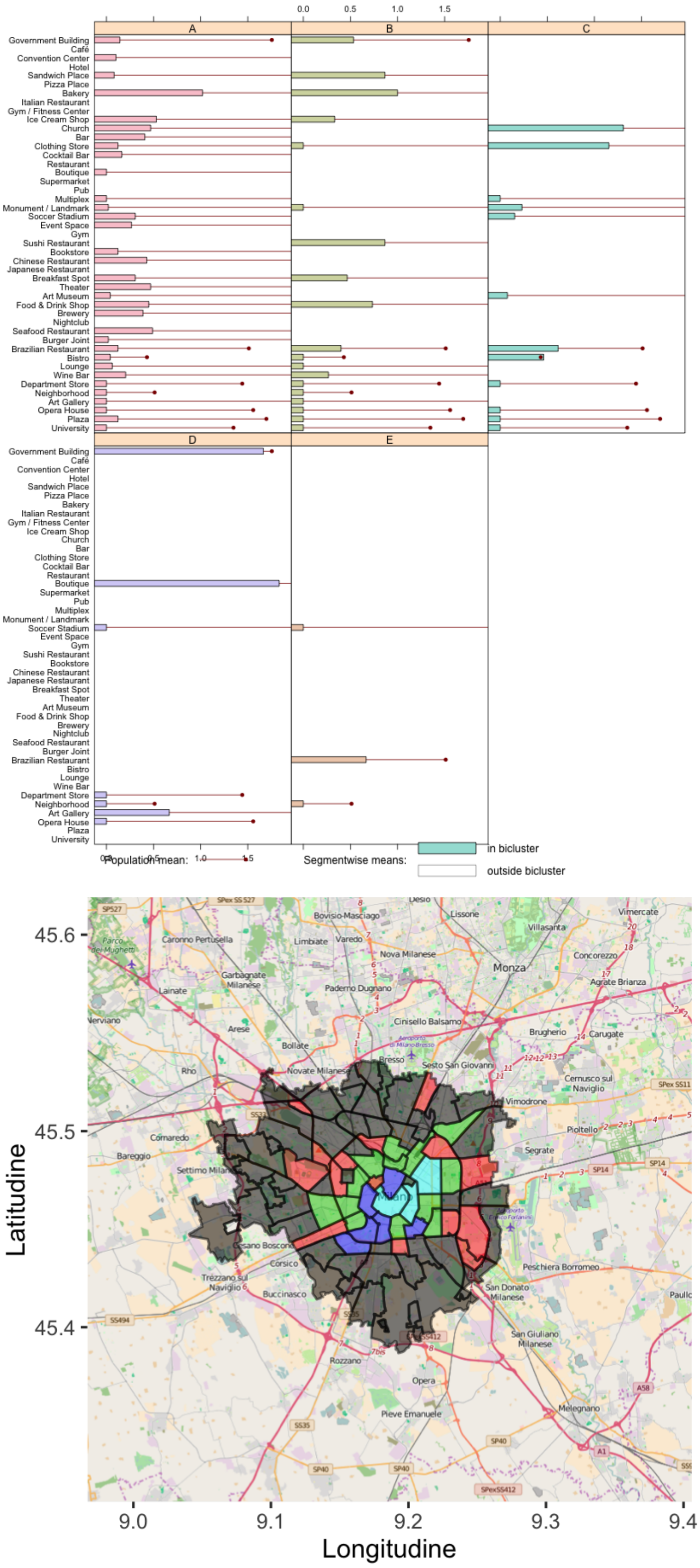


Figure 5.4: Biclusters for the month of October 2014

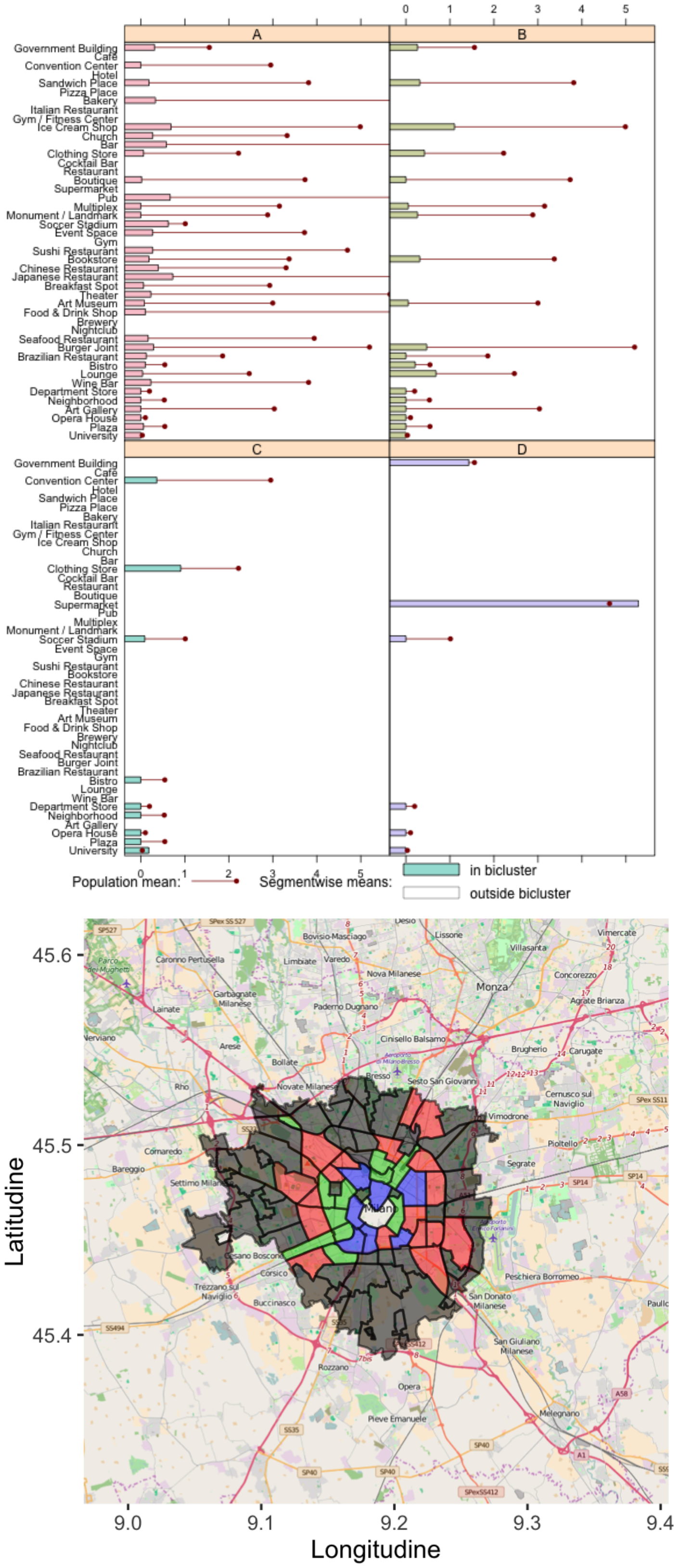


Figure 5.5: Biclusters for the month of November 2014

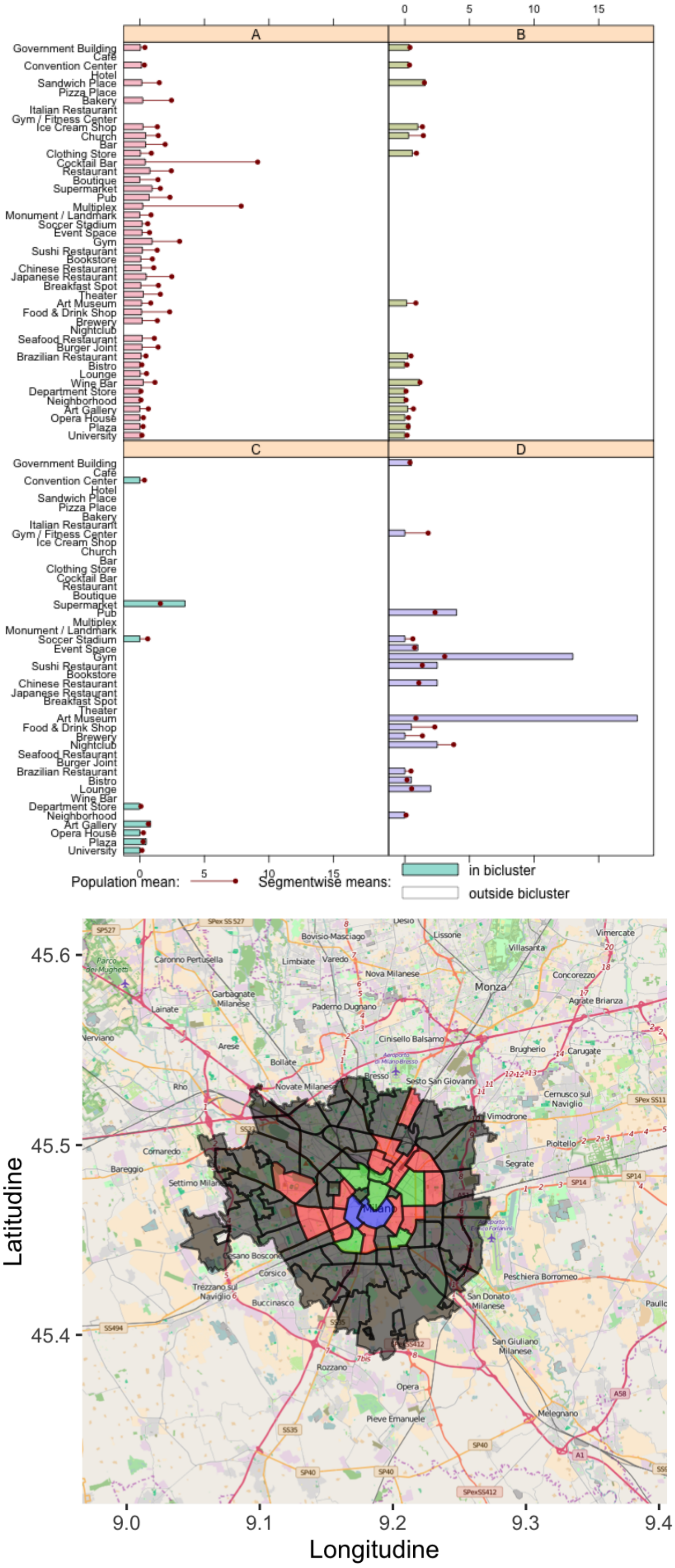


Figure 5.6: Biclusters for the month of December 2014

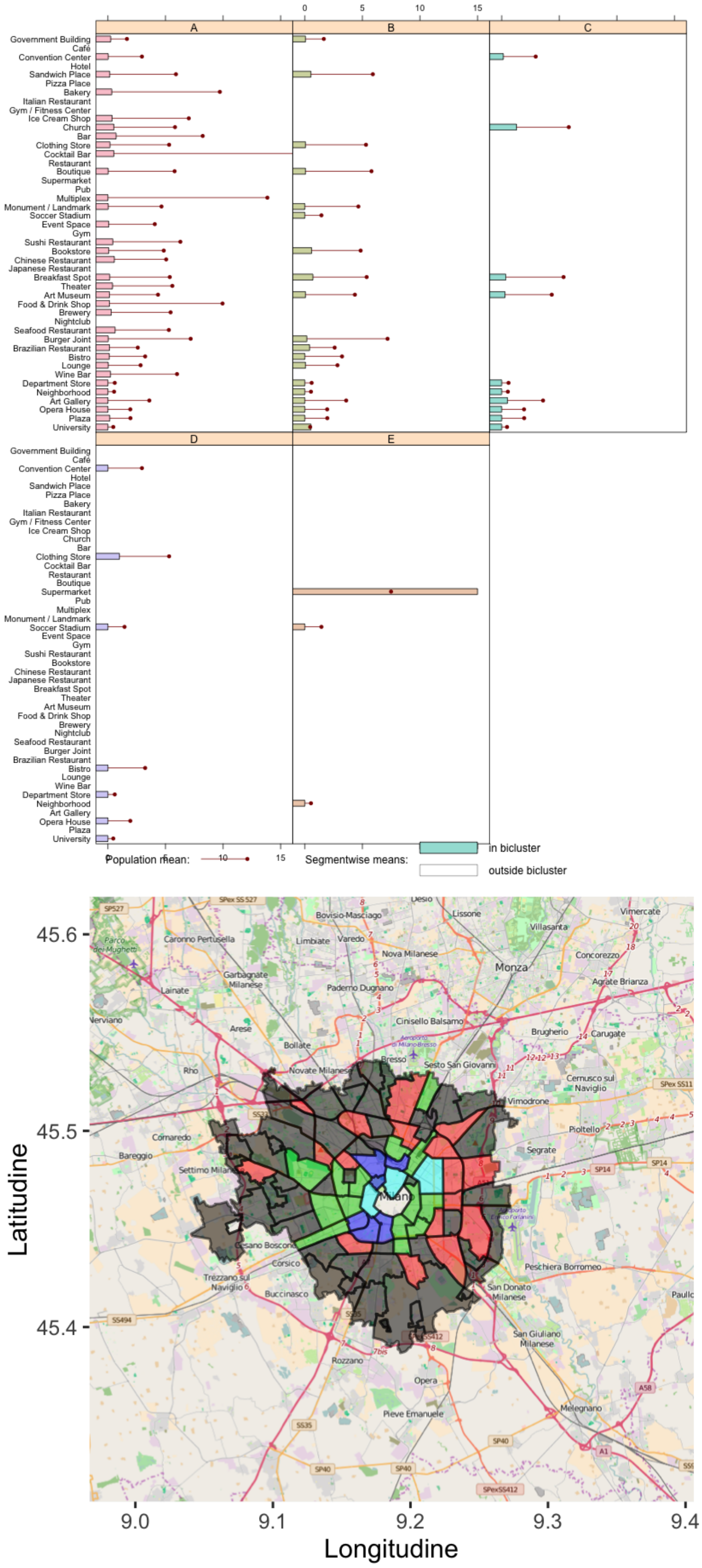


Figure 5.7: Biclusters for the month of January 2015

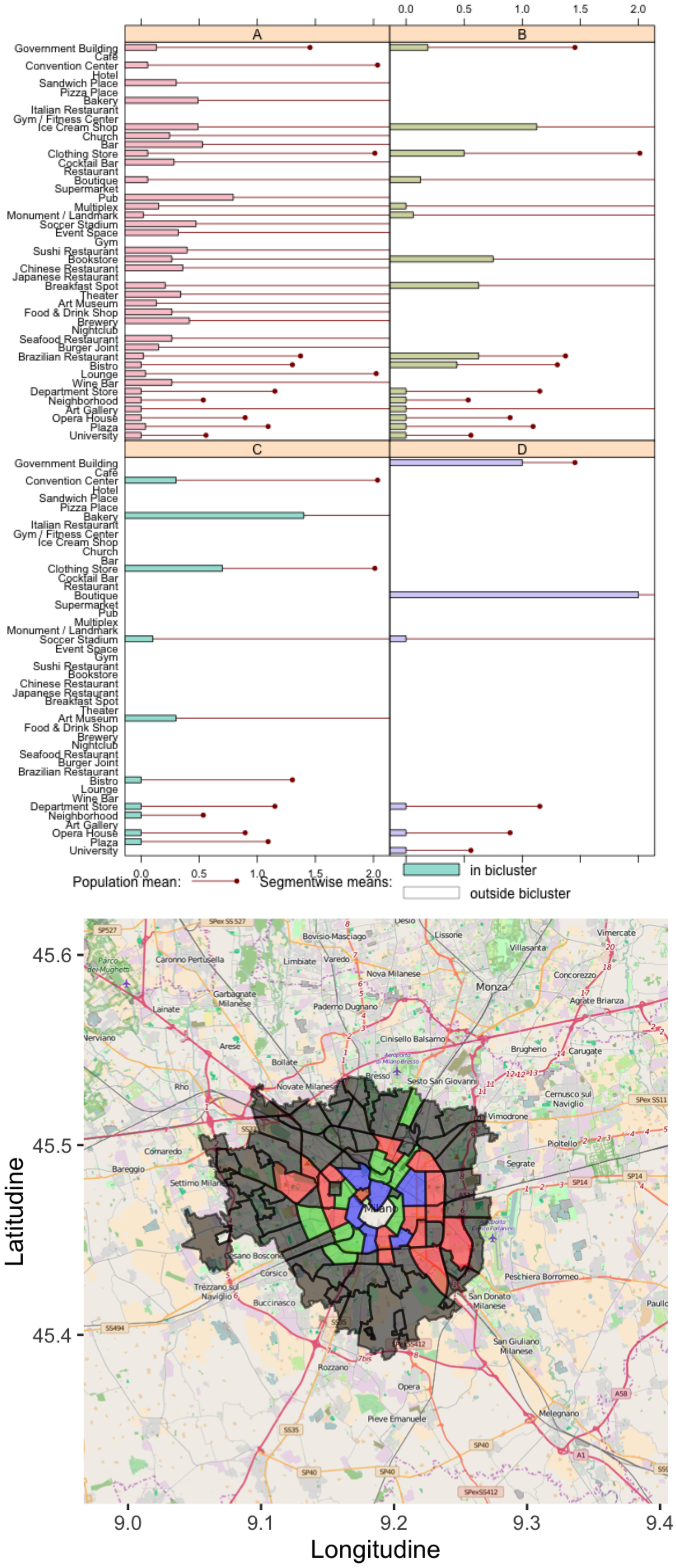


Figure 5.8: Biclusters for the month of February 2015

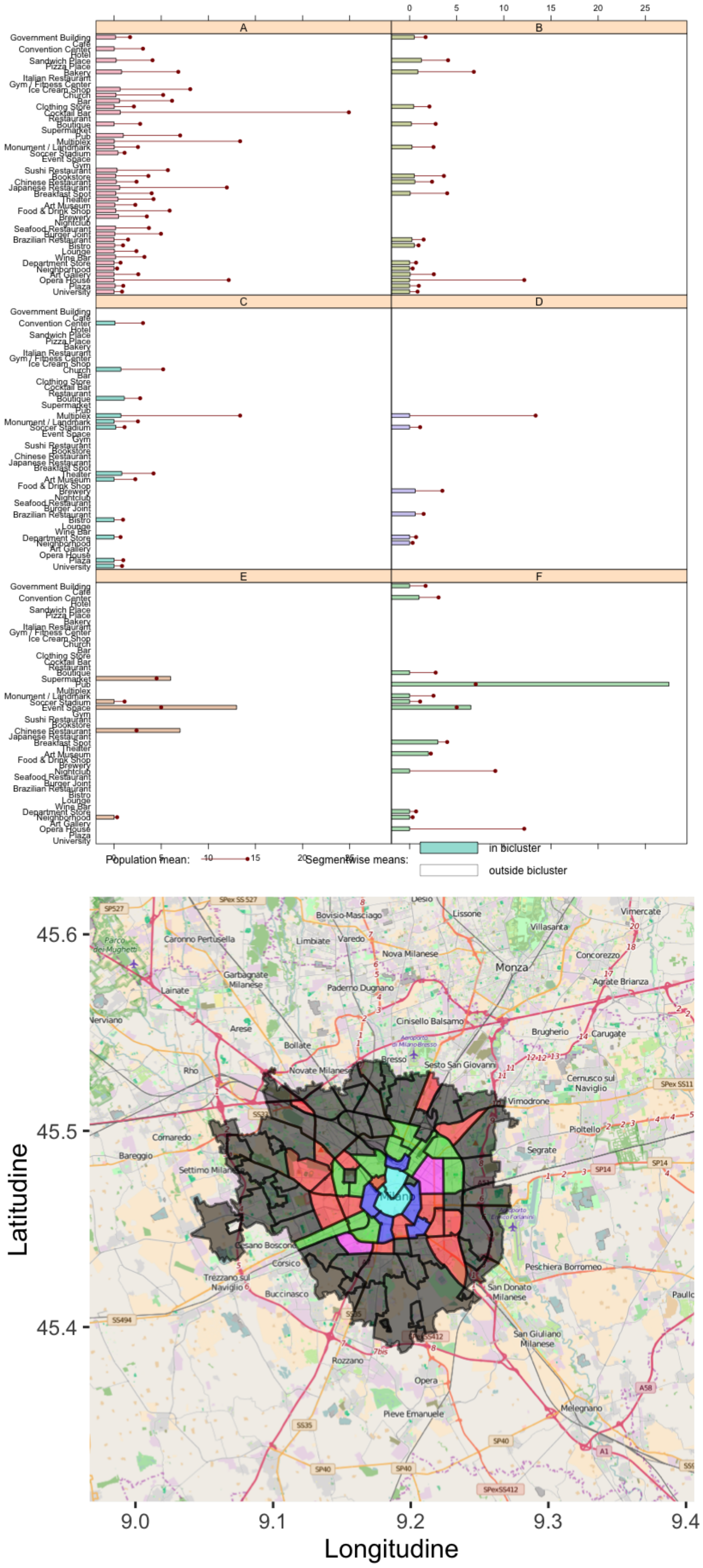


Figure 5.9: Biclusters for the month of March 2015

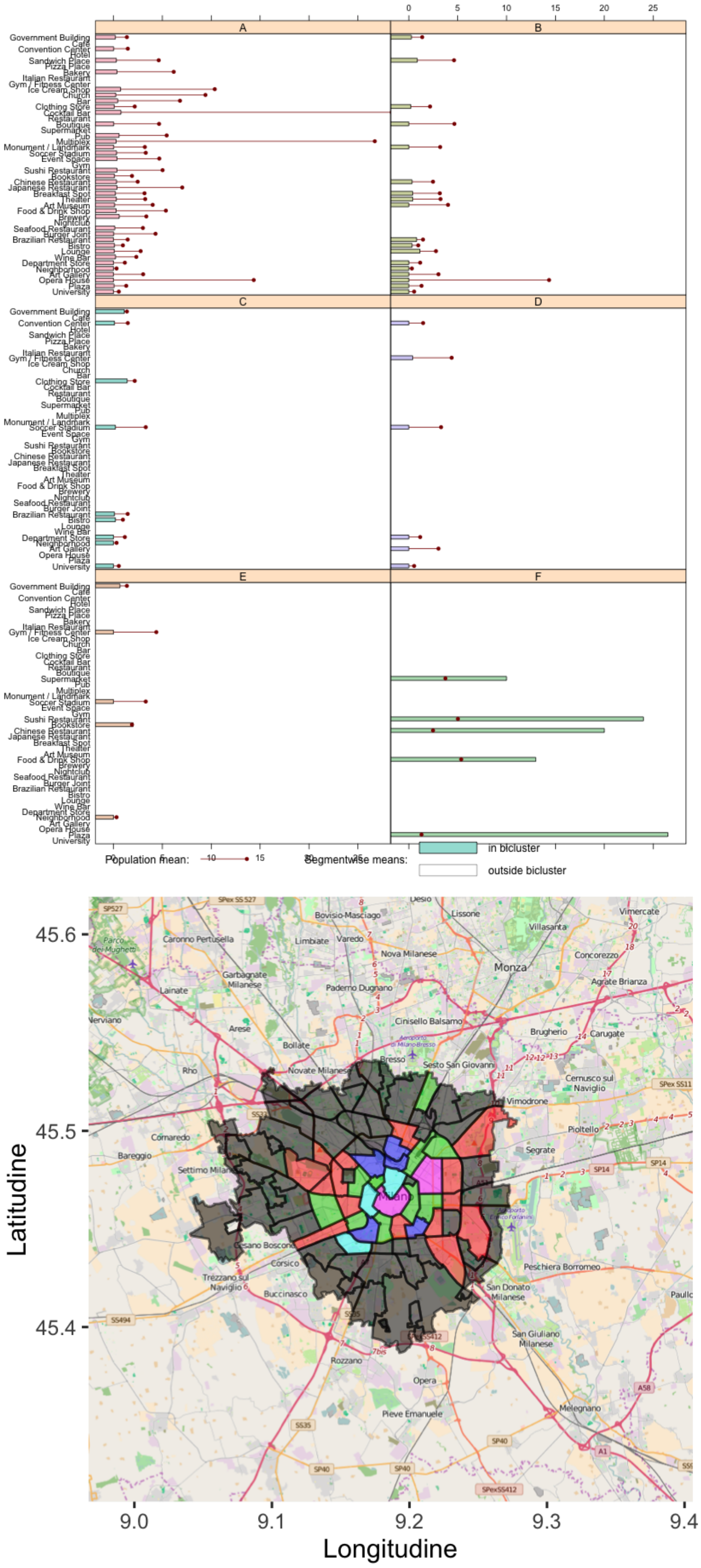


Figure 5.10: Biclusters for the month of April 2015

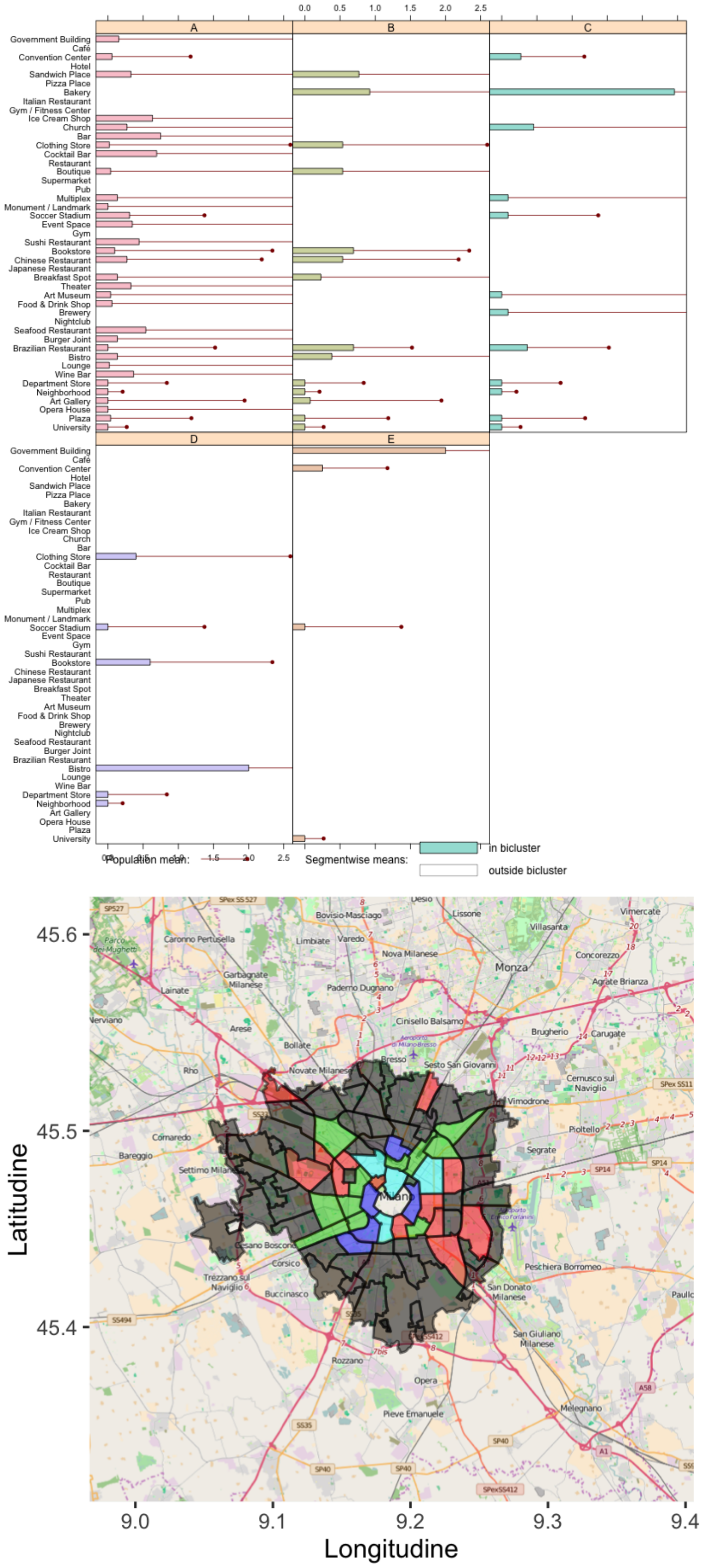


Figure 5.11: Biclusters for the month of May 2015

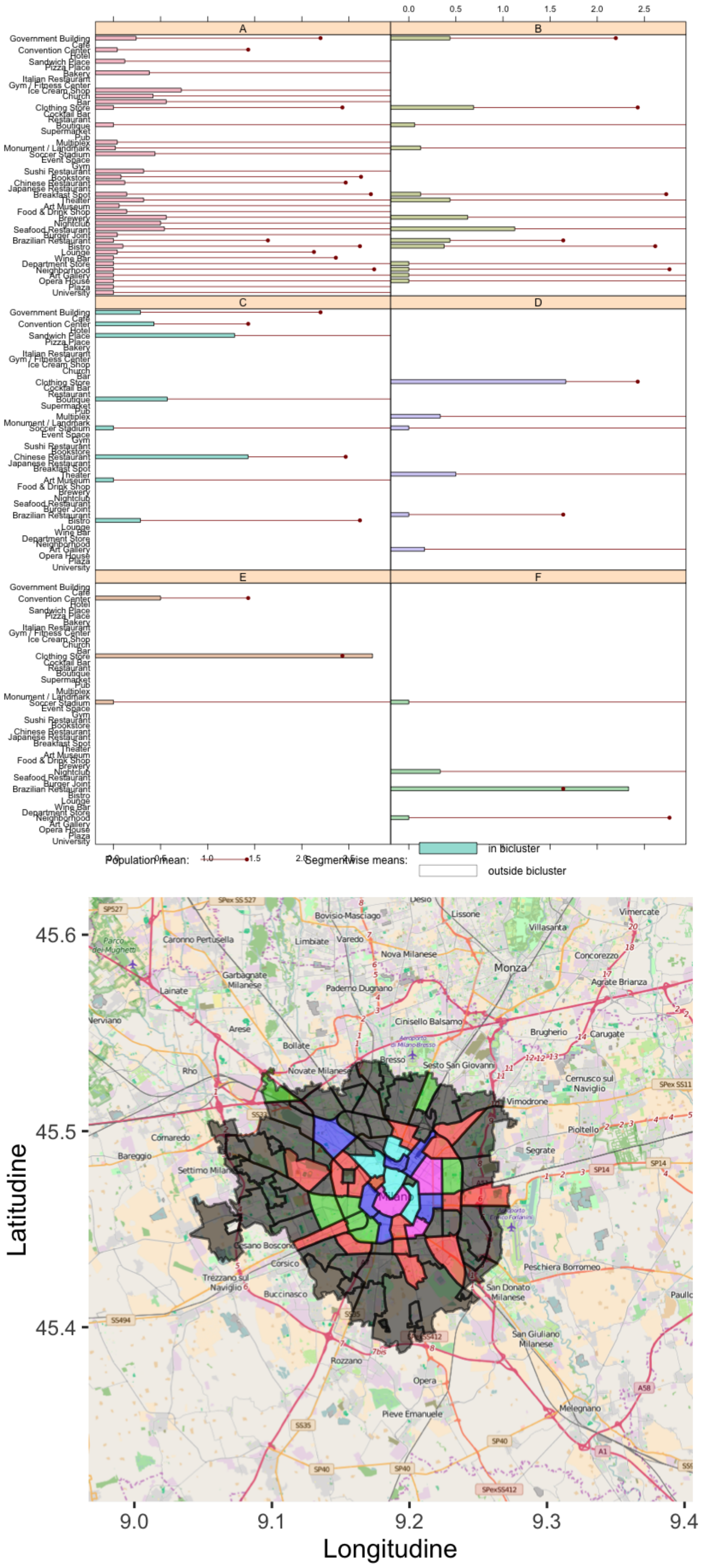


Figure 5.12: Biclusters for the month of June 2015

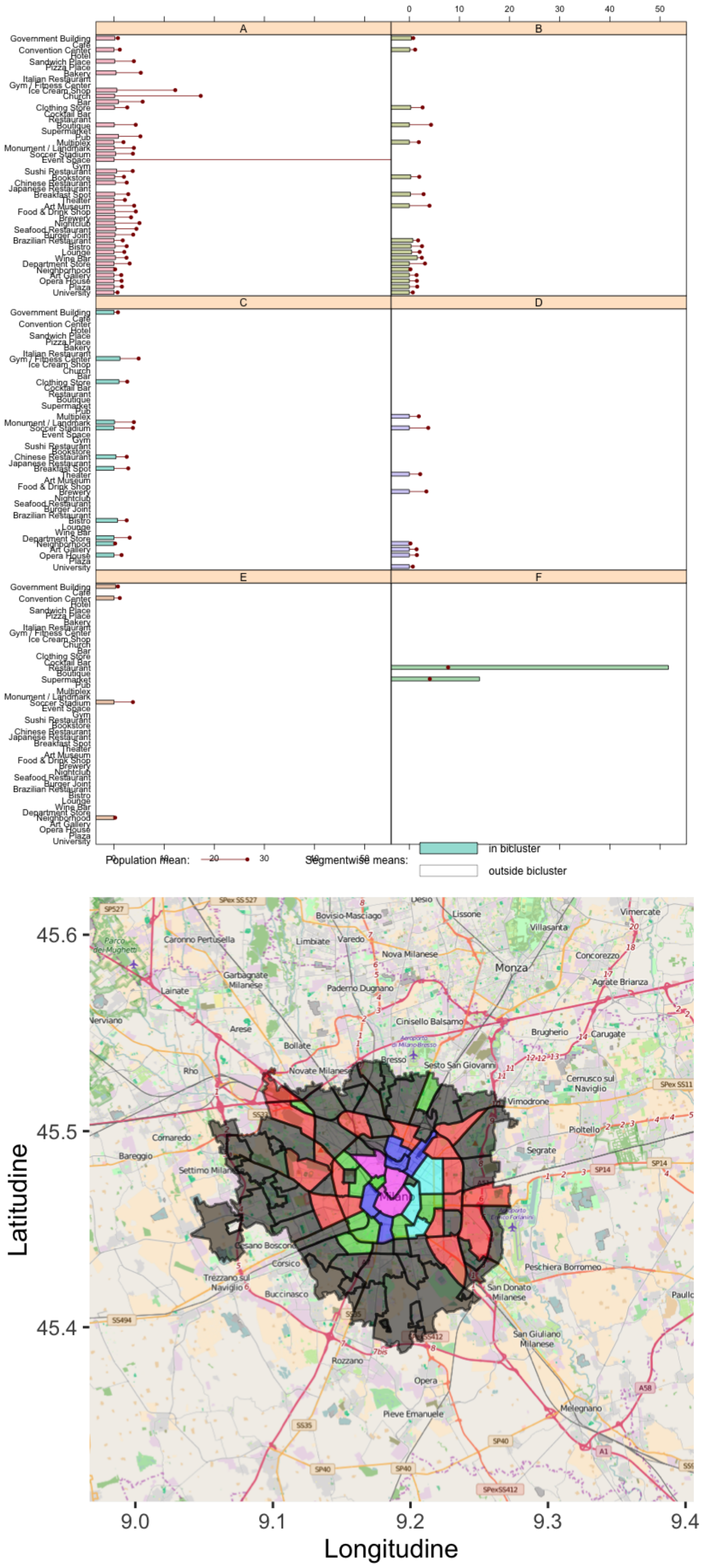


Figure 5.13: Biclusters for the month of July 2015

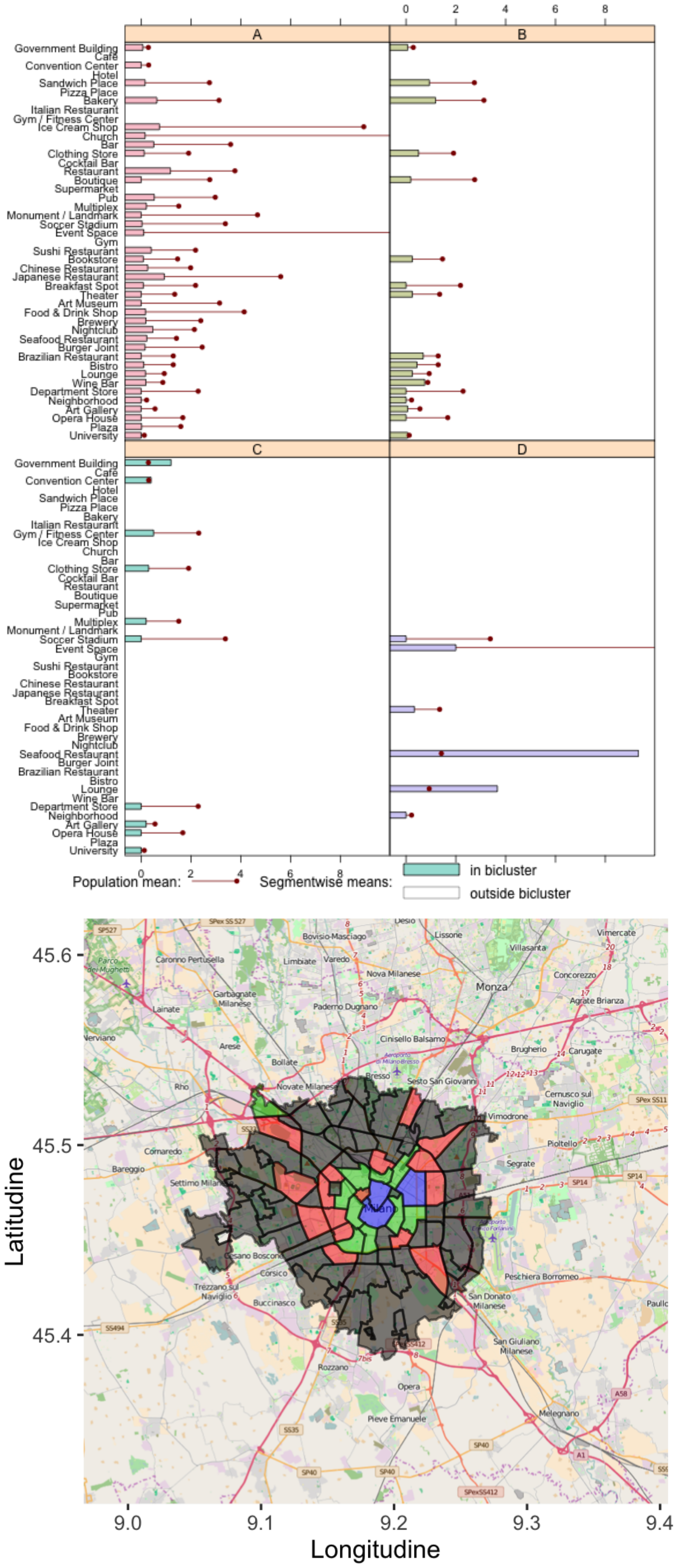


Figure 5.14: Biclusters for the month of August 2015

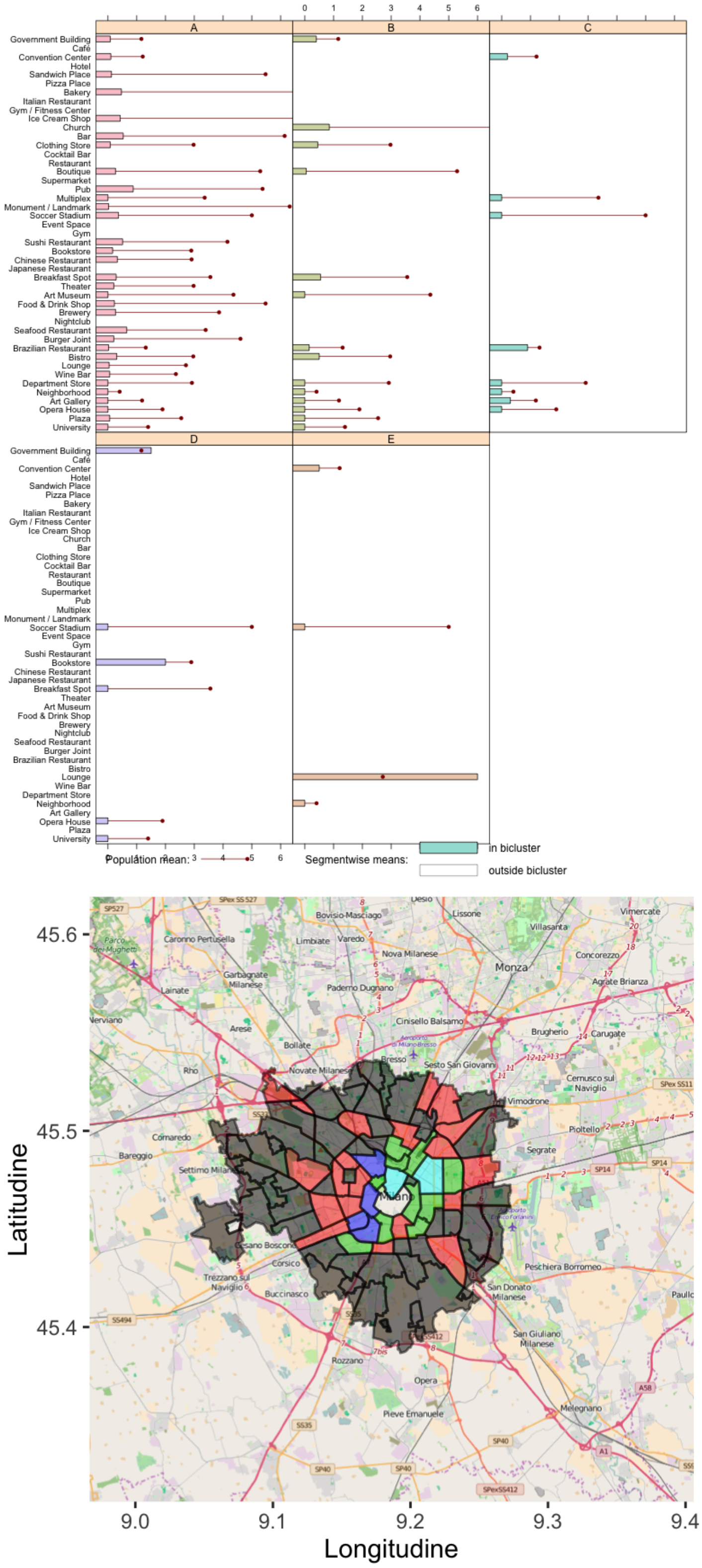


Figure 5.15: Biclusters for the month of September 2015

5.3 Creating the view

Section 5.2.1 showed the results obtained performing the Cheng and Church's algorithm on every single month presented in the dataset. Unfortunately they are not immediate to comprehend but they can only be used as a cornerstone to create the final visualisation. In fact, it is clear to everyone that the maps and the barplots need to be changed in order to have a polished view of the analysis done.

The major problem that needs to be solved is evident while watching the barplot describing the categories: they do not present any visual matching with the coloured maps, they are too small and it is difficult to read the labels of the categories and the length of the bars. It is therefore necessary to create an improved view approaching the problem from different points of views.

The first problem one can solve concerns the category labels. In fact, they are too numerous, often overlapped and difficult to read. Therefore, it has been decided to present a different kind of ordering, deciding to group them in order to ease the comprehension. In this way, the Urbanscope user will immediately understand which kinds of category identify the resulting bicluster, instead of scrutinising every single category one by one. However, in order not to intact the integrity of the analysis, all the categories are not blended but just conceptually organised. Therefore, the number of categories remains the same but the way they are proposed to the user completely changes. Precisely they are not just listed as shown in the barplots but they are divided in macro-categories. The decision of the organisation of the 45 categories in macro-categories depends on the criterion utilised. In this case two similar criteria have been proposed.

First of all, it is possible to use the category Clustering explained in Chapter 4. In Section 4.4.1, the 45 categories have been clustered together in 6 different clusters. In order to find a reason under the obtained groups, it has been decided to approach the problem from a managerial point of view.

This new perspective led us to manually modify the resulting clusters and name them in the following way:

- Events: Event Space;
- Student Life: University, Bakery, Ice Cream Shop, Monument / Landmark, Clothing Store, Art Gallery, Sandwich Place, Burger Spot, Boutique, Gym;
- Life Style: Japanese Restaurant, Pub, Bar, Restaurant, Government Boulding, Lounge, Brewery, Chinese Restaurant, Neighbourhood, Supermarket, Sushi Restaurant, Bookstore, Bistro, Theater, Seafood Restaurant, Brazilian Restaurant, Wine

Bar, Convention Center, Gym / Fitnees Center, Plaza, Food & Drink Shop, Nightclub;

- Church: Church;
- Mass Entertainment: Soccer Stadium, Multiplex;
- Tourist Life: Hotel, Cafè, Pizza Place, Italian Restaurant, Cocktail Bar, Art Museum, Department Store, Opera House;

The groups just shown are our new macro-categories and they reorganise all the categories. All the details about the macro-categories are explained in subsection 4.4.1.

The use of these macro-categories will be really helpful to reorder the barplot visualisation. Every categories will be presented under the corresponding macro-group. For example, Soccer Stadium will always be presented together with Multiplex under the label of "Mass Entertainment". Therefore, if one is only interested in the biclusters regarding the Multiplex category, he will not have to skim all the 45 categories but he will be able to go directly inside the Mass Entertainment macro-category. In addition, it is possible to give a more instantaneous information concerning the categories involved in the bicluster by using the macro-categories: if a macro-category does not present any category involved in the bicluster then it will not be highlighted. In this way, if one is interested only in the biclusters regarding the Multiplex category then he will only takes into account the biclusters which highlight the Mass Entertainment macro-category.

It is evident how this strategy leads to a more user-friendly view. However, one can notice that the macro-categories are sometimes chaotic and they can mislead the user. For example, it is not trivial which macro-category presents Sandwich Place. For this reason, it has been considered to use another criterion to organise each category.

The second way, that will not be used in the final prototype presented in this thesis, is principally based on a managerial point of view. Each category is organised in macro-categories according to its meaning and objective function. For example everything regarding food and restaurants will belong to the same macro-group. One can notice that this approach does not use any mathematical instruments but it is more intuitive. Therefore, it is now almost trivial to understand which macro-category presents Sandwich Place. However, as said before, the visualization proposed for the 'Analyse" section of the "City magnets" lens uses the first criterion, the one based on Clustering.

The prototype that has been created aims at showing a new way to visualise biclusters. The main elements that compose the view are two: the barplot and the map. The mission is to connect the two in an easy-to-read way and to show all the most important information at the same time. It has been thought that the user will have to select one of the 15 months he wants to investigate. Before doing so, the graph is quite empty as showed in Figure 5.16. All the months are presented in a clickable bar.

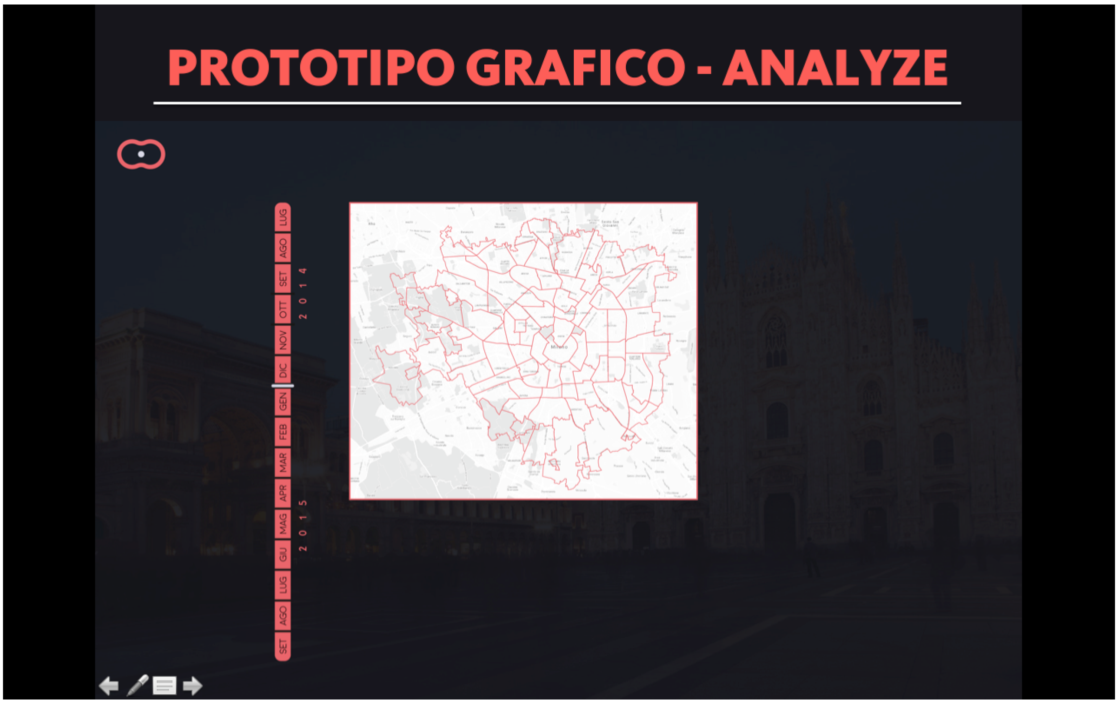
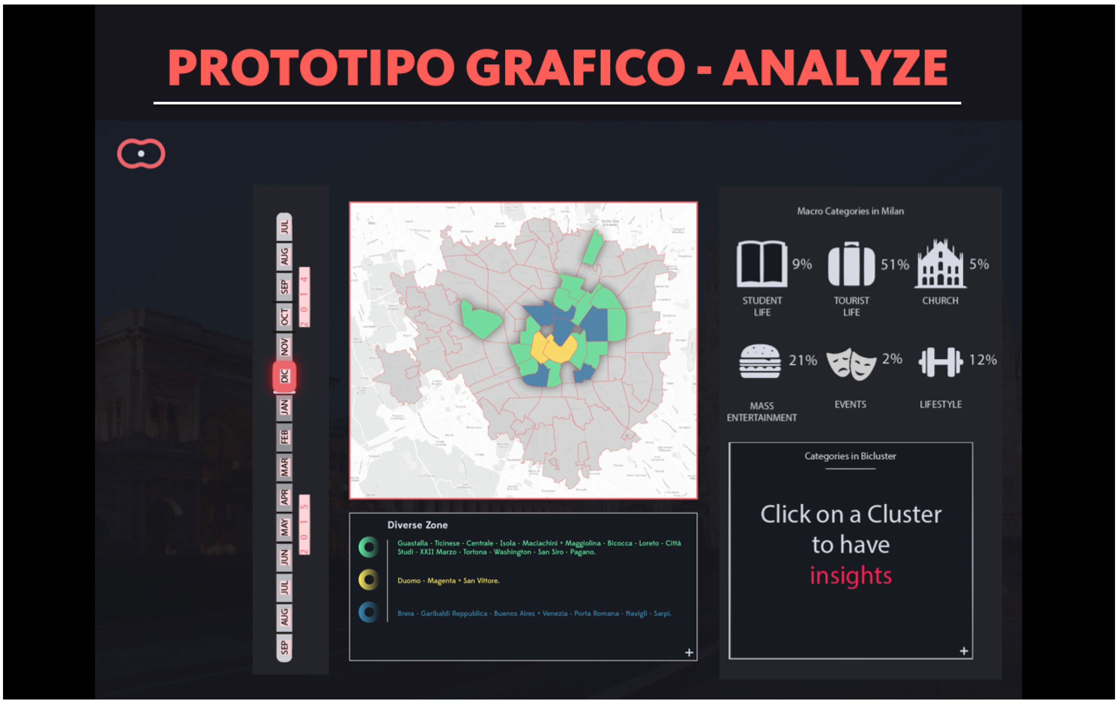


Figure 5.16: A prototype of the visualisation before selecting a month of interest

By clicking on a chosen month the graph will change, displaying all the interesting features concerning the biclusters which characterise the selected time period. Precisely the map will be refreshed presenting all the subset of NILs referring to each bicluster. In Figure 5.17 there are 4 biclusters. All the zones coloured in grey belong to one bicluster, the ones coloured in green to another bicluster and so on. Below the map there is a recap of all the NILs belonging to every bicluster. To the right of the map, instead, there are two windows. One suggests to the user how to interact with the visualisation. The other one, instead, shows all the six macro-categories. Every macro-group presents a number explaining the percentage of check-ins it has with respect to the total sum of check-ins in the city of Milan during the same time period. In Figure 5.17, for example, the 9% of the check-ins characterising Milan in the month of December 2014 was made in categories belonging to Student Life.

At this point, the user can investigate a chosen bicluster by clicking on the NILs composing it, as the bottom-left window suggests. In our case, it has been decided to show what happens after selecting the bicluster with green coloured NILs. Figure 5.18 displays how the view changes by chosing a bicluster. Precisely, the green coloured NILs stand out, the recap below the map gets refreshed and the macro-categories update. Let us remark that only the macro-categories whose at least one category belongs to the bicluster turn green, the others, instead, remain grey. This allows to have an immediate view of which categories are involved in the bicluster. To have a much detailed information about the 45 categories, it is necessary to click on the macro-category icons. In the case here presented, the Student Life macro-category has been clicked in order to show, in the apposite window, everything about the categories it contains. All the categories belonging to Student Life are listed in the bottom right window. Precisely, if a category is referring to the bicluster then it is highlighted and presents two bars. The two bars simply have the same functions of the two bars characterising the original barplot: the green bar indicates the average number of check-ins the category has in the NILs composing the bicluster, the other one, instead, is the population average of the same category. In Figure 5.18, the categories belonging to Student Life which take part to the biclusters are: Clothing Store, Art Gallery, Sandwich Place and Ice Cream Shop. For example the average number of check-ins in the category of Sandwich Place in the green coloured NILs for the month of December 2015 is 1.57. It is larger than the population mean, which is around 1.51. On the contrary, the mean of Art Gallery is lower than the population average.



Diverse Zone

Guastalla - Ticinese - Centrale - Isola - Maciachini + Maggiolina - Bicocca - Loreto - Città Studi - XXII Marzo - Tortona - Washington - San Siro - Pagano.

Duomo - Magenta + San Vittore.

Brera - Garibaldi Reppublica - Buenos Aires + Venezia - Porta Romana - Navigli - Sarpi.

Click on a Cluster to have insights

Figure 5.17: A prototype of the visualisation after selecting a chosen month

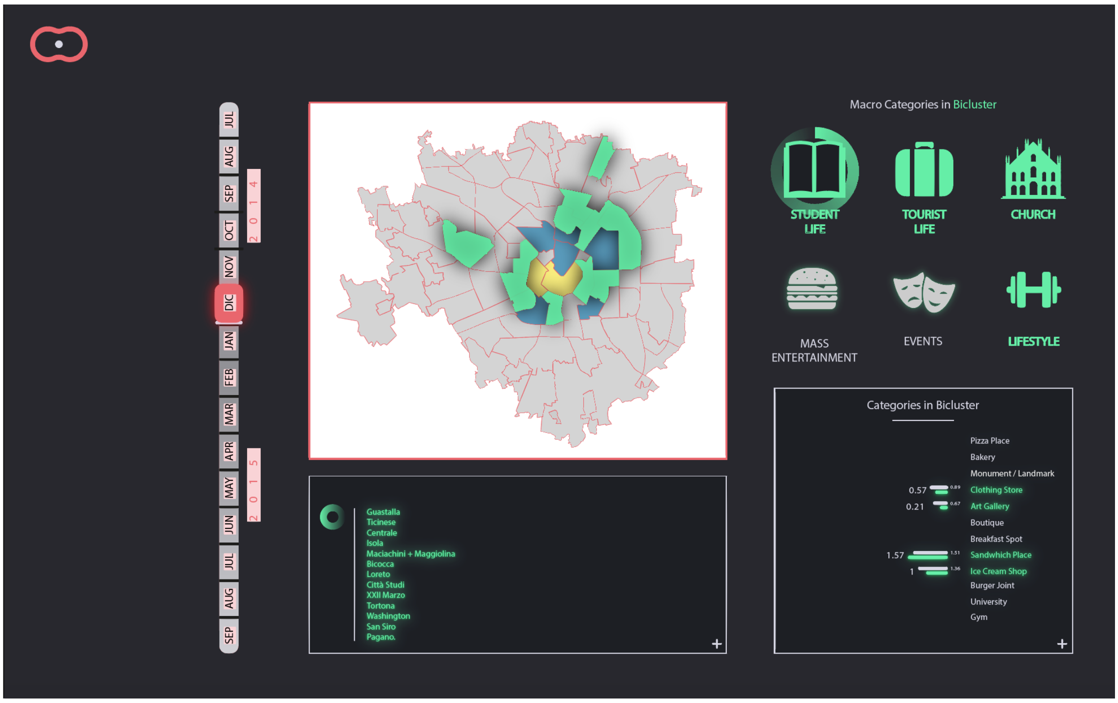


Figure 5.18: A prototype of the visualisation after selecting a chosen bicluster

Conclusions

Nowadays, cities are not mere physical and organisational structures but they can be considered as the juxtaposition of two layers: a stable material layer and a dynamical digital one. While the first one slowly changes through time, the second one evolves on a daily basis thanks to the contribution of social media users. Therefore, big data deriving from the digital layer provide valuable insights about the perception of the city. The main task of the Urbanscope project is to use these data to create and to divulge privileged views of Milan aiming at fostering comprehension and decision making. The study here proposed was set out to build a new section for the Urbanscope website, precisely the "Analyse" section of the "City magnets" module that aims at revealing which are the attractive zones in Milan, by considering Foursquare data.

Foursquare is a local search and discovery service mobile application which provides search results for its users and features a social networking system that enables a user to share their location with friends, via the "check-in". Precisely, the user would manually tell the application the position by selecting from a list of venues the application locates nearby. Every venue is categorised in groups called categories according to its function and it is geo-referenced using the names of the neighbourhoods, called NILs, in which the municipality of Milan divided the urban area.

It has been decided to perform a Biclustering algorithm in order to identify biclusters, i.e. groups of similar NILs according to subgroups of categories. Due to the fact that the application of these procedures to social data is a unicum, all the Biclustering literature being conceived on genetic problems, it was necessary to review all the possible methods in order to find the most proper one. The review of the literature leads to the choice of the Cheng and Church's Biclustering algorithm, the seminal paper of gene expression data Biclustering. However, before applying the method, the data have been preprocessed in order to reduce the elevated number of categories to a more manageable one.

Initially, the dataset was simplified by considering only the features which are useful to the analysis. Then, different Clustering approaches were used to make descriptive analysis of NILs and categories. In fact, it was possible to identify which are the most visited zones and the most popular categories during the considered time period. In addition, it was evident how the city evolves according to major events that characterise it. The main example is given by the NIL of Cascina Triulza - EXPO which became popular thanks to EXPO2015. The gained information served to guide the vectorial clustering procedure in order to reduce the number of categories from 247 to 45. In fact, the dimensional reduction was mandatory in order to create an appealing and user-friendly view of the analysis.

The last efforts were done to complete the two-sided task of the "Analysis" section of the "City magnets" module: implementation and visualisation. The two faces of the project mission demand to mix together different branches of knowledge in order to have the most self-explanatory and homogeneous representation of Milan evolution. Precisely, biclusters were identified by performing the Cheng and Church's algorithm which ended up to be the most suitable choice among all the possible algorithms. Unfortunately the outputs were too difficult to be presented in their original form and, therefore, they were used as a cornerstone to create the final visualisation. To simplify the view, it has been decided to reorganise the 45 categories in macro-categories obtained by a Clustering procedure with managerial validation. Therefore, everything was set up to create the final prototype of the section that combines interactive features, maps and barplots to show to Urbanscope users the Biclustering analysis.

However, one can notice that many future updates are possible. Firstly, the macro-categories are sometimes chaotic and they can mislead the user. Therefore, it has been considered to use another criterion to organise each category. Precisely, each category can be organised in macro-categories according to its meaning and objective function. For example everything regarding food and restaurants will belong to the same macro-group. Another important update consists in writing the "Analysis method" of the section. In that page, everything concerning the procedure that leads to the final visualisation is explained. However the explanation needs to divulge technical issues without being to difficult for the general Urbanscope user.

In addition, a compositional analysis of the data could be a nice solution to identify new clusters and biclusters. Unfortunately, the sparsity of the data makes this path hard to pursue. However, with a less sparse dataset, the compositional analysis could produce new important outputs. Thanks to the interesting characteristics of Biclustering, it could be a brilliant idea to perform Biclustering algorithms to enrich the already existing modules, as in the case of "City and the world" lens, or to create new ones.

Bibliography

Califano, A., G. Stolovitzky, Y. Tu, et al. (2000). "Analysis of gene expression microarrays for phenotype classification." In: *Ismb*. Vol. 8, pp. 75–85.

Cheng, Y. and G. M. Church (2000). "Biclustering of expression data." In: *Ismb*. Vol. 8, pp. 93–103.

Eren, K. *An Introduction to Biclustering.* `http://www.kemaleren.com/an-introduction-to-biclustering.html`. Accessed: 2016-04-02.

Getz, G., E. Levine, and E. Domany (2000). "Coupled two-way clustering analysis of gene microarray data". In: *Proceedings of the National Academy of Sciences* 97.22, pp. 12079–12084.

Hartigan, J. A. (1972). "Direct clustering of a data matrix". In: *Journal of the american statistical association* 67.337, pp. 123–129.

Kluger, Y., R. Basri, J. T. Chang, and M. Gerstein (2003). "Spectral biclustering of microarray data: coclustering genes and conditions". In: *Genome research* 13.4, pp. 703–716.

Lazzeroni, L. and A. Owen (2002). "Plaid models for gene expression data". In: *Statistica sinica*, pp. 61–86.

Madeira, S. C. and A. L. Oliveira (2004). "Biclustering algorithms for biological data analysis: a survey". In: *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)* 1.1, pp. 24–45.

Murali, T. and S. Kasif (2003). "Extracting conserved gene expression motifs from gene expression data". In: *Pacific symposium on biocomputing*. Vol. 8, pp. 77–88.

Needham, R. (1965). "Automatic classification: models and problems". In: *Mathematics and Computer Science in Biology and Medicine*, pp. 111–114.

Prelić, A., S. Bleuler, P. Zimmermann, A. Wille, P. Bühlmann, W. Gruissem, L. Hennig, L. Thiele, and E. Zitzler (2006). "A systematic comparison and evaluation of biclustering methods for gene expression data". In: *Bioinformatics* 22.9, pp. 1122–1129.

Tanay, A., R. Sharan, and R. Shamir (2002). "Discovering statistically significant biclusters in gene expression data". In: *Bioinformatics* 18.suppl 1, S136–S144.

Tang, C., L. Zhang, A. Zhang, and M. Ramanathan (2001). "Interrelated two-way clustering: an unsupervised approach for gene expression data analysis". In: *Bioinformatics and Bioengineering Conference, 2001. Proceedings of the IEEE 2nd International Symposium on.* IEEE, pp. 41–48.

Tibshirani, R., T. Hastie, M. Eisen, D. Ross, D. Botstein, P. Brown, et al. (1999). "Clustering methods for the analysis of DNA microarray data". In: *Dept. Statist., Stanford Univ., Stanford, CA, Tech. Rep.*

Wang, H., W. Wang, J. Yang, and P. S. Yu (2002). "Clustering by pattern similarity in large data sets". In: *Proceedings of the 2002 ACM SIGMOD international conference on Management of data.* ACM, pp. 394–405.

Yang, J., W. Wang, H. Wang, and P. Yu (2002). "δ-clusters: Capturing subspace correlation in a large data set". In: *Data Engineering, 2002. Proceedings. 18th International Conference on.* IEEE, pp. 517–528.

Ringraziamenti

Siamo arrivati alla conclusione di questo nostro breve lungo viaggio e, finalmente, si torna all'italico idioma a noi tanto caro giusto in tempo per i ringraziamenti. Che ci vorrà! "Dopo pagine e pagine di commenti di risultati in inglese potrò finalmente dare libero sfogo al mio estro creativo" dicevo, e invece tutto sembra essere estremamente complicato: un po' come scendere dalla vetta dopo essersi spaccati le gambe per raggiungerla. Eppure qualcosa bisogna pure scriverla e non me la posso nemmeno cavare con una pretenziosissima citazione in cui sotto sotto ringrazio il mio essere poliedrico e geniale. Seguiamo quindi un classico copione.

Partiamo ringraziando il gruppo Urbanscope, in particolare il professore Simone Vantini (oramai semplicemente Simone) sempre pronto e disponibile a indicarmi un porto sicuro a cui approdare nelle mie sudate analisi. Non posso dimenticarmi poi dei molti collaboratori che mi hanno affiancato in questi mesi, aiutandomi a trovare nuove sfaccettature e punti di vista nel mio lavoro. Grazie quindi ad ormai cervelle-in-fuga, a maestre delle categorizzazione, a gerontofobiche compagne di corso e a francofoni designer vicini di casa.

Fondamentale per tutto il mio percorso è stato soprattutto l'appoggio del mio focolare casalingo e di tutte le persone ad esso raccolte. Un abbraccio a mia madre e a mio padre, a mia nonna, a mia sorella, al mio incimurritissimo e ringhiosissimo cane e a tutti i miei parenti, che sempre, con il loro sostegno, hanno diratato la nebbia, permettendo "ch'io" non "veda soltanto la siepe dell'orto, la mura ch'ha piene le crepe di valerïane" ma molto di più.

E mentre mi riempivo gli occhi di nuovi orizzonti ho avuto la possibilità di condividerli con amici davvero speciali che, nella grande incognita su "chi va e chi resta", so già che resteranno. Facciamo la Ola per i miei chicchissimi compagni del liceo featuring nuove leve, tutti pianisti, futuri notai, giornalisti, scienziati, avvocati, filosofi, medici, veterinari, dentisti, economisti, biologi, che assieme a me combattono quotidianamente contro gli scorni che il bituminoso "demonie" ci manda contro.

Un abbaio gioioso ai miei coinquilini vecchi e nuovi con i quali ho mangiato pizze, commentato programmi trash in tv e twerkato contro armadi. Un sorriso abbagliante ai miei amici di questa vita da Ing. Mat, fatta di sabati a studiare a Baghdad, teoremi, scambi di Pokémon, concerti con svenimenti e popcorn davanti a Xfactor. Il mio ringraziamento va soprattutto a coloro che mi hanno pagato il MAV, sopportato nei miei momenti di crazyness, sedato, insegnato, accolto e, soprattutto, deciso di procedere al mio fianco tra gli uomini che non si voltano.

Grazie a tutti di cuore.

Grazie a tutti per aver acceso i lumi dei vostri porti anche se “me al largo sospinge ancora il non domato spirito, e della vita il doloroso amore.”