Politecnico di Milano

School of Industrial and Information Engineering

Master of Science in Telecommunication Engineering

Characterizing signatures of mobile traffic via non-matrix factorization based model

Supervisor:
Prof. GUIDO ALBERTO MAIER
Co-Supervisor:
SEBASTIAN TROIA

Candidate: MOSTAFA ELFAZARI
Matricola: 872200

Contents

List of Figures

Abstract

We are moving into an era when telecommunication data analysis becomes extremely important. The mobile traffic data collected in urban areas tends to have repetitive patterns with spatio-temporal variations. Analyzing the relationship between the repetitive patterns of traffic and the urban area can play a vital role in traffic engineering, network design and urban planning.

In this work, we investigated the research field of pattern recognition to derive an effective method to extract signatures from the mobile traffic data-set, then to cluster the signatures based on their profiles.

The signature proposed in this work shows enhancement in clustering mobile traffic, in comparison with the state of the art methods. The signature extraction was based on non-negative matrix factorization, where we factorize the dimension of two months of mobile traffic in 24 hours that reflects working days and weekends. By applying clustering based on our model, we obtained better coverage and entropy of land-usage inside the classes that represent a specific traffic profile behaviour.

Introduction

The aim of this thesis is to analyse mobile traffic in urban areas, to find correlations between the traffic profiles and the landusage in the city. Due to the repetitive behaviour in the traffic(the so-called tidal effect), we can discover groups of similar traffic signatures within the mobile network. That correlation in traffic signatures can help in both engineering and urban planning. Signatures represent the level of interaction of the users with the mobile network and the urban area.

Signature is defined as the typical traffic profile that recurs in a specific area of the mobile network in the city.

The goal is to propose a new method to extract signatures from the mobile traffic data-set, then to cluster the signatures based on their profiles, which are representative of distinct types of traffic associated with human activities in the urban area.

The data-set we are using in this work is the result of a computation over the Call Detail Records (CDR) generated by the Telecom Italia cellular network over the city of Milan.The CDRs log the user activity for billing purposes and network management. As described more in details in chapter three, the CDR dataset contains the following records

- Received SMS: a CDR is generated each time a user receives an SMS
- Sent SMS: a CDR is generated each time a user sends an SMS
- Incoming Calls: a CDR is generated each time a user receives a call
- Outgoing Calls: CDR is generated each time a user issues a call
- Internet: a CDR is generated each time a user starts an internet connection or a user ends an internet connection

The method we adopted to obtain the traffic signature is based on non-negative matrix factorization (NMF), where we factorized CDR Matrix in order to reduce it into constituent parts that make it easier to extract the signatures. We called the signature obtained from the NMF method, a non-negative matrix factorization based signature (NMFS). We have applied NMFS along with other state-of-the-art methods, to assess NMFS effectiveness.

In order to validate the clustering, we made evaluation using the ground-truth data-set. The ground-truth gives the geolocalized characteristic information known as land-use for each area of Milan, such as: residential, office, transportation, touristic, university, shopping and nightlife.

Finally, the results of the clustering obtained with different methods were quantitatively compared using that land-use information. We were able to estimate the density of each land-use category, for example, university or office, inside all clusters. Another evaluation metric was to find the coverage, which is defined as the percentage of each land-use category included within those clusters. Then we obtained the entropy, which is the uncertainty of a specific land-use category within a specific cluster, the lower is the entropy, the higher is the precision of clustering. The last evaluation was the F-score index which allows determining a single final score, by combining entropy and coverage. The F-score index ranges in [0, 1], with 1 indicating the best performance achievable by the given cluster set, with respect to land-use.

By comparing NMFS with other methods available in the literature on the basis of the performance parameters introduced above, we were able to show that the NMFS has several advantages.

0.1 Thesis structure

After the introduction, in the first chapter, we will explain the basic knowledge needed to understand this thesis. Starting from what is machine learning and the branches of machine learning. Then we will focus on explaining the unsupervised machine learning and why we used it. Afterward, we will explain the clustering algorithm and we will focus on K-means. we will explain the state of the art methods that have been proposed for signatures characterization for mobile network traffic.

The second chapter will explain the core of the thesis, i.e. the data-set used in this thesis. We have two data-sets the first is CDR, consisting of telecommunication activity records in the city of Milan. In our study, we focused on mobile traffic data in the period of November, December 2013, where the temporal unit is 1-hour interval. The city of Milan was divided into a 100x100 square grid each square 550 km^2 with a side length of 235 m in figure 2.2, this is the areal unit we use throughout the thesis, and we refer to it as a “square”.

The second data set is the ground truth. it has been obtained by mapping the geometry from "Population distribution data in Italy" to Milan grid geometry. Where we have the land-use categorized as: residential, office, transportation, touristic, university, shopping and nightlife per each area.

Finally we will explain the NMFS, the method used for the dimensional reduction of dataset and feature extraction. This method was used to extract significant traffic signature using Non -negative matrix factorization. The motivation is visualization, compressing the data, and finding a representation that is more informative for further processing.

The third chapter is dedicated to apply the state of the art and NMFS on clustering algorithms and obtaining the clusters. Then performing the evaluation methods between the clustered classes and the ground-truth data, have some insights about the efficiency of the signatures clustering.

The fourth chapter is the conclusion about this thesis and the proposed signature characterization method NMFS. Then, in the end, we will talk about future work in order to enhance signature characterization.

Chapter 1

Background knowledge

In this chapter, we will explain the basic knowledge needed to understand this thesis, starting from what's machine learning and the branches of machine learning. Then we will focus on explaining the unsupervised machine learning and why we used it. Then we will explain the clustering algorithm and we will focus on K-means. Then we will explain the state of the art methods that have been proposed for signatures characterization for mobile network traffic.

1.1 Machine learning

Machine learning is a subfield of computer science that evolved from the study of pattern recognition and computational learning theory in artificial intelligence. In 1959, Arthur Samuel defined machine learning as a Field of study that gives computers the ability to learn without being explicitly programmed. Machine learning explores algorithms that can learn from and make predictions on data. Such algorithms operate by building a model to make predictions or decisions, rather than following strictly static program instructions. Machine learning tasks are typically classified into three different categories :

1. Supervised learning

   A computer is presented with inputs and their desired outputs, and the goal is to learn a general rule that maps inputs to outputs, and able to perform predictions.

2. Unsupervised learning

No outputs are given to the learning algorithm, leaving it on its own to find structure in its input. Unsupervised learning can be a goal in itself, as discovering hidden patterns in data.

3. Reinforcement learning

   A computer program interacts with a dynamic environment in which it must perform a certain goal (such as driving a vehicle), without being programmed to do so. Another example is learning to play a game facing an opponent.

1.2 Clustering techniques

Clustering is the grouping of a particular set of objects based on their characteristics, aggregating them according to their similarities.

There are several different ways to implement clustering, based on different models. Different algorithms are applied to each model, differentiating its properties and results. These models are distinguished by their organization and type of relationship between them. The most important ones are:

- Connectivity models: As the name suggests, these models are based on the notion that the data points closer in data space exhibit more similarity to each other than the data points lying farther away. These models can follow two approaches. In the first approach, they start with classifying all data points into separate clusters then aggregating them as the distance decreases. In the second approach, all data points are classified as a single cluster and then partitioned as the distance increases. Also, the choice of distance function is subjective. These models are very easy to interpret but lack scalability for handling big datasets. Examples of these models are hierarchical clustering algorithm and its variants.

- Centroid models: These are iterative clustering algorithms in which the notion of similarity is derived by the closeness of a data point to the centroid of the clusters. K-Means clustering algorithm is a popular algorithm that falls into this category. In these models, the no. of clusters required at the end have to be mentioned beforehand, which makes it important to have prior knowledge of the dataset. These models run iteratively to find the local optima.

- Distribution models: These clustering models are based on the notion of how probable is it that all data points in the cluster belong to the same distribution (For example: Normal, Gaussian). These models often suffer from overfitting. A popular example of these models is the Expectation-maximization algorithm which uses multivariate normal distributions.

- Density Models: These models search the data space for areas of varied density of data points in the data space. It isolates various different density regions and assign the data points within these regions in the same cluster. Popular examples of density models are DBSCAN and OPTICS.

1.2.1 K-Means Clustering

The k-means algorithm searches for a predetermined number of clusters within an unlabeled multidimensional dataset. It accomplishes this using a simple conception of what the optimal clustering looks like:

- The "cluster center" is the arithmetic mean of all the points belonging to the cluster.

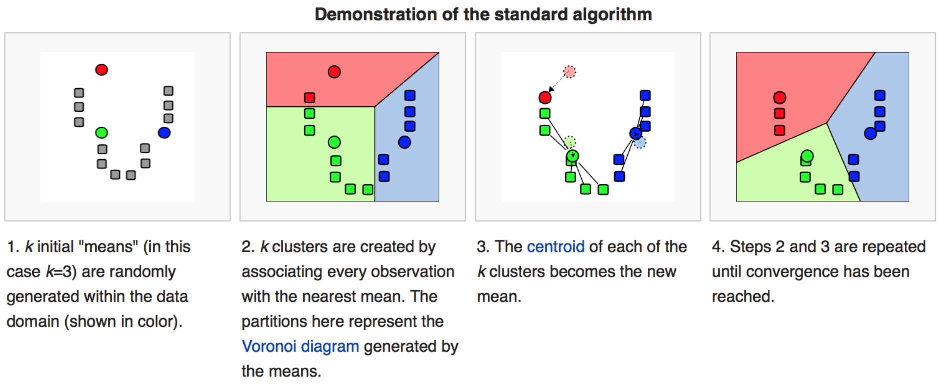
- Each point is closer to its own cluster center than to other cluster centers.

To run a k-means algorithm, you have to randomly initialize let's take for example in figure 1.1. let's say we have three cluster centroids because we want to group the data into three clusters. K-means is an iterative algorithm and it does two steps: 1. Cluster assignment step 2. Move the centroid step.

In Cluster assignment step, the algorithm goes through each of the data points and depending on which cluster is closer, whether the first cluster centroid or the second cluster centroid or the third. It assigns the data points to one of the three cluster centroids. In the move centroid step, K-means moves the centroids to the average of the points in a cluster. In other words, the algorithm calculates the average of all the points in a cluster and moves the centroid to that average location.

This process is repeated until there is no change in the clusters (or possibly until some other stopping condition is met). K is chosen randomly or by giving specific initial starting points by the user.

K-means is usually run many times, starting with different random centroids each time. The results can be compared by examining the clusters or by



Demonstration of the standard algorithm

1. *k* initial "means" (in this case *k*=3) are randomly generated within the data domain (shown in color).

2. *k* clusters are created by associating every observation with the nearest mean. The partitions here represent the Voronoi diagram generated by the means.

3. The centroid of each of the *k* clusters becomes the new mean.

4. Steps 2 and 3 are repeated until convergence has been reached.

Figure 1.1: Kmeans algorithm

a numeric measure such as the clusters' distortion, which is the sum of the squared differences between each data point and its corresponding centroid. In cluster distortion case, the clustering with the lowest distortion value can be chosen as the best clustering.

The K-means algorithm defined above aims at minimizing an objective function, which in this case is the squared error function.

$$J=\sum_{i=1}^{k}\sum_{j=1}^{n}||X_i - V_j||^2 \qquad (1.1)$$

$||X_i - V_j||$ is the Euclidean distance between a point,X_i and a centroid V_j, iterated over all k points in the i^{th} cluster, for all n clusters.

1.2.2 Hierarchical clustering

Hierarchical clustering, an algorithm that builds a hierarchy of clusters. This algorithm starts with all the data points assigned to a cluster of their own. Then two nearest clusters are merged into the same cluster. In the end, this algorithm terminates when there is only a single cluster left.

The results of hierarchical clustering can be shown using dendrogram

At the bottom in figure 1.2, we start with 25 data points, each assigned to separate clusters. Two closest clusters are then merged till we have just one

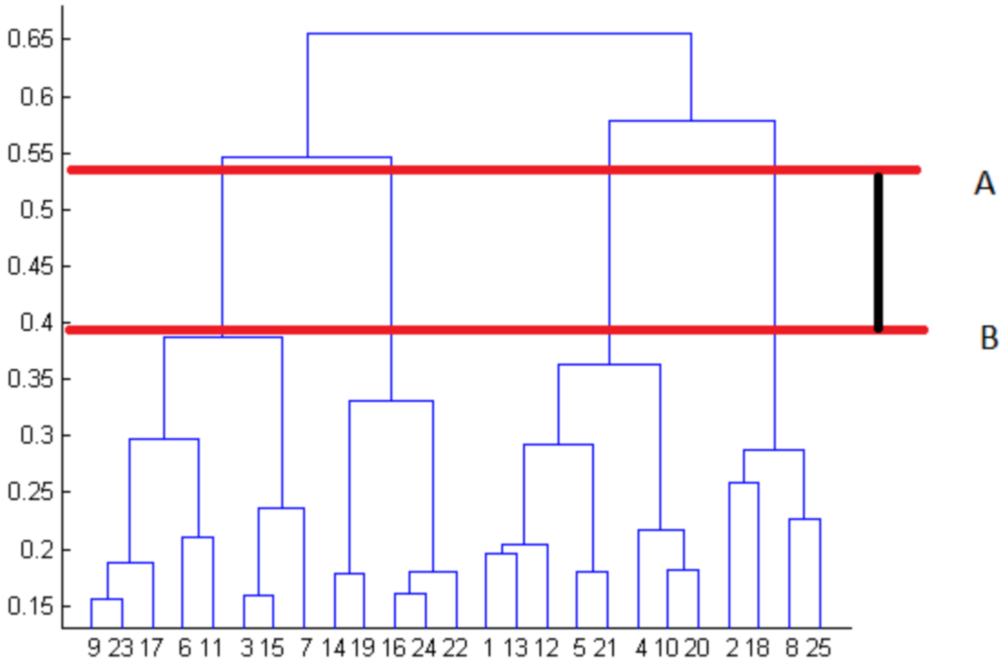


Figure 1.2: Dendrogram

cluster at the top. The height in the dendrogram at which two clusters are merged represents the distance between two clusters in the data space.

The decision of the number of clusters can be chosen by observing the dendrogram. The best choice of the number of clusters is the no. of vertical lines in the dendrogram cut by a horizontal line that can transverse the maximum distance vertically without intersecting a cluster.

In the above example, the best choice of no. of clusters will be 4 as the red horizontal line in the dendrogram below covers maximum vertical distance AB.

1.2.3 Gaussian Mixture Models

The k-means clustering model explored is simple and relatively easy to understand, but its simplicity leads to practical challenges in its application. In particular, the nonprobabilistic nature of k-means and its use of simple distance-from-cluster-center to assign cluster membership leads to poor performance for many real-world situations. The Gaussian mixture models can be viewed as an extension of the ideas behind k-means, but can also, be a powerful tool for estimation beyond simple clustering.

A Gaussian mixture model (GMM) attempts to find a mixture of multidimensional Gaussian probability distributions that best model any input dataset. In the simplest case, GMMs can be used for finding clusters in the same manner as k-means.

Gaussian mixture model is very similar to k-means: it uses an expectation-maximization approach which qualitatively does the following:

- Choose starting guesses for the location and shape
- Repeat until converged:

    1. E-step: for each point, find weights encoding the probability of membership in each cluster
    2. M-step: for each cluster, update its location, normalization, and shape based on all data points, making use of the weights

The result of this is that each cluster is associated not with a hard-edged sphere, but with a smooth Gaussian model. Just as in the k-means expectation-maximization approach, this algorithm can sometimes miss the globally optimal solution, and thus in practice, multiple random initializations are used.

1.3 Number of clusters initialization

we will mention some Proposed Measures that were mentioned in literature and we used to obtain the initial number of clusters.

Since the k-means method aims to minimize the sum of squared distances from all points to their cluster centers, this should result in compact clusters. We can, therefore, use the distances of the points from their cluster center to determine whether the clusters are compact. For this purpose, we use the intra-cluster distance measure, which is simply the distance between a point and its cluster center and we take the average of all of these distances, defined as

$$intra = \frac{1}{N} \sum_{i=1}^{K} \sum_{x \epsilon C_i} |x - z_i|^2 \tag{1.2}$$

where N is the number of signatures in CDR data-set, K is the number of clusters, and z_i is the cluster center of cluster C_i. We obviously want to minimize this measure. We can also measure the inter-cluster distance, or the distance between clusters, which we want to be as big as possible. We calculate this as the distance between cluster centres and take the minimum of this value, defined as

$$inter = min(|z_i - z_j|^2) \tag{1.3}$$

We take only the minimum of this value as we want the smallest of this distance to be maximized, and the other larger values will automatically be bigger than this value.

Since we want both of these measures to help us determine if we have a good clustering, we must combine them in some way. The obvious way is to take the ratio, defined as:

$$Validity = \frac{intra}{inter} \tag{1.4}$$

Since we want to minimize the intra-cluster distance and this measure is in the numerator, we consequently want to minimize the validity measure. We also want to maximize the intercluster distance measure, and since this is in the denominator, we again want to minimize the validity measure. Therefore, the clustering which gives a minimum value for the validity measure will tell us what the ideal value of K is in the k-means procedure.

1.3.1 Calinski Harabasz (CH)

Called sometimes variance ratio criterion (VRC), evaluates the cluster validity based on the average between and within-cluster sum of squares. Well defined clusters have a large between-cluster variance and a small within-cluster variance.

The score is higher when clusters are dense and well separated, which relates to a standard concept of a cluster.

1.3.2 Davies-Bouldin (DB)

Davies Bouldin is based on a ratio of within-cluster and between cluster distances. For each cluster C, the similarities between C and all the other clusters are computed, and the highest value is assigned to C as its cluster similarity. The index is obtained by averaging all clusters of similarities. So, we are looking for the smallest index.

1.3.3 Silhouette Coefficient

The Silhouette Coefficient is calculated using the mean intra-cluster distance (a) and the mean nearest-cluster distance (b) for each sample. The Silhouette Coefficient for a sample is (b - a) / max(a, b). To clarify, b is the distance between a sample and the nearest cluster that the sample is not a part of. Note that Silhouette Coefficient is only defined if number of labels is 2 $\leq Nlabels \leq Nsamples - 1$.

The best value is 1 and the worst value is -1. Values near 0 indicate overlapping clusters. Negative values generally indicate that a sample has been assigned to the wrong cluster, as a different cluster is more similar.

1.4 Signature characterization

Since the tidal traffic has repetitive behavior for the activities of the mobile network, we can extract a signature that represents the mobile traffic behavior in a compact form called mobile traffic signature. The mobile traffic signature is defined as the typical traffic profile that recurs in a specific area of the mobile network.

we will explain the methods used in literature [15] [3] [2] to obtain signatures mobile traffic data in an urban area.

Our study has been done using mobile traffic data provided by Telecom Italia that will be explain in detail in chapter three. The data-set is the result of a computation over the Call Detail Records (CDR) generated by the Telecom Italia cellular network over the city of Milano. In next section we will show the results of the state of the art signatures based on this data-set.

1.4.1 Soto

The Soto approach [15] considers mobile traffic signatures that correspond to the average mobile traffic volume observed during (1) a working day, and (2) a weekend day. We refer to these as average weekday-weekend signatures. Formally, the set of days d is split into two sets d_{wd} and d_{we}, which contains all Mondays-to-Fridays, and all Saturdays and Sundays, respectively. Then, the element associated to t in the signature of a unit area is :

$$S_a(wd, d) = \frac{1}{d_{wd}} \sum_{d \epsilon d_{wd}} v_a(d, t) \tag{1.5}$$

for time slots t during working days, and

$$S_a(we, d) = \frac{1}{d_{we}} \sum_{d \epsilon d_{we}} v_a(d, t) \tag{1.6}$$

for time slots t during weekends. The signature of a is then

$$S_a = \|_{d \epsilon \boldsymbol{d'}} \; (\|_{t \epsilon \boldsymbol{t}} \; S_a(d, t)) \tag{1.7}$$

In 1.7 $\boldsymbol{d'}$ is the condensed set of days, which, in the case of Soto approach is $\boldsymbol{d'} = (wd, we)$. Also, k indicates the concatenation of all elements in a set: sa is thus the concatenation of all elements referring to the average working day and to the average weekend day. Signatures then go through a standard score normalization phase, where each time slot signature obtained in and is normalized with respect to the mean and standard deviation of all those referring to the same unit area. Formally, for the signature element of unit area at time slot t

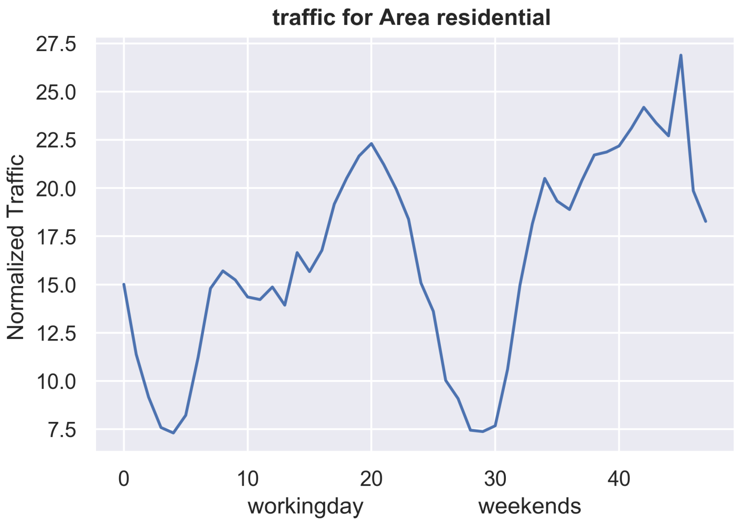


Figure 1.3: Signature of Soto

$$\hat{S}_a(we, d) = \frac{S_a(we, d) - \mu(S_a)}{\sigma(S_a)} \tag{1.8}$$

where $d\epsilon\boldsymbol{d'} = (wd, we)$, whereas $\mu(S_a)$ and $\sigma(S_a)$ denote the mean and standard deviation of the set of elements concatenated in the signature sa. Then, the normalized signature $\hat{S_a}$ is simply obtained by concatenation of $\hat{S}_a(d, t)$ for all $d\epsilon d'$ and $t\epsilon\boldsymbol{t}$, as in 1.8 .

As far as distances between signatures are concerned, Soto considers the Euclidean distance between the corresponding ordered vectors. Given the signatures of two unit areas a and b, their distance is

$$d_{ab} = \sqrt{\sum_{d\epsilon\boldsymbol{d'}} \sum_{t\epsilon\boldsymbol{t}} (\hat{S}_a(d,t) - \hat{S}_b(d,t))^2} \tag{1.9}$$

Finally, the clustering of signatures is performed in Soto by running a k-means algorithm over the set of $\hat{s_a}, a\epsilon\boldsymbol{a}$ using 1.9 as the k-means distance measure. The algorithm requires the parametrization of k, the desired number of clusters: in Soto, k is selected according to the validity index proposed in [13] and the metrics that were mentioned in 2.6. In their considered dataset, the best results are obtained with k=8 that also equal to the number of the landuse categories.

The figure 1.3, represents the signature obtained by applying Soto to the CDR data-set.

1.4.2 Cici

Cici applies a Fast Fourier Transform (FFT) to the signature in figure 2.1, so as to clean it from irregular patterns. Specifically, once converted to the frequency domain with FFT, only the highest power frequencies are kept, and the time signal is reconstructed with an Inverse FFT (IFFT) from the selected frequencies. The filtering returns the Seasonal Communication Series (SCS) of the original signature. Normalization of whole time- series SCS-filtered signatures are then performed using the standard-score approach in. However, in the case of Cici, $d\epsilon\boldsymbol{d}$, since signatures do not condense days, but include the full-time series [3].

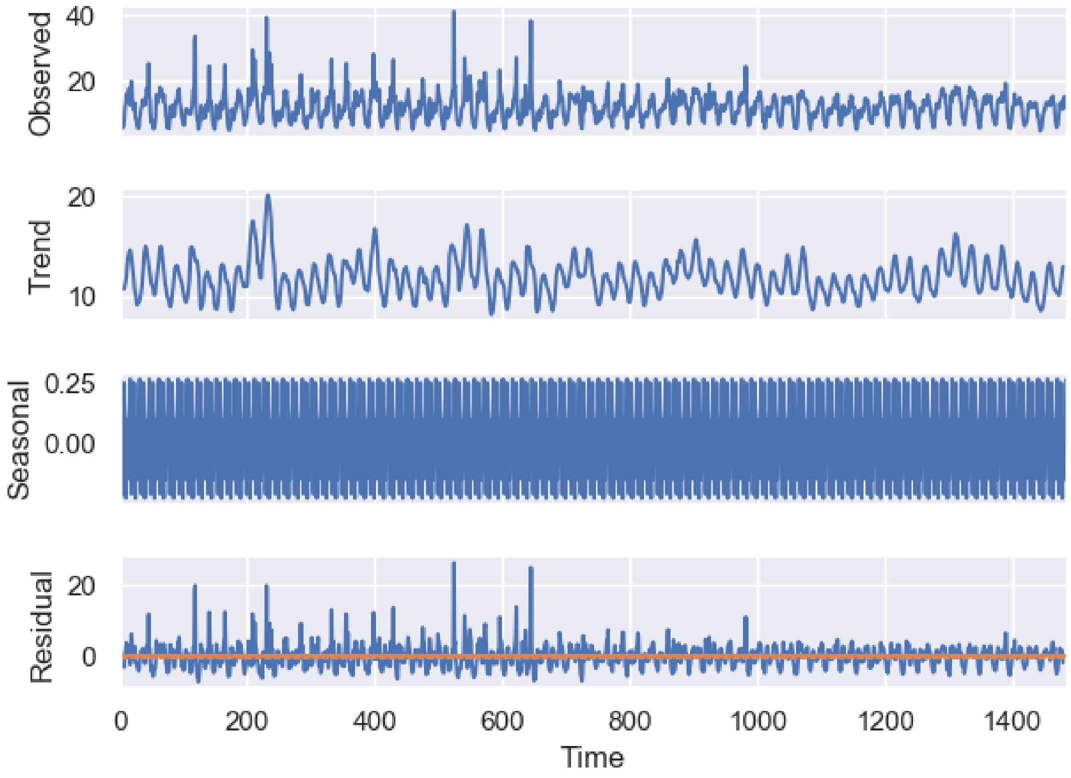


Figure 1.4: Decomposition of the CDR time series data

The Cici solution in [3] considers a whole-time-series signature for each unit area. In other words, the signature of the area is

$$S_a = \|_{d\epsilon \boldsymbol{d'}} \left(\|_{t\epsilon \boldsymbol{t}} S_a(d,t)\right) \tag{1.10}$$

and the number of elements that compose it is not bounded, but depends on the timespan of the dataset D. In addition,Time series decomposition involves thinking of a series as a combination of level, trend, seasonality, and noise components.

The figure 1.4, represents the time series decomposition considered in Cici, applied to the CDR data-set. We get rid of residual traffic that represent noise and also the seasonality in traffic.

1.4.2.1 Time Series decomposition

Real-world data is messy and noisy. There may be additive and multiplicative components. There may be an increasing trend followed by a decreasing trend. There may be non-repeating cycles mixed in with the repeating seasonality components.

There are two components within the time series data.

- Systematic

  Components of the time series that have consistency or recurrence and can be described and modeled.

- Non-Systematic

  Components of the time series that cannot be directly modeled.

A given time series is thought to consist of three systematic components including level, trend, seasonality, and one non-systematic component called noise.
These components are defined as follows:

1. Level: The average value in the series.

2. Trend: The increasing or decreasing value in the series.

3. Seasonality: The repeating short-term cycle in the series.

4. Noise: The random variation in the series.

1.4.3 Median week signature (MWS)

The Soto approach is based on a signature definition that is very compact but omits too much information when confronted to the original data [15]. On the other hand, the signature employed in Cici has a number of elements equal to that of the original time series, and, intuitively, is a loss-less representation of the same. However, when considering months of mobile traffic activity, clustering Cici signatures incurs into the well-known curse of dimensionality [3], and is in all cases very expensive from a computational standpoint.

In [2] and [1] a novel signature model has been proposed that aims at combining the advantages of Soto and Cici signatures while overcoming their limitations. the median week signature (MWS) is based on two considerations.

- First, it has been repeatedly shown that there exists a strong weekly periodicity in human occupations [8]- [6], which implies that most of the diversity in mobile traffic activity occurs within a one-week period. We thus speculate that a signature describing the typical weekly behavior of the mobile demand at one unit area contains the vast majority of the significant information about the nature of that area. This allows defining a compact, week-long signature that avoids the dimensional problems of the Cici model, and does not lose important knowledge as in Soto.
- Second, we deem the median to be a more reliable statistical measure than others used in Soto or Cici (e.g., the average or the absolute values), when it comes to assessing the typical activity in mobile traffic. As a matter of fact, the median is much more robust to outliers, which are frequent in mobile traffic due to special events of social, political, sports, or cultural nature [11]- [5].

The MWS is computed according to these guidelines, as follows. The whole set of days d is divided into seven sets, each containing elements of the dataset D that refer to one day of the week, from Monday to Sunday. In other words, $d_{mon} \cup d_{tue} \cup d_{wed} \cup d_{thu} \cup d_{fri} \cup d_{sat} \cup d_{sun} = d$ Then, the element associated to time slot t in the signature of unit area a is

$$S_a(mon, t) = \mu_{1/2}(\{V_a(d, t) | d \epsilon d_{mon}\}) \tag{1.11}$$

for time slots t corresponding to Mondays, and equivalently for all other days. In 1.11, $\mu_{1/2}$ represents the median of the set within parenthesis.

Then, the MWS is defined as the concatenation in 1.7, where

$d' = \{mon, tue, wed, thu, fri, sat, sun\}$is the condensed set of days. Taking the MWS model as a pivot.

First, we assess the impact of SCS filtering, proposed in [3] by considering both the case where the MWS is passed through the FFT/IFFT and the case where MWS is used as-is.

Second, we evaluate two different techniques to normalize MWS. One option is the standard score normalization introduced above; in this case, MWS are normalized according to 1.8, where $d' = \{mon, tue, wed, thu, fri, sat, sun\}$. The other option is daily normalization, where the signature element of unit area a at time slot t

$$\hat{S_a}(d,t) = \frac{S_a(d,t)}{\sum_{t\epsilon\boldsymbol{t}} S_a(d,t)}$$
(1.12)

where again $d\epsilon\boldsymbol{d'} = \{mon, tue, wed, thu, fri, sat, sun\}$. Thus, daily normalization normalizes each element with respect to the total activity during the weekday the element belongs to.

Third, we combine MWS with both distance measures used in Soto and Cici, i.e., the Euclidean distance in 1.9 and the distance based on the Pearson correlation coefficient; in both cases $d\epsilon\boldsymbol{d'} = \{mon, tue, wed, thu, fri, sat, sun\}$.

Finally, signature clustering is performed as in Cici, using the agglomerative hierarchical algorithm.

1.4.4 Non-negative matrix factorization(NMF)

Non-Negative Matrix Factorization (NMF) is a set of algorithms in multivariate analysis and linear algebra and also one of the unsupervised machine

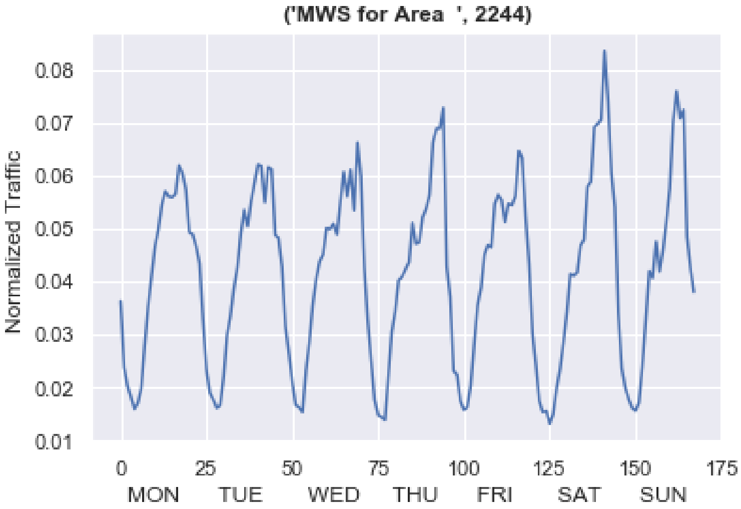


Figure 1.5: Median week signature

learning algorithm that which we take in our account as a way to obtain the signatures. NMF aims to extract useful features from the data and perform dimension reduction, here a matrix X is factorized into two matrices W and H, with non negative elements figure 1.6. The non-negativity is a useful constraint for matrix factorization that can learn a partial representation of the data. The basic idea is to divide the matrix of observations X in a product of two matrices:

Now, the W is composed of m rows $w_1, w_2, ...w_m$ and k rows $w_1, w_2, ...w_k$ which represents the features, H is composed of m rows $h_1, h_2, ...h_m$ which represents the weights.

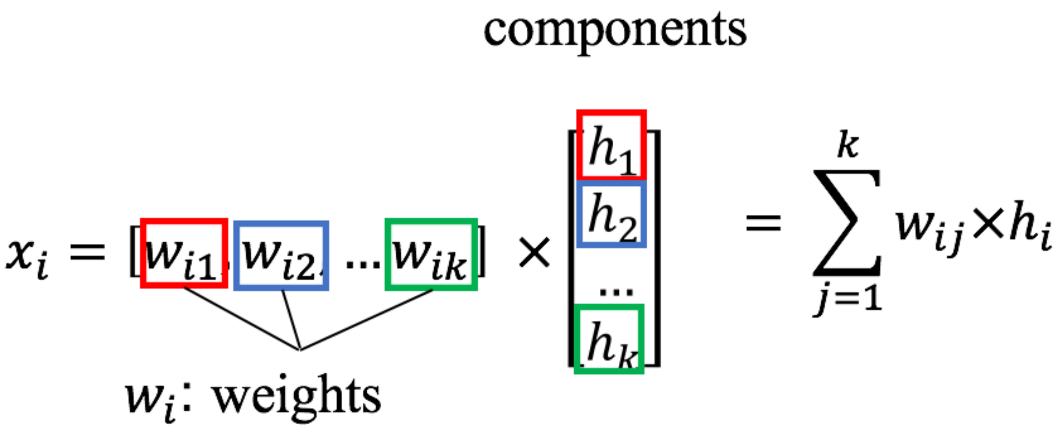

components
$x_i = [w_{i1}, w_{i2}, \dots w_{ik}] \times$
h_1
h_2
...
h_k
$= \sum_{j=1}^{k} w_{ij} \times h_i$
w_i: weights

Figure 1.6: NMF concept

To find an approximate factorization $X \approx WH$, we first need to define an objective function that quantifies the quality of the approximation. Such a the objective function can be constructed using a measure of the distance between two non-negative matrices A and B. We can use the square of the Euclidean distance:

$$||A - B||^2 = \sum_{ij} (A_{ij} - B_{ij})^2 \tag{1.13}$$

The objective function is to minimize $||V - WH||^2$ with respect to W and H, subject to the constraints W,H ≥ 0.

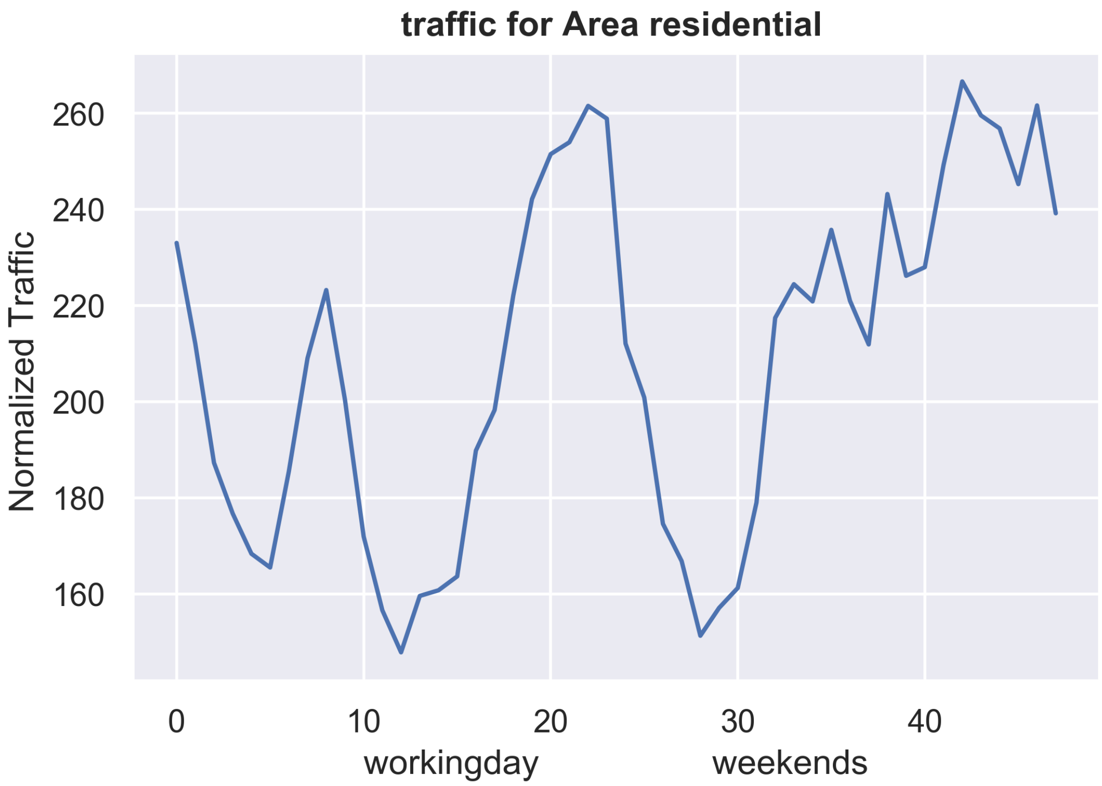


Figure 1.7: Signature based on NMF

There are plenty of techniques for numerical optimization that can be applied to find local minima. One of these is the gradient descent, but convergence can be slow. Other methods such as conjugate gradient have faster convergence, at least in the vicinity of local minima, but are more complicated to implement rather than gradient descent. The convergence of gradient-based methods also has the disadvantage of being very sensitive to the choice of step size, which can be very inconvenient for large applications.

We applied Coordinate descent from Sklearn [12], which is an optimization algorithm that successively minimizes along coordinates directions to find the minimum of a function. Coordinate descent updates one parameter at a time, while gradient descent attempts to update all parameters at once. In Coordinate descent, we minimize one coordinate of the w vector at a time while keeping all others fixed. while gradient descent attempts to update all parameters at once.

In coordinate descent, There is no step size hyper-parameter. A hyperparameter is a parameter whose value is used to control the learning process.

The figure 1.7, represents the signature obtained by applying NMF to the CDR data-set. We will explian in details in section 4.3 the characterization of NMFS signature.

1.5 Evaluation metrics

In order to measure the quality of clustering, We evaluate the quality of the unit area classification with respect to the available ground-truth data according to the following metrics density, coverage, entropy and F-score as mentioned in [2] [1].

1.5.1 Density

The density $D_G(\boldsymbol{c}, c)$ is a measure of the frequency of ground-truth elements of a given class G within a cluster $c \epsilon \boldsymbol{c}$, where $\boldsymbol{c}$ is the set of clusters determined by the current urban fabric detection approach. Let us define as k_G the set of elements of class G (e.g., the set of universities) in the ground-truth data; also, $1_c(k)$ is an indicator function that is one if a ground-truth element kk_G ends up in unit areas belonging to cluster c, and zero otherwise. Formally, the density is then defined, for a given clustering c as

$$D_G(\boldsymbol{c}, c) = \frac{1}{|c|} \sum_{k \epsilon k_G} 1_c(k) \tag{1.14}$$

where $|c|$ denotes the size of the cluster $c \epsilon \boldsymbol{c}$, i.e., the number of unit areas it includes. The density allows comparing different clusters for the same class G, so as to understand in which clusters elements of G are more frequent.

1.5.2 Entropy

The entropy $H_G(c)$ associated to a ground-truth class G for a given clustering $\boldsymbol{c}$ allows estimating the dispersion of G across the clusters defined by $\boldsymbol{c}$ It is defined as:

$$H_G(\boldsymbol{c}) = -\sum_{c \epsilon \boldsymbol{c}} P_G(c) \log(P_G(c)) \tag{1.15}$$

In (1.5.3), $P_G(c)$ is the probability that a ground-truth element of class G falls into cluster c, i.e.,

$$P_G(c) = \frac{1}{K_G} \sum_{k \epsilon k_G} 1_c(k) \tag{1.16}$$

Lower entropy is thus an indicator of a less random, i.e., more precise, assignment of ground-truth data of a given class to clusters defined by the detection strategy.

1.5.3 Coverage

The coverage $C_G\boldsymbol{c})$ of G-class elements for a clustering $\boldsymbol{c}$ is defined as the percentage of groundtruth elements of class G included within those clusters of $\boldsymbol{c}$ that are the most relevant to G. Specifically, let us define a subset of clusters $C_G \subseteq \boldsymbol{c}$ that have higher than-average density for ground-truth class G, Then, the coverage is defined as:

$\mathrm{C}_G(\boldsymbol{c}) = \sum_{c\epsilon\boldsymbol{c}} P_G(c)$
(1.17)

Higher coverage indicates that the ground-truth data for a class G is better matched by those clusters that are deemed considered meaningful for G.

1.5.4 F-score

The F-score index allows determining a single, final score to each detection technique, by combining entropy and coverage for each class G, as follows

$$\hat{F}_G(\boldsymbol{c}) = \frac{1 - \hat{H}_G(\boldsymbol{c}) * C_G(\boldsymbol{c})}{1 - \hat{H}_G(\boldsymbol{c}) + C_G(\boldsymbol{c})} \tag{1.18}$$

where $\hat{H}_G(\boldsymbol{c}) = \frac{H_G(\boldsymbol{c})}{\log(|\boldsymbol{c}|)}$ is the normalized Shannon entropy.

The F-score index ranges in [0, 1], with 1 indicating the best performance achievable by the given cluster set, with respect to ground-truth class G.

Chapter 2

Mobile metro traffic analysis

In this chapter, we will explain the mobile metro traffic analysis used to develop this thesis. Explaining the data-sets used to perform the traffic analysis. Introduce the adopted method based on NMF to obtain the traffic signature.

The aim of this thesis is to analyse mobile traffic in urban areas, to find correlations between the traffic profiles and the land-usage in the city. Due to the repetitive behaviour in the traffic so-called tidal effect, we can discover groups of similar traffic signatures within the mobile network. That correlation in traffic signatures can help in both engineering and urban planning. Signatures represent the level of interaction of the users with the mobile network and the urban area.

2.1 The Big Data Challenge

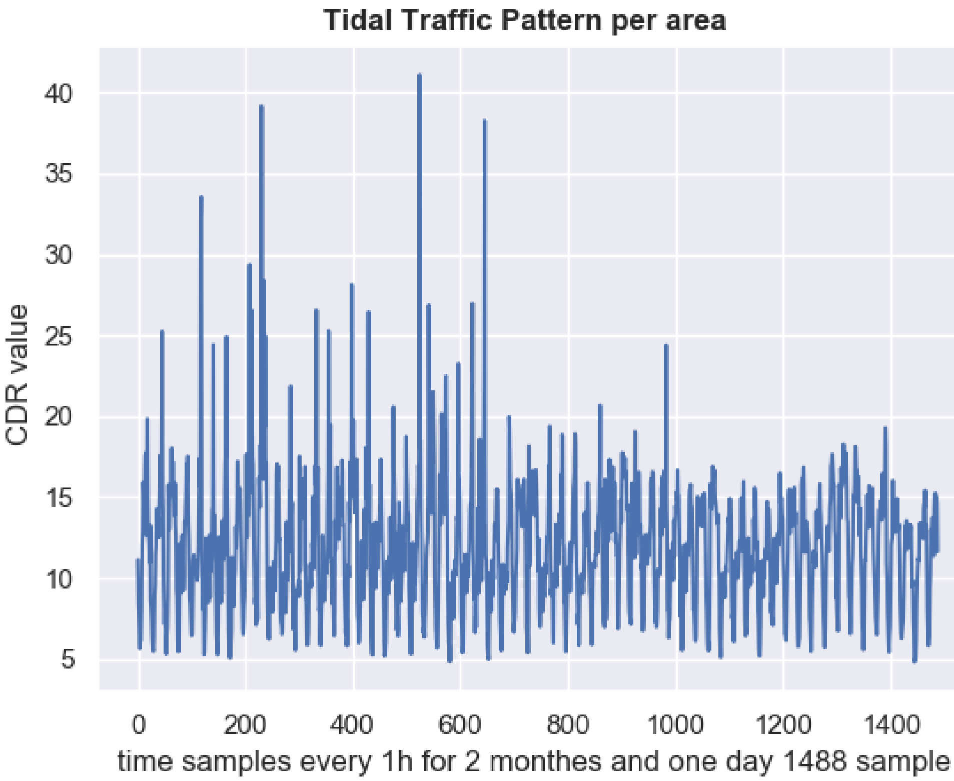


Figure 2.1: Time series CDR associated to a specific area

Our study has been done using mobile traffic data provided by Telecom Italia Mobile as a part of the Big Data Challenge [10] competition. The dataset is the result of a computation over the Call Detail Records (CDR) generated by the Telecom Italia cellular network over the city of Milano. CDR logs the user activity for billing purposes and network management. The CDR contains the following records:

- Received SMS: a CDR is generated each time a user receives an SMS
- Sent SMS: a CDR is generated each time a user sends an SMS
- Incoming Calls: a CDR is generated each time a user receives a call
- Outgoing Calls: CDR is generated each time a user issues a call

- Internet: a CDR is generated each time a user starts an internet connection or a user ends an internet connection

CDR data-set measures the level of interaction of the users with the mobile phone network.

The city of Milan was divided into a 100x100 square grid each square 550 km^2 with a side length of 235 m in figure 2.2 and figure 2.3, this is the areal unit we use throughout the thesis, and we refer to it as a “square”. figure 3.3 has size of 10000x1488, where the 10000 represents Milan grid squares and the 1488 is the CDR values. The CDR values start from a 1^{st} November 2013 at 12:00 am till 1 January 2014, where the temporal unit is 1 hour

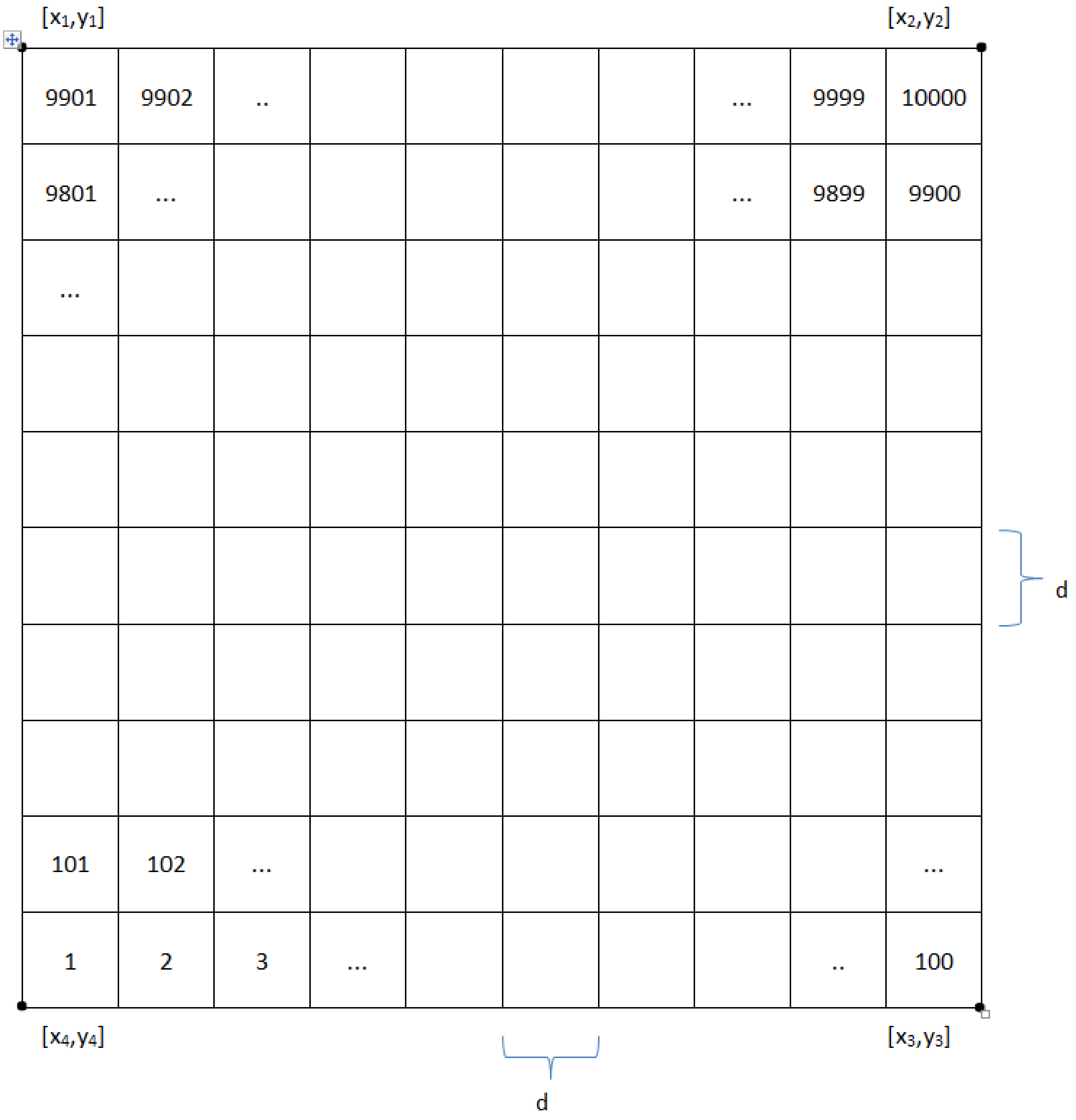


Figure 2.2: Milan grid

Each CDR activity value corresponds to the level of interaction of all the users in the square with the mobile phone network. Our work has been done with CDR for only mobile traffic activity which is a single value that describes the total activity volume in a square.

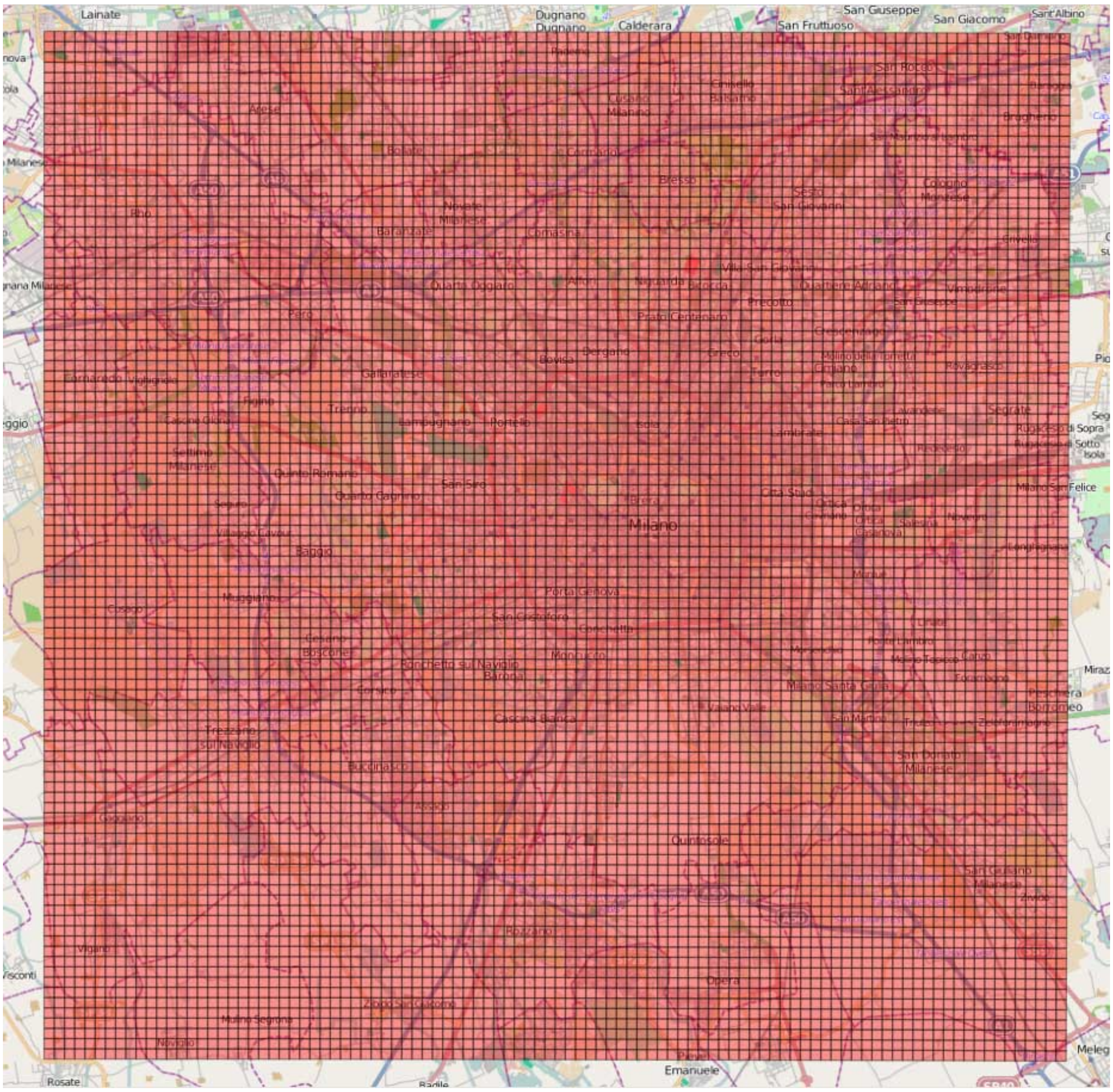


Figure 2.3: Milan map

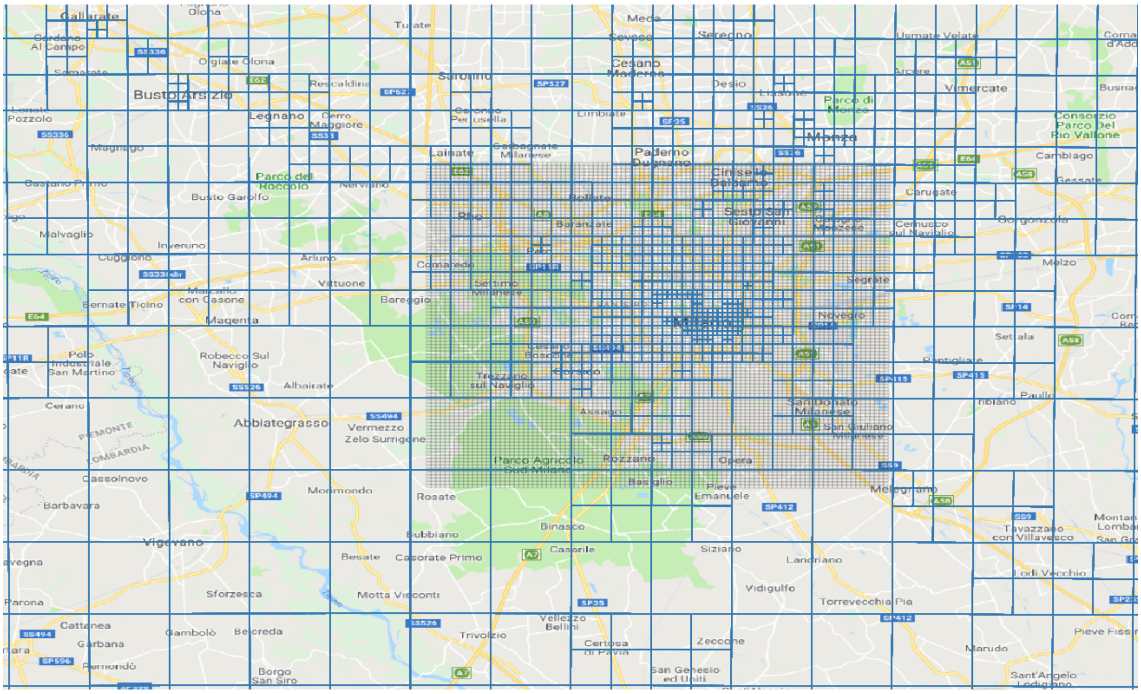


Figure 2.4: Mapping Ground-truth to Milan grid

2.2 Ground Truth

The second data-set contains geographic locations of different land-use categories present in the territory, indicating: Milan-grid cell-id, geometry, and land-use. The land-use information is critical to the accurate estimation of clustering. The major activity types we have in the land-use categorized as: residential in figure 2.8, office in figure 2.6, transportation in figure 2.5, touristic in figure 2.9, university in figure 2.7, shopping in figure 2.10, and nightlife.

We used Qgis in order to map the two data-set in figure 2.4. we have our raw ground-truth with nonuniform polygons represented as the blue squares, then we have the Milan grid represented as grey squares in figure 2.4. We mapped using Python the land-uses from the ground-truth to the squares the have been matched in the Milan grid.

Population distribution data [7] It includes: population counts, a survey of structural attributes, update and review of municipal. Anagraphical lists, the number and structural features of houses and buildings. Specifically, population counts are measured in terms of families, cohabitants, persons temporarily present, domiciles, and other types of lodging and buildings, for each administrative area. In our study, we will consider land-use data as ground truth for clustering in the reference urban area.

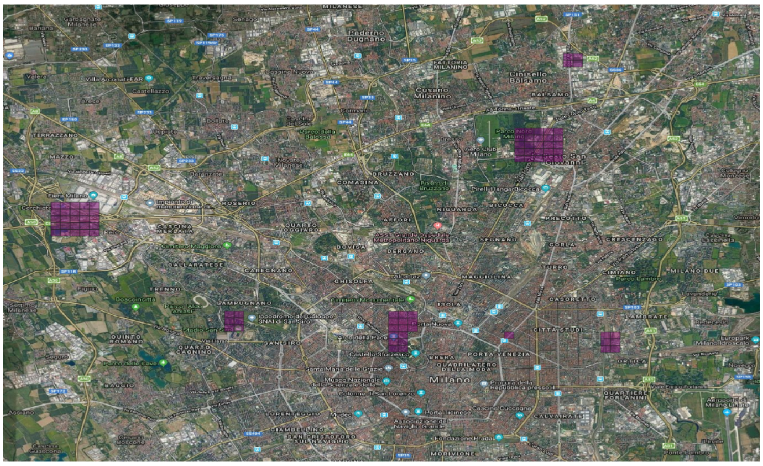


Figure 2.5: Transportation land-use

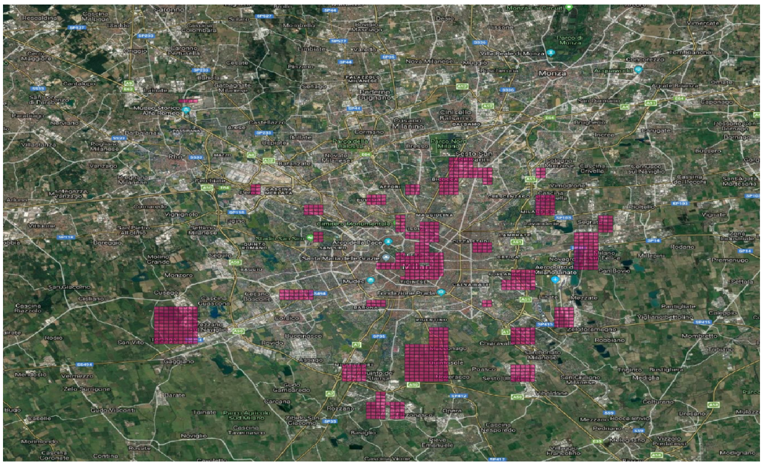


Figure 2.6: Office land-use

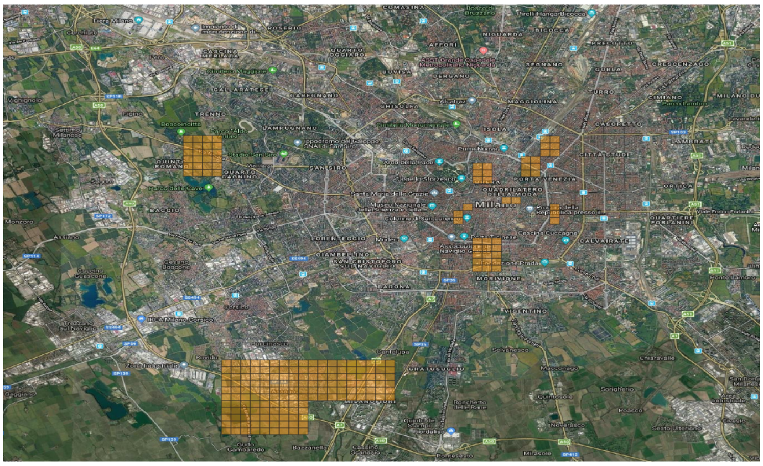


Figure 2.7: Education land-use

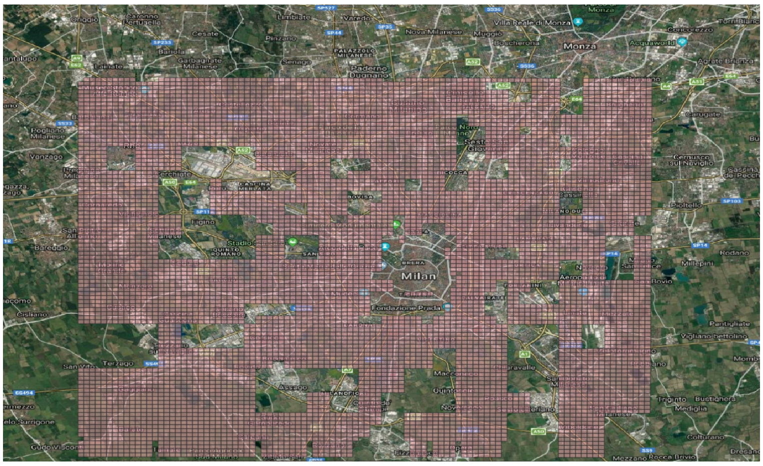


Figure 2.8: Residential land-use

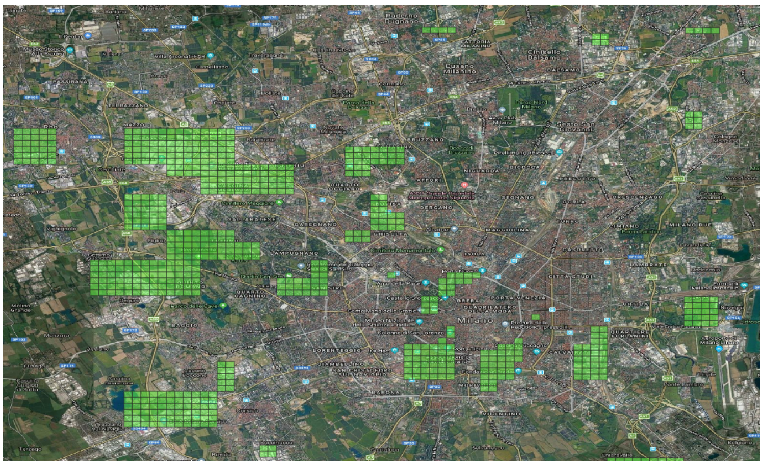


Figure 2.9: Touristic land-use

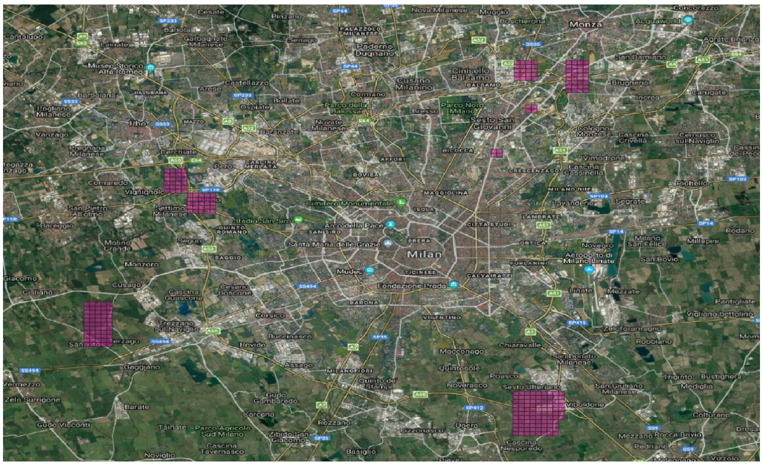


Figure 2.10: Shopping land-use

2.3 Non-negative matrix factorization based signature(NMFS)

NMFS is the novel method proposed in this work, aims to extract traffic signatures from the data and perform dimension reduction.

As described in detail in section 2.2.4. NMF aims to extract useful features from the data and perform dimension reduction, here a matrix X that represents CDR data-set is factorized into two matrices W and H, with non negative elements.

The W is composed of m rows $w_1, w_2, ...w_m$ and k rows $w_1, w_2, ...w_k$ which represents the features, H is composed of m rows $h_1, h_2, ...h_m$ which represents the weights.

In section 3.1 we described the CDR data-set, we used it in order to apply our approach for signature characterization. The CDR associated to each area has a traffic for two months and one day. In order to find similarities between 1488 hour of traffic, we reorganized the CDR data-set into (1) a working day, and (2) a weekend day. The set of days d is split into two sets d_{wd} and d_{we}, which contains all Mondays-to-Fridays, and all Saturdays and Sundays, respectively.

We performed that reorganization of the data-set, to separate the traffic behaviour to weekdays and weekend as inspired by the work in [15], [2]

After obtaining the matrices with weekday-weekend, we factorized them using NMF method explianed in section 2.24.The result of the factorization is Traffic signature for the weekday and another for the weekends. Finally we concatenated the weekend and weekdays, which represent the signature for each area shown in figure 2.11

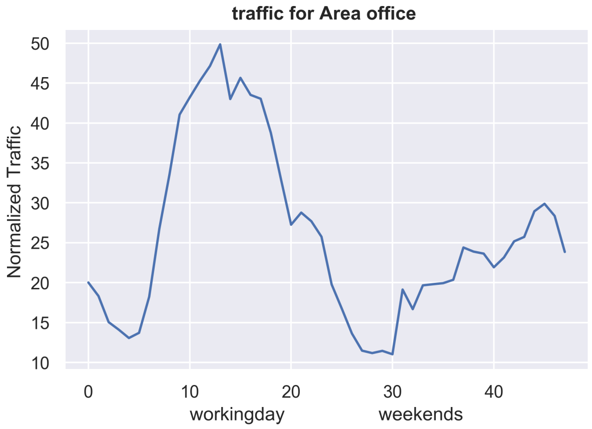


Figure 2.11: Area 1980 with office land-use for NMF signature

Chapter 3

Experiments and results

In this chapter, we will discuss the data prepossessing we applied on the CDR data-set and the ground-truth. Starting from, dealing with missed data, timestamp the data. Then the ground truth mapping to the Milan grid and how we performed it. Then signatures are clustered based on their profiles. Afterward, we performed the evaluation methods between the clustered classes and the ground-truth to have some insights about the efficiency of the signatures clustering. Finally, we made a discussion about the performance of our model in comparison with the state of the art methods.

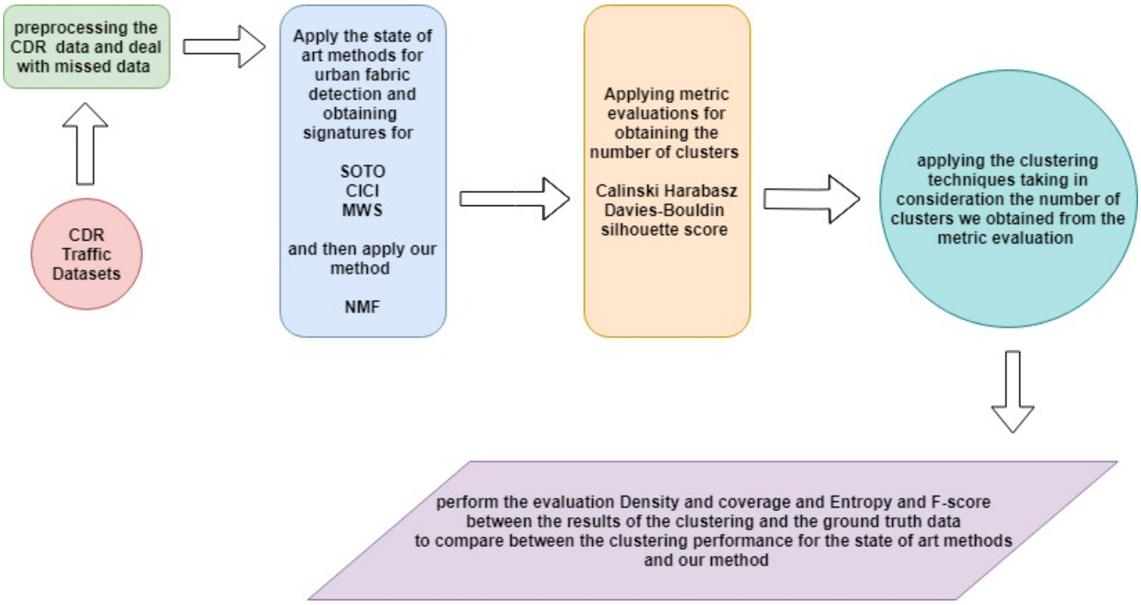
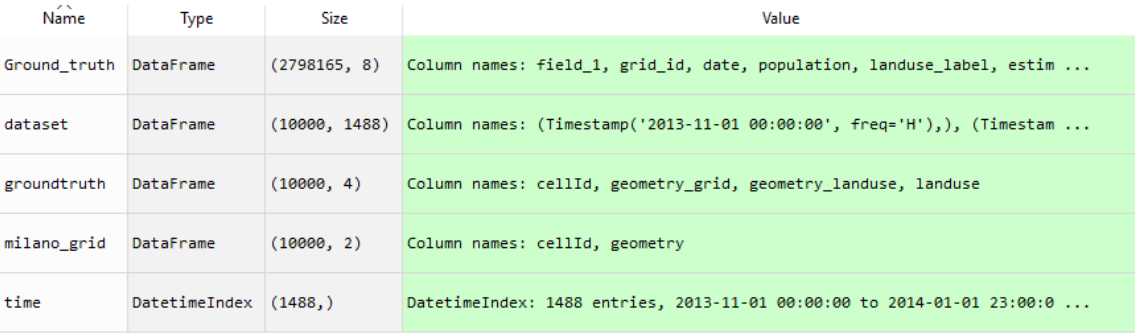

preprocessing the CDR data and deal with missed data
CDR Traffic Datasets
Apply the state of art methods for urban fabric detection and obtaining signatures for
SOTO
CICI
MWS
and then apply our method
NMF
Applying metric evaluations for obtaining the number of clusters
Calinski Harabasz
Davies-Bouldin
silhouette score
applying the clustering techniques taking in consideration the number of clusters we obtained from the metric evaluation
perform the evaluation Density and coverage and Entropy and F-score between the results of the clustering and the ground truth data to compare between the clustering performance for the state of art methods and our method

Figure 3.1: workflow of this thesis

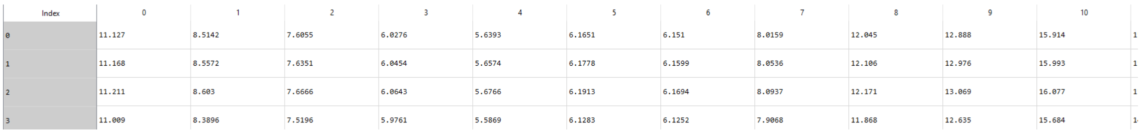
3.1 CDR prepossessing



Name	Type	Size	Value
Ground_truth	DataFrame	(2798165, 8)	Column names: field_1, grid_id, date, population, landuse_label, estim ...
dataset	DataFrame	(10000, 1488)	Column names: (Timestamp('2013-11-01 00:00:00', freq='H'),), (Timestam ...
groundtruth	DataFrame	(10000, 4)	Column names: cellId, geometry_grid, geometry_landuse, landuse
milano_grid	DataFrame	(10000, 2)	Column names: cellId, geometry
time	DatetimeIndex	(1488,)	DatetimeIndex: 1488 entries, 2013-11-01 00:00:00 to 2014-01-01 23:00:0 ...

Figure 3.2: Data-sets size

CDR matrix in figure 3.3 has size of 10000x1488, where the 10000 represents Milan grid squares and the 1488 is the CDR values. The CDR values start from a 1^{st} November 2013 at 12:00 am till 1 January 2014, where the temporal unit is 1 hour. We first dealt with the missed data (NAN) and zeroes that we have in our CDR data-set. otherwise, it will later cause a problem while obtaining the signatures and performing clustering.



Index	0	1	2	3	4	5	6	7	8	9	10	
0	11.127	8.5142	7.6055	6.0276	5.6393	6.1651	6.151	8.0159	12.045	12.888	15.914	1
1	11.168	8.5572	7.6351	6.0454	5.6574	6.1778	6.1599	8.0536	12.106	12.976	15.993	1
2	11.211	8.603	7.6666	6.0643	5.6766	6.1913	6.1694	8.0937	12.171	13.069	16.077	1
3	11.009	8.3896	7.5196	5.9761	5.5869	6.1283	6.1252	7.9068	11.868	12.635	15.684	1

Figure 3.3: CDR data-set

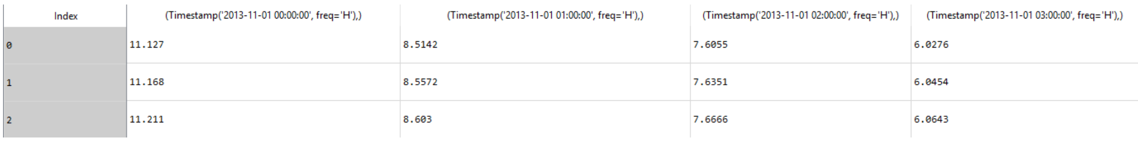
3.1.0.1 Dealing with missed data

Data in the real-world are rarely clean and homogeneous. Typically, they tend to be incomplete, noisy, and inconsistent. It is an important task of prepossessing the data by filling missing values. It is important to be handled as they could lead to wrong prediction or classification for any given model being used.

In the CDR we dealt with that missed data using fillna from Panadas in Python. Where we fill the missed data with the mean of the same column the represent the same hour for all the areas.$dataset.fillna(dataset.mean(), inplace = True)$

3.1.0.2 Time stamping the data

In Sotto model for example we have to split CDR data into working-days and weekends. On the other hand in MWS the whole set of days d is divided into seven sets, each containing elements of the data-set D that refer to one day of the week, from Monday to Sunday. In other words, $d_{mon} \cup d_{tue} \cup d_{wed} \cup d_{thu} \cup d_{fri} \cup d_{sat} \cup d_{sun} = d$ .In order to apply the state of the art approaches [15] [2] [3]. To do so, we have to stamp our data-set that have CDR values, in order to differentiate different days and hours. We used Pandads from Python to achieve that, by generated a range of timestamps starting from 11/1/2013 generated each hour $time = pd.daterange('11/1/2013', periods = 1488, freq =' H')$ Then by assign that to the columns of the data $dataset.columns = [time]$ in Figure3.4.

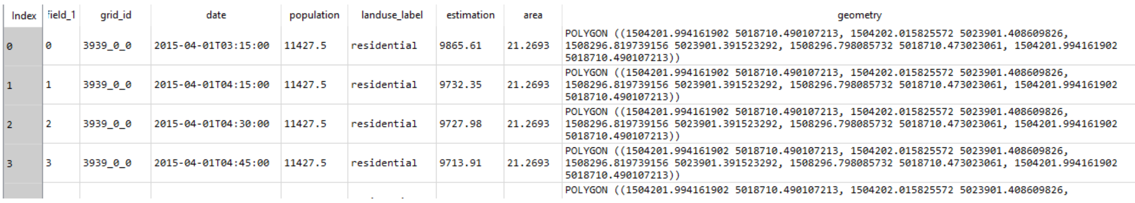


Index	(Timestamp('2013-11-01 00:00:00', freq='H'),)	(Timestamp('2013-11-01 01:00:00', freq='H'),)	(Timestamp('2013-11-01 02:00:00', freq='H'),)	(Timestamp('2013-11-01 03:00:00', freq='H'),)
0	11.127	8.5142	7.6055	6.0276
1	11.168	8.5572	7.6351	6.0454
2	11.211	8.603	7.6666	6.0643

Figure 3.4: CDR data-sets timestamped

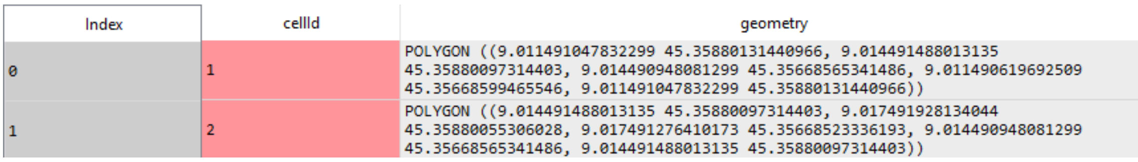
3.2 Ground-truth prepossessing

Figure 3.5 represents the data-set of "Population distribution comes from the 2011 housing census in Italy by the national organization for statistics in Italy, ISTAT". which we used the land-uses in that data as the ground-truth. It originally have a size of 2798165x8 in figure 3.2. The 2798165 represents polygons of all Lombardy. On the other hand, what we are interested in the columns is the coordinates of the polygon and the land-use.



Index	field_1	grid_id	date	population	landuse_label	estimation	area	geometry
0	0	3939_0_0	2015-04-01T03:15:00	11427.5	residential	9865.61	21.2693	POLYGON ((1504201.994161902 5018710.490107213, 1504202.015825572 5023901.408609826, 1508296.819739156 5023901.391523292, 1508296.798085732 5018710.473023061, 1504201.994161902 5018710.490107213))
1	1	3939_0_0	2015-04-01T04:15:00	11427.5	residential	9732.35	21.2693	POLYGON ((1504201.994161902 5018710.490107213, 1504202.015825572 5023901.408609826, 1508296.819739156 5023901.391523292, 1508296.798085732 5018710.473023061, 1504201.994161902 5018710.490107213))
2	2	3939_0_0	2015-04-01T04:30:00	11427.5	residential	9727.98	21.2693	POLYGON ((1504201.994161902 5018710.490107213, 1504202.015825572 5023901.408609826, 1508296.819739156 5023901.391523292, 1508296.798085732 5018710.473023061, 1504201.994161902 5018710.490107213))
3	3	3939_0_0	2015-04-01T04:45:00	11427.5	residential	9713.91	21.2693	POLYGON ((1504201.994161902 5018710.490107213, 1504202.015825572 5023901.408609826, 1508296.819739156 5023901.391523292, 1508296.798085732 5018710.473023061, 1504201.994161902 5018710.490107213))
								POLYGON ((1504201.994161902 5018710.490107213, 1504202.015825572 5023901.408609826,

Figure 3.5: Population distribution ground-truth data-sets



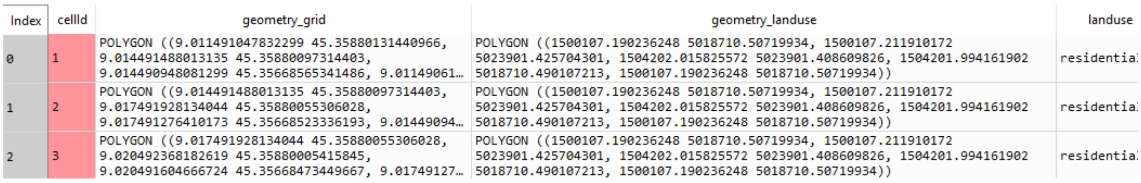
Index	cellId	geometry
0	1	POLYGON ((9.011491047832299 45.35880131440966, 9.014491488013135 45.35880097314403, 9.014490948081299 45.35668565341486, 9.011490619692509 45.35668599465546, 9.011491047832299 45.35880131440966))
1	2	POLYGON ((9.014491488013135 45.35880097314403, 9.017491928134044 45.35880055306028, 9.017491276410173 45.35668523336193, 9.014490948081299 45.35668565341486, 9.014491488013135 45.35880097314403))

Figure 3.6: Milan grid coordinates

3.2.0.1 Mapping of the land-use

In order to build the ground-truth data-set based on the land-use of Population distribution data-set, we need to make the coordinate system compatibl for the ground-truth data-set and Milan grid. So we transformed the coordinate system of Milano grid from the EPSG:4326 to EPSG:3003, using geopandas in python $MilangridEPSG3003 = milanogrid.geometry.tocrs(epsg = 3003)$

After having both coordinate systems compatible, we made a loop to find the coordinates of the polygons of Milan grid within the ground-truth data-set of Population distribution. Then we mapped the values of the land-use categories that matches Milan grid. At the end of that iteration we got 10000x4 matrix in figure 3.7 includes the cell ID, geometry of the grid, geometry of the ground-truth of Population distribution and finally the land use.



Index	cellId	geometry_grid	geometry_landuse	landuse
0	1	POLYGON ((9.011491047832299 45.35880131440966, 9.014491488013135 45.35880097314403, 9.014490948081299 45.35668565341486, 9.01149061…	POLYGON ((1500107.190236248 5018710.50719934, 1500107.211910172 5023901.425704301, 1504202.015825572 5023901.408609826, 1504201.994161902 5018710.490107213, 1500107.190236248 5018710.50719934))	residentia
1	2	POLYGON ((9.014491488013135 45.35880097314403, 9.017491928134044 45.35880055306028, 9.017491276410173 45.35668523336193, 9.01449094…	POLYGON ((1500107.190236248 5018710.50719934, 1500107.211910172 5023901.425704301, 1504202.015825572 5023901.408609826, 1504201.994161902 5018710.490107213, 1500107.190236248 5018710.50719934))	residentia
2	3	POLYGON ((9.017491928134044 45.35880055306028, 9.020492368182619 45.35880005415845, 9.020491604666724 45.35668473449667, 9.01749127…	POLYGON ((1500107.190236248 5018710.50719934, 1500107.211910172 5023901.425704301, 1504202.015825572 5023901.408609826, 1504201.994161902 5018710.490107213, 1500107.190236248 5018710.50719934))	residentia

Figure 3.7: Ground-truth matrix after the iteration

3.3 Results

All approaches for clustering from mobile traffic data rely on the same processing chain, consisting of the following steps.

- Mobile traffic signature. First,a representation of the typical mobile traffic observed at a unit area, as we introduced signature characteristics in 2.7. This can map to the complete time series of traffic, or to a summary of this, a statistical measure or compressed representation. Also, since signatures need to be comparable, they are normalized according to a normalization rule.

- Distance between signatures. Second, one computes how similar, or different, each signature is from all other signatures in the data-set. To that end, a signature distance measure is defined.
- Clustering of signatures. Third, having computed the distances among all signatures, a clustering algorithm is run so as to separate groups of signatures that have similar shapes, i.e., that are representative of equivalent traffic activities.

3.3.1 Number of clusters analysis

In figure 3.8, the Silhouette Coefficient has been obtained for the CDR data without obtaining signatures. We can notice that the Silhouette Coefficient didn't reach a maximum point and didn't converge within the first 50 clusters. We will see later after obtaining the signature we will have convergence within the first 50 iteration of clusters.

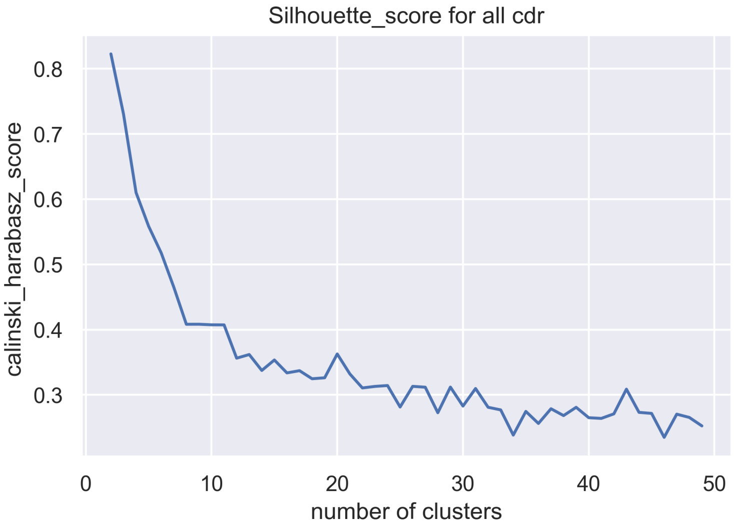


Figure 3.8: Silhouette Coefficient for the whole CDR without signatures

In figure 3.9, the Calinski Harabasz score has been calculated for the MWS signature. The higher value of the Calinski Harabasz score is 2 corsponds to number of clusters equal to 6.That local maximum reflects that the clusters are dense and well separated.In figure 3.10 we calculated the number of cluster index using the Davies-Bouldin index for the MWS,and its results in local minimum corresponding to the lower value of the Davies-Bouldin index.

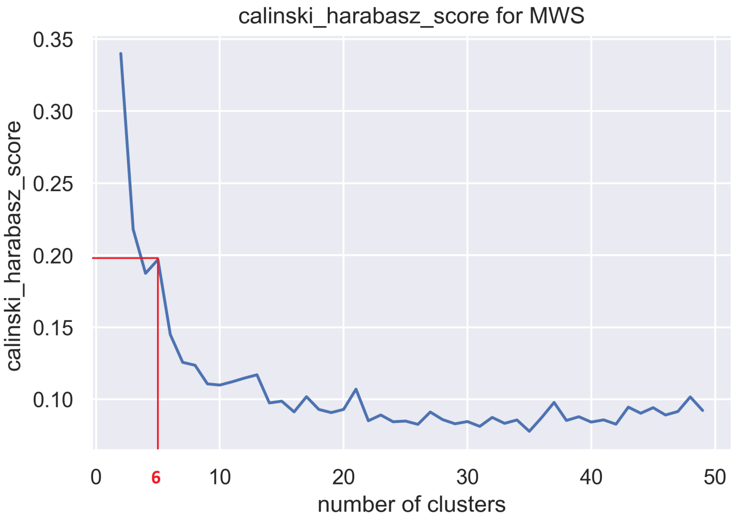


Figure 3.9: Calinski Harabasz for MWS

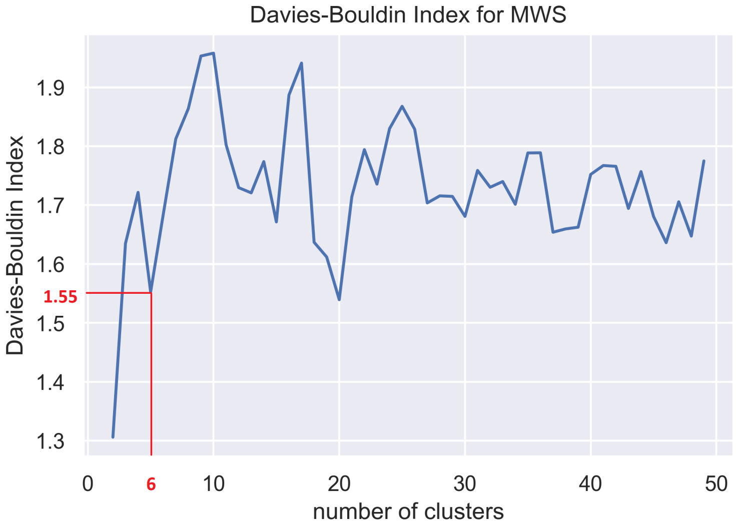


Figure 3.10: Davies-Bouldin index for MWS

In figure 3.11 is the result of the Silhouette Coefficient for the NMFS, the number of clusters obtained is corresponding to the high value of the Silhou-

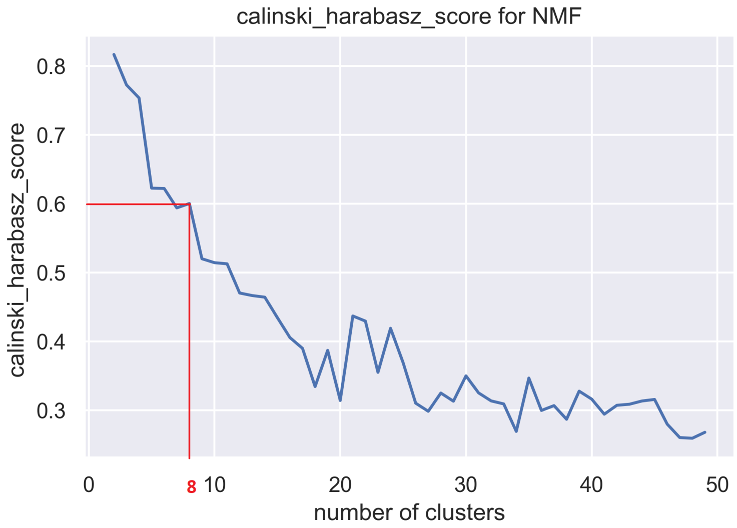


Figure 3.11: Silhouette Coefficient for NMF

ette Coefficient local maximum equal 0.6 with number of clusters equal to 8.

3.3.2 Clusters analysis

In this work, we leverage the telecommunication traffic data itself to cluster the traffic signatures based on their primary land-use. We employ MWS, SOTO, CICI, and NMFS in order to find a correlation between traffic signatures. Then signatures are clustered based on their profiles, which are representative of distinct types of traffic associated with human activities. When applied to our reference data-sets NMFS identifies seven major activity types: residential, office, transportation, touristic, university, shopping, and nightlife.

The signatures obtained in 2.7 are clustered using K-means algorithm, based on their profiles, and distances among all them.

The result of the clustering is a vector of the areas and the class that these areas are most likely be available in. Each class has a high correlation in terms of mobile traffic data.

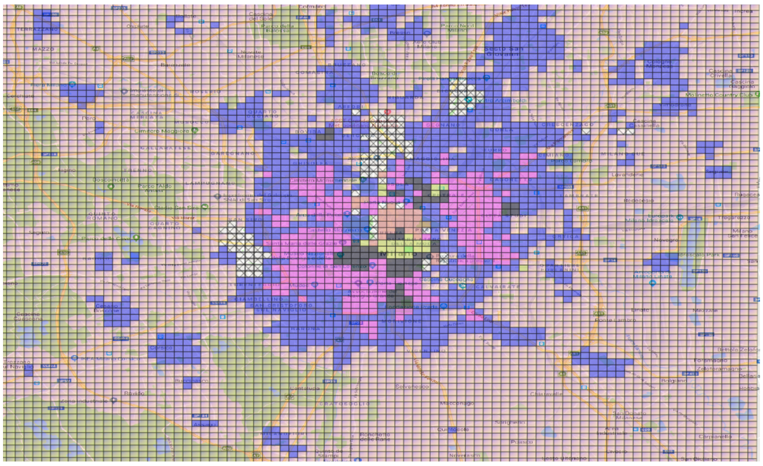


Figure 3.12: NMFS signature clustering result

Figure 3.12 represents the result of the clustering NMFS signature. It grouped the signatures in classes, which has the highest signature correlation, we are searching for the pattern that represents the geographical distribution of that class with the city.

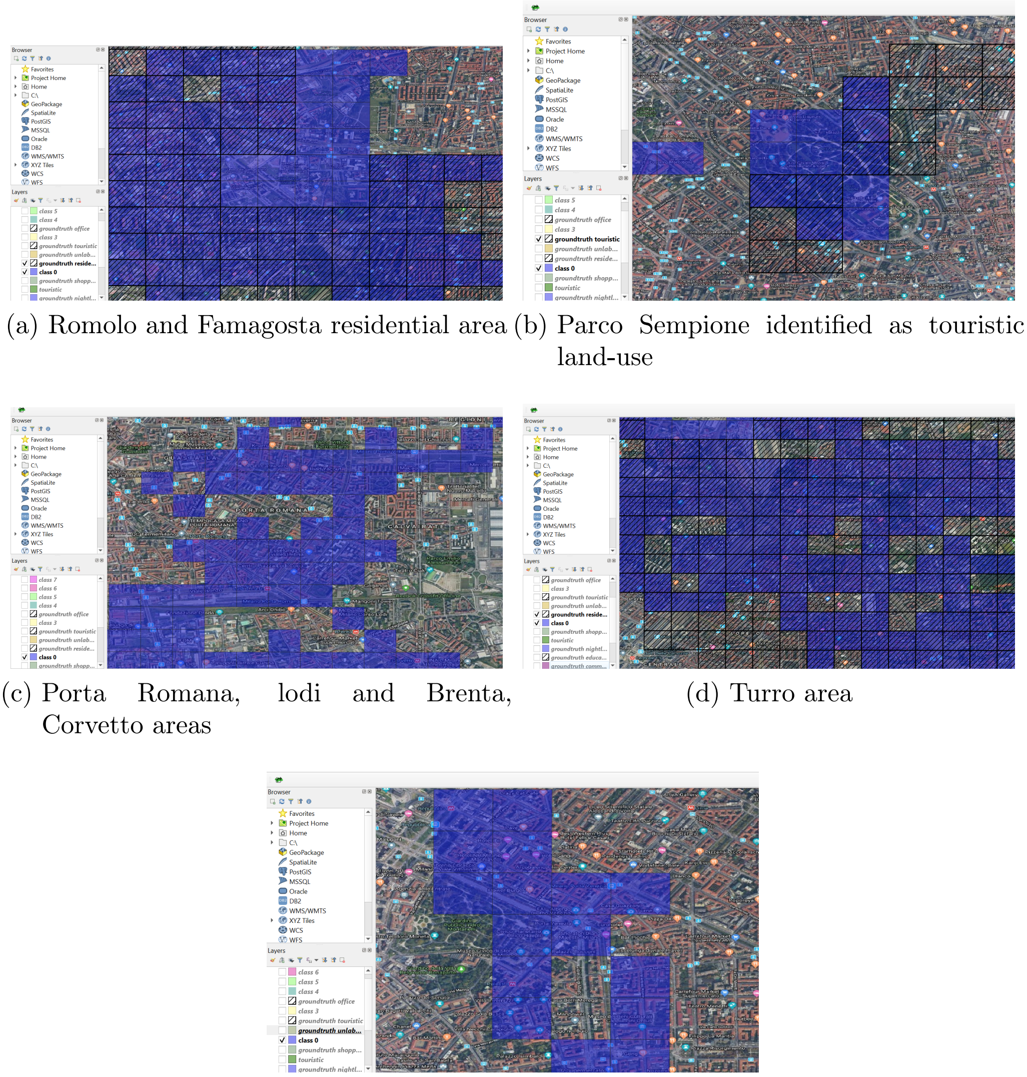
In table 3.1 is the percentage of the available land-uses in a specific class. Class0 and class1 are significantly bigger than they other classes, they have high percentage of residential land-use. Class0 also contains touristic land-use but really small in compare with the class size. Class1 contains like 80 percent of the commuting land-use, that just represents 10 percentage of the size of class1.

	c0	c1	c2	c3	c4	c5	c6	c7
Education			20%	80%	50%			10%
Office	20%		40%	20%				
Commuting		10%				20%		
Touristic			40%			80%	100%	60%
Residential	80%	90%			50%			30%

Table 3.1: Percentage of the land-uses in each class

Class0 shown in figure 3.13b, has Parco Sempione identified as touristic land-use. Also class0 includes area porta venezia, Palestro figure 3.13e. Via 22 Marzo,Porta romana, lodi and Brenta, Corvetto, foundation Prada was all included as residential area in class0 3.13c. Along the red line of the metro, Turro, Rovereto, shown in figure 3.13d and alone green metro line

Romolo and Famagosta in figure 3.13a all was identified using ground truth as residential areas.



(a) Romolo and Famagosta residential area

(b) Parco Sempione identified as touristic land-use

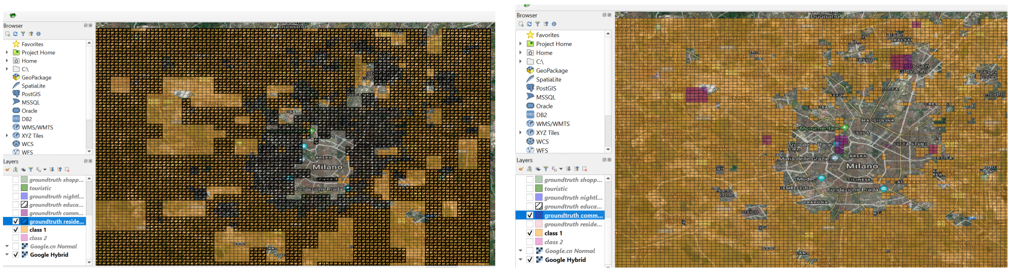
(c) Porta Romana, lodi and Brenta, Corvetto areas

(d) Turro area

(e) Porta Venezia, Paslestro areas

Figure 3.13: Class0

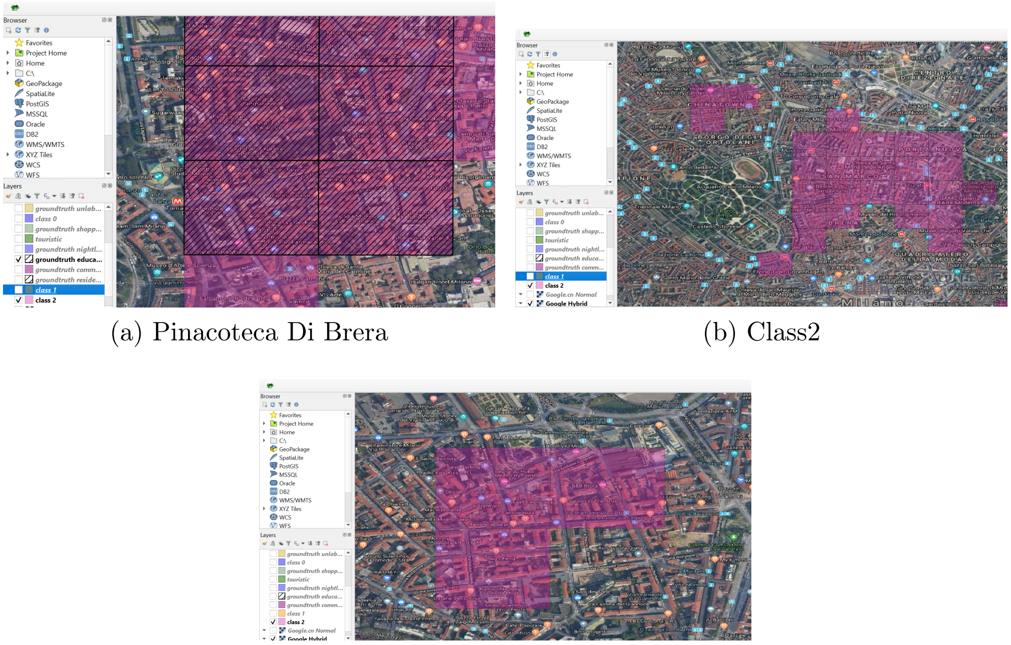
Class1, shown in figure 3.14a represents the largest cluster contains the biggest residential districts in the city. Figure 3.14b shows all the Commuting areas in Milan that are also inside class1, for example, stazione Centrale, Garibaldi, Sesto maggio Fs, lambrate, lampugnano and Pero.



(a) Residential areas (b) Commuting areas

Figure 3.14: Class1

Class2, shown in figure 3.15b, includes China town via Sarp in figure 3.15c, also area Moscova, Lanza, Turati, Brerea area. In figure 3.15a shows that Pinacoteca Di Brera is included in the land-use category as education area.

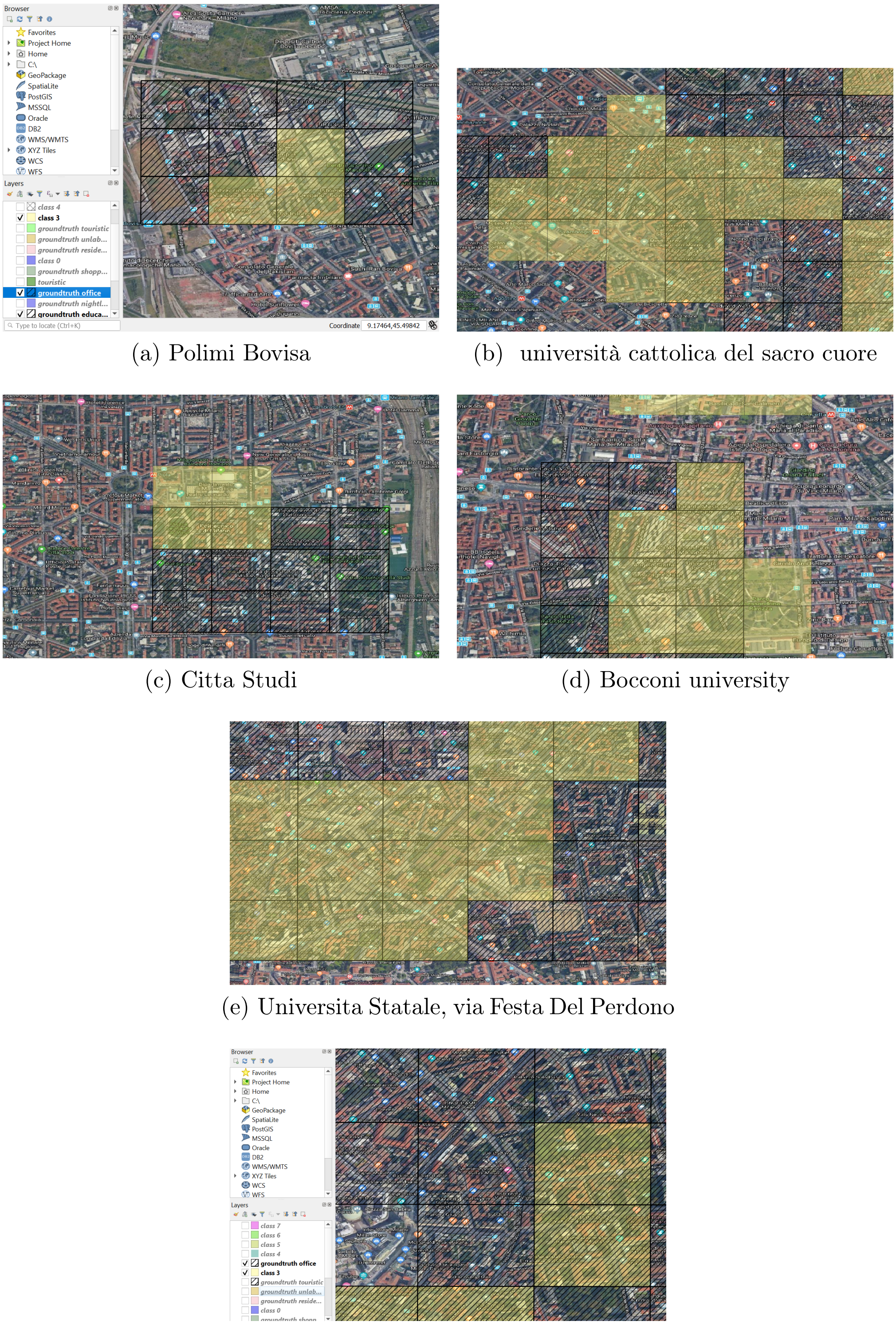


(a) Pinacoteca Di Brera (b) Class2

(c) Chinatown district

Figure 3.15: Class2

Class3, contains the majority of the universities in Milan. Citta studi shown in figure 3.16c. Polimi Bovisa show in figure 3.16a. Statale via festa del perdono in figure 3.16e. Bocconi university in figure 3.16d. Università cattolica del sacro cuore shown in figure 3.16b. Finally facoltà di scienza politica shown in figure 3.16f.



(a) Polimi Bovisa

(b) università cattolica del sacro cuore

(c) Citta Studi

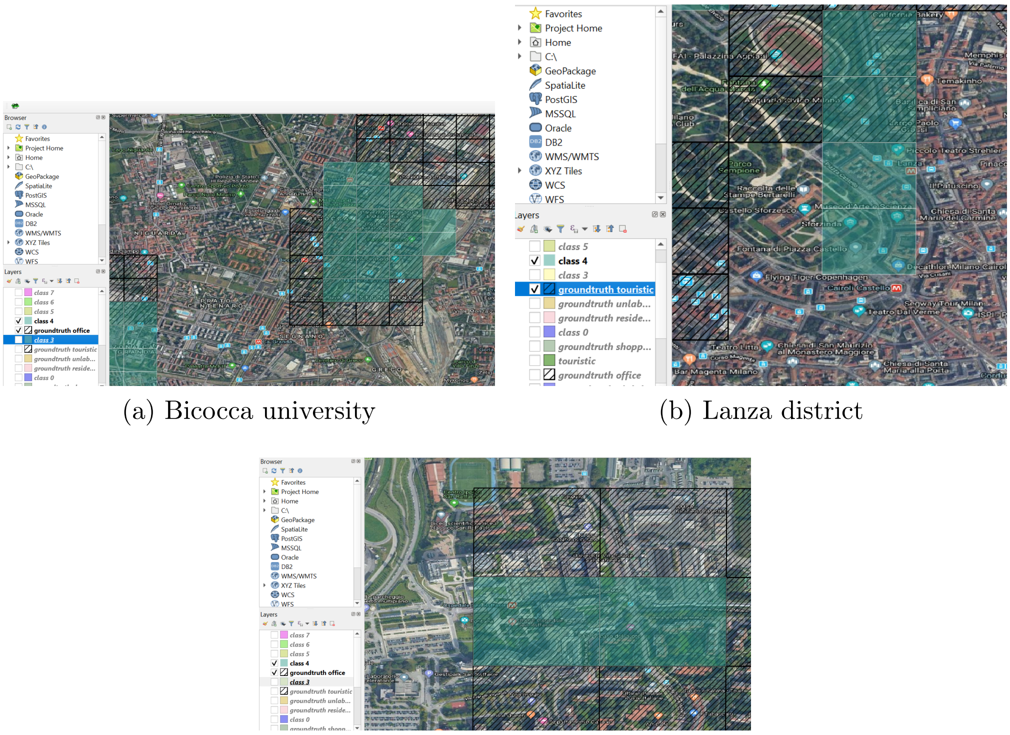
(d) Bocconi university

(e) Universita Statale, via Festa Del Perdono

(f) Facoltà Di Scienza Politica

Figure 3.16: Class3

In class4, we can find Bicocca university shown in figure 3.17a, also hospital san raffael in figure 3.17c. In figure 3.17b we can see class4 include also a part from Lanza area and piccolo teatro within the touristic land-use.



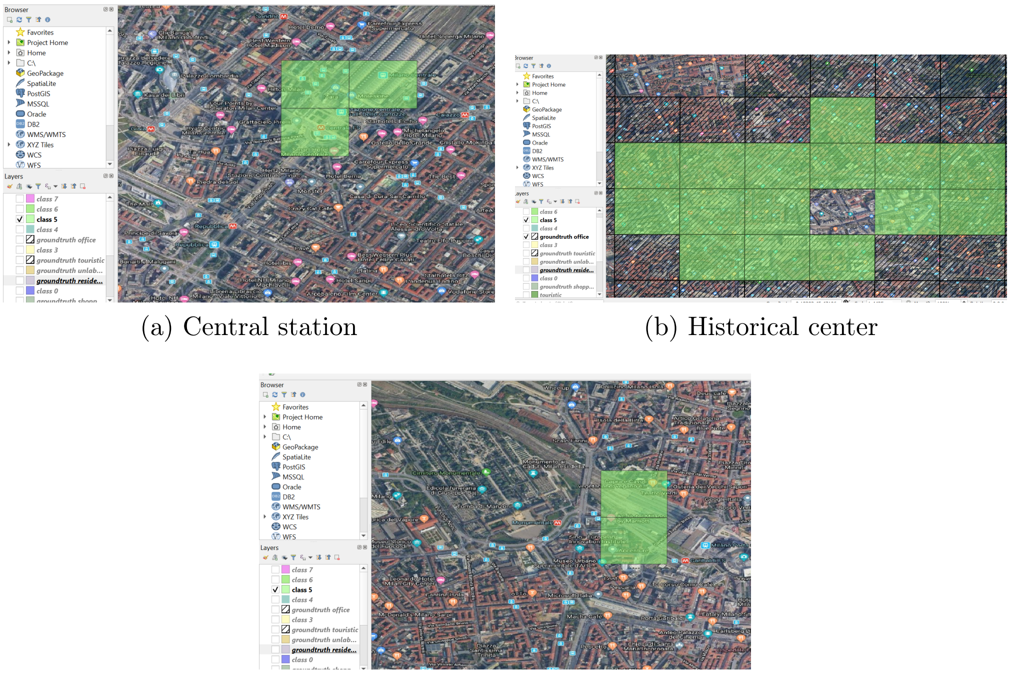
(a) Bicocca university

(b) Lanza district

(c) San Raffele hospital

Figure 3.17: Class4

In class5 we can see in figure 3.18b the historical center of Milano represented in Duomo, san Babila, montenapoleone, also we can find the central station in figure 3.18a and porta garibaldi in figure 3.18c.



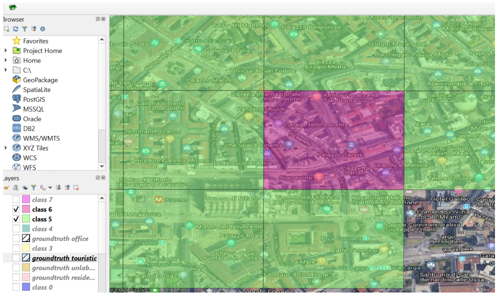
(a) Central station

(b) Historical center

(c) Porta Garibaldi

Figure 3.18: Class5

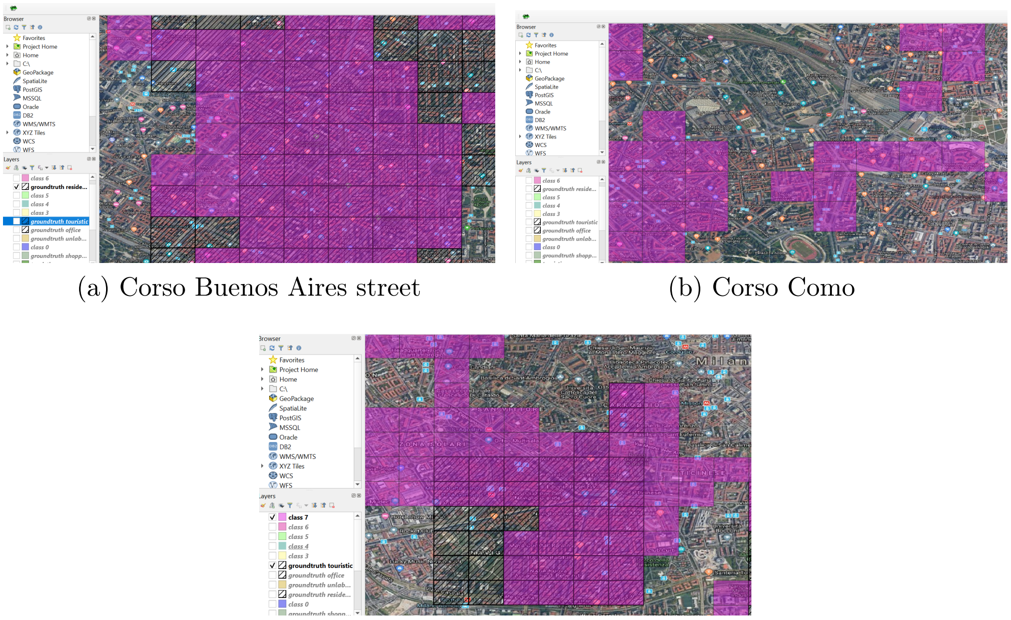
Class6, it's only one area located in the center shown in figure 3.19a



(a) Class6

Figure 3.19: Class6

Class7, as shown in figure 3.20a, it represent Corso Buenos Aires and Loreto and Lima all has residinatial land-use. In figure 3.20b show Corso Como and China town. In figure 3.20c is the area of Porta Genova, Darsena, and via Torino, via Ticinesi.



(a) Corso Buenos Aires street

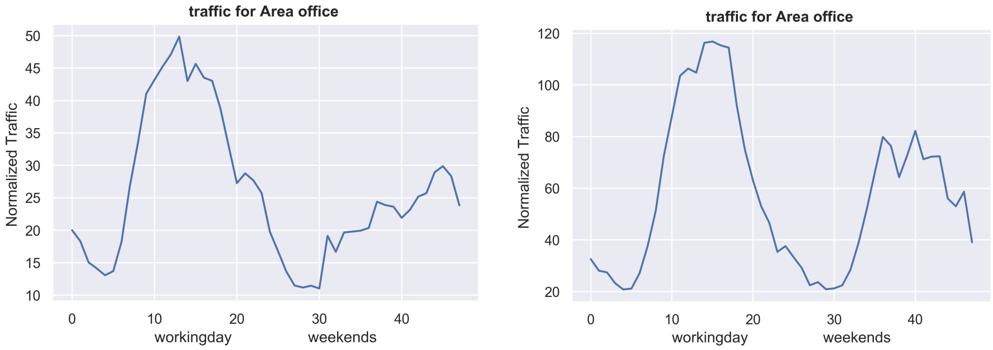
(b) Corso Como

(c) Darsena, Porta Genova, Via Torino

Figure 3.20: Class7

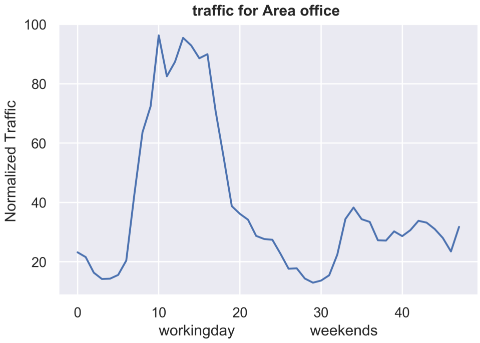
3.3.3 NMFS signature results

Figure 3.21 represent the signatures of offices in different areas. The three signatures have the same behaviour that can describe the behaviour of the land-use. We notice a lower traffic profile in weekends than the working-days, The traffic increases rapidly in the morning from around 6 am. Figure 3.22 represent a signatures of a residential areas. The signature has traffic profile in the working days. We can notice a signicant peak in the traffic around 20:00 in the working-days signature. Figure 3.23 represents a commuting area. These commuting areas 7621, 7220, are located in lampugnano bus station. Figure 3.25 represents the signature of night-life area. Finally figure 3.24 represents the education signature.



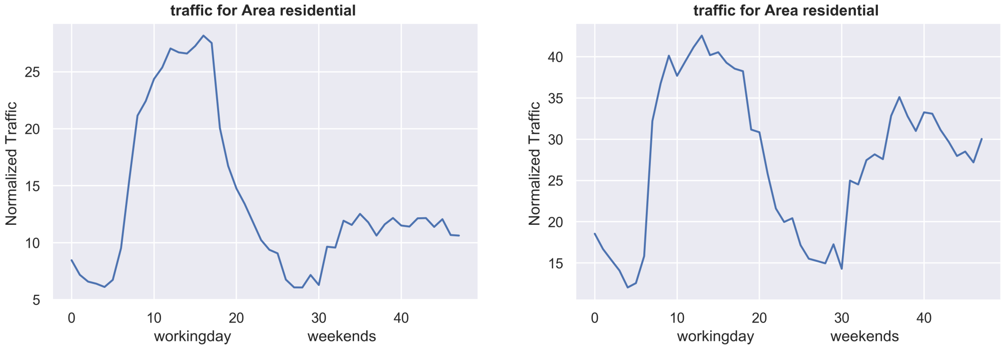
(a) Area 1980 office signature

(b) Area 2047 office signature



(c) Area 2079 office signature

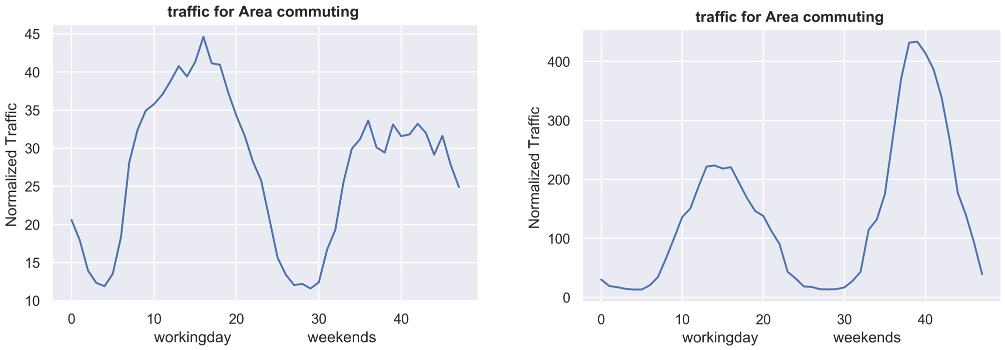
Figure 3.21: Office signature



(a) Area 2627 residential signature

(b) Area 3934 residential signature

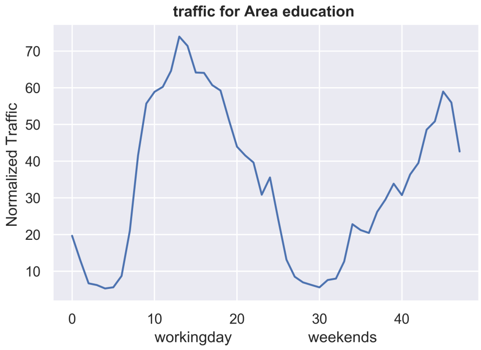
Figure 3.22: Residential signature



(a) Area 7220 commuting signature

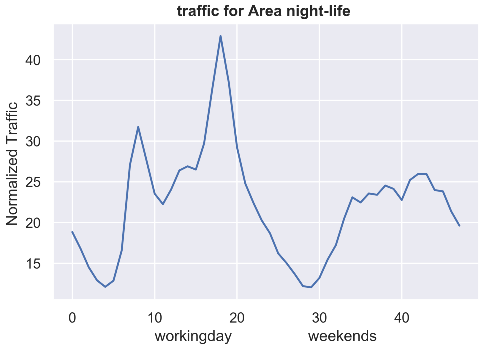
(b) Area 7621 commuting signature

Figure 3.23: Commuting signature



(a) Area 2544 education signature

Figure 3.24: Education signature



(a) Area 3376 night-life signature

Figure 3.25: Night-life signature

3.3.4 Evaluation results

In order to measure the quality of clustering, , We evaluate the quality of the unit area classification with respect to the available ground-truth data according to the following metrics mentioned in [2] [1]. We preformed this evaluations between the proposed signature and the state of the art methods.

3.3.4.1 Entropy

The entropy figure 3.26 associated to a ground-truth class within given clusters allows estimating the dispersion of the class.

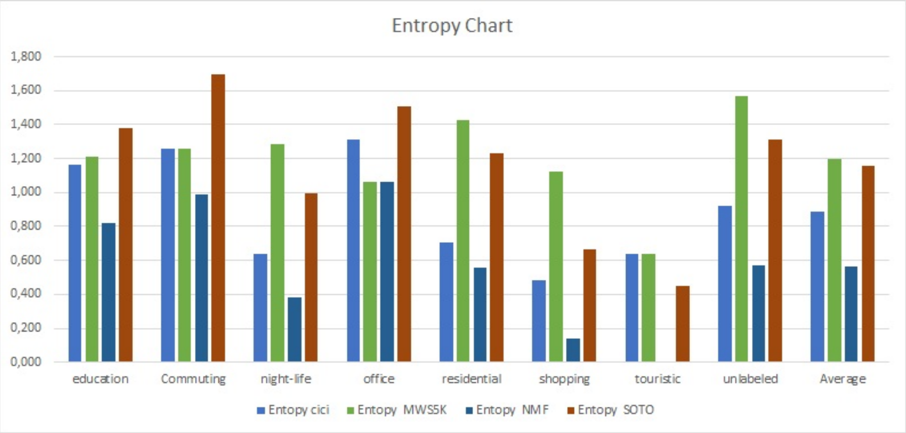


Figure 3.26: Entropy of NMF with the state of the art

In figure 3.26, We can see the 7 land-usage and the entropy for each signature technique. The NMFS has lower entropy than the other signatures.

Lower entropy is thus an indicator of a less random, i.e., more precise, assignment of ground-truth data of a given class to clusters defined by the detection strategy.

3.3.4.2 Coverage

The coverage figure 3.27 is defined as the percentage of ground-truth elements for example office included within those clusters.In figure 3.27 We can see the 7 land-usage and the coverage for each signature technique. The NMFS has higher coverage than the other signatures. Higher coverage indicates a reduced level of randomness, and thus a more precise clustering

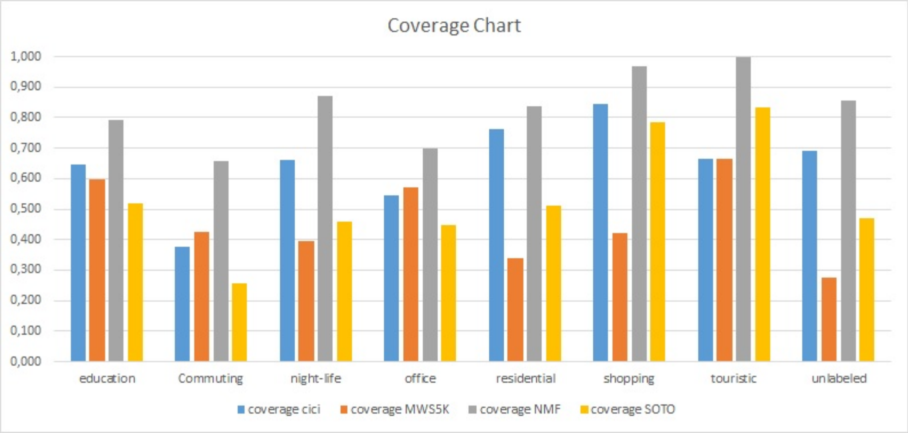


Figure 3.27: Coverage for NMF and state of the art

3.3.4.3 F-score

The F-score index figure 3.28 allows determining a single, final score to each detection technique, by combining entropy and coverage for each class of clusters.

The F-score index ranges in [0, 1], with 1 indicating the best performance achievable by the given cluster set, with respect to ground-truth.

We can see in figure 3.28 the 7 land-usage and the F-score for each signature technique. The NMFS has a higher F-score than the other signatures. That means that has higher coverage and lower entropy.

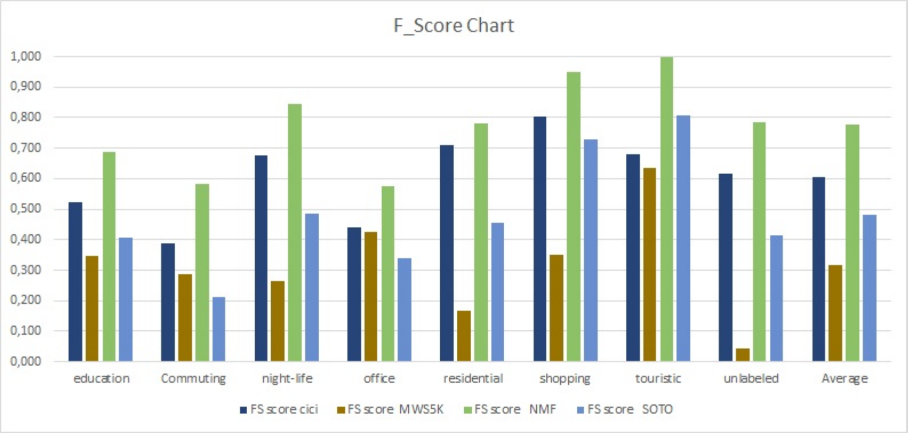


Figure 3.28: F-score for NMF with the state of the art

Chapter 4

Conclusion

We addressed in this thesis finding typical correlated traffic pattern in the mobile network by exploiting NMF. The main goal is to characterize the internet traffic in order to better describe the tidal effect that occurs in the metro network.The aim to optimize the resources and better allocation of resources and better traffic planning to avoid network congestion.

In the field of pattern recognition, the Non Negative Matrix Facorization is one of the most used method. In [14] [4] [9],this method is widely studied. They concluded that this kind of pattern extraction is useful for a large group of data, from image to speech recognition. Therefore, thanks to the matrix decomposition, it is possible to detect the traffic signatures in the metro network.

We compared the four signature characterization methods solutions reported in figure 3.28 and figure 3.26 and figure 3.27. Thus provide results in terms of entropy, coverage and F-score for the state of-the-art approaches of Soto, Cici , MWS as well as for our proposed approach NMFS.

The signature based on the NMFS attain a significantly lower entropy than solutions based on the other signatures proposed by Soto and Cici and MWS. This indicates a reduced level of randomness, and thus a more precise classification of unit areas with respect to the ground-truth data in Milan. Also, the increased accuracy does not come at a cost in terms of coverage, in figure 3.27 . In fact, the entropy gain granted by NMFS is associated to an increase in coverage, thus proving the higher effectiveness of NMF signatures of unit areas.

The result above is summarized in figure 3.28, which depicts the F-score that is a single, final score considers both entropy and coverage. The F-score further evidences that solution based on NMFS improve current state-of-the-art techniques in the analysis of mobile network traffic in urban area. By comparing NMFS with other methods available in the literature on the basis of the performance parameters introduced above, we were able to show that the NMFS has several advantages.

The analysis of mobile traffic in an urban area could be improved by aggregating other information coming from mobile providers and social media, such as Facebook or Twitter. The collection of these information could be used for many purposes, one of them is the prediction of the traffic load on-demand avoiding bottlenecks and providing dynamic allocation of network. Also to understand why the network experiences unexpected peaks of traffic data during the day. resources [14].

Bibliography

[1] Razvan Stanica Angelo Furn, Marco Fiore. A comparative evaluation of urban fabric detection techniques based on mobile traffic data. *ASONAM '15 Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pages 689–696, 2015.

[2] Razvan Stanica Angelo Furno, Marco Fiore. A tale of ten cities: Characterizing signatures of mobile traffic in urban areas. *IEEE Transactions on Mobile Computing*, 16(10):2682 – 2696, 2017.

[3] Athina Markopoulou Carter T. Butts Blerim Cici, Minas Gjoka. On the decomposition of cell phone activity patterns and their connection with urban ecology. *the 16th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, pages 317–326, 2015.

[4] Nicholas Blumm Chaoming Song, Zehui Qu and Albert Lszl Barabsi. Limits of predictability in human mobility. *Science*, 2010.

[5] M. Fiore D. Naboulsi, R. Stanica. Classifying call profiles in largescale mobile traffic datasets. *Proc. IEEE Infocom*, 2014.

[6] L. Liu C. Ratti F. Calabrese, G. Di Lorenzo. Estimating origin- destination flows using mobile phone location data. *IEEE Pervasive Computing*, 10(4):3644, 2011.

[7] Mounim El-Yacoubi Marco Fiore Ghazaleh Khodabandelou, Vincent Gauthie. Population estimation from mobile network traffic metadata. *IEEE 17th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, 2016.

[8] J. Bolot H. Zang. Mining call and mobility data to improve paging efficiency in cellular networks. *Proc. ACM MobiCom*, 2007.

[9] Patrik O Hoyer. Non-negative matrix factorization with sparseness constraints. *Journal of machine learning research*, page 1457–1469, 2014.

[10] Telecom Italia. Big data challenge, 2014.

[11] A.X. Liu-J. Pang S. Venkataraman J. Wang M.Z. Shafiq, L. Ji. A first look at cellular network performance during crowded events. *Proc. ACM SIGMETRICS*, 2013.

[12] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[13] R. H. Turi S. Ray. Determination of number of clusters in kmeans clustering and application in colour image segmentation. *Proc. ICAPRDT*, 1999.

[14] Guido Alberto Maier Achille Pattavina Sebastian Troia, Gao Sheng. Identification of tidal-traffic patterns in metro-area mobile networks via matrix factorization based model. *IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, 2017.

[15] Vıctor Soto and Enrique Frias-Martinez. Robust land use characterization of urban lanscapes using cell phone data. *Proceedings of the 1st Workshop on Pervasive Urban Applications, in conjunction with 9th Int. Conf. Pervasive Computing*, 2011.