Clustering on multivariate road traffic time series

May 18, 2021

1 Introduction

The aim of the project is to preliminary test a clustering procedure on multivariate road traffic time series in order to separate different paths between the days of the week and between different months. To create the multivariate time series two fundamental variables are considered the flow $q(t, x)$ and the density $\rho(t, x)$. The clustering technique used is K-means with soft-dynamic time warping to compare the series. To see if the clustering technique is able to generalize well traffic dynamics, it is also applied to unseen data that varies both in time (same segment road different intervals of time considered for train and test) and in space (different segments roads for train and test but same interval of time considered). The train and test sets studied share similar boundary levels of flow, density and speed. In order to decide the most suitable number of clusters the silhouette coefficient is computed as well as a similarity measure between nearest clusters based on soft-dynamic time warping. Moreover a lower bound for a minimum number of days inside a cluster is fixed to overcome the trade off between interpretability of the clusters and their number.

In section 2 I present the data used, the detectors considered S60 and S1816,the time intervals measured 2013 and 2014, the pre-process strategy implemented to treat rough data and normalize them. In section 3 I explain the soft-DTW similarity measure between time series, the K-means algorithm and a set of metrics to evaluate the technique. Then in section 4 the results are showed: both detectors are used for training the centroids and as test set, for S60 detector the train on 2013 set is then validated also on 2014 set. The procedure is repeated for different aggregation time and only significant results are presented. Finally in section 5 conclusions and outlook are reported.

2 Data and Pre-process

To test the procedure I used data from Minnesota Department of Transportation.The roads considered are I-35W and I-94 the segments of the roads considered are showed in Figure 1.1

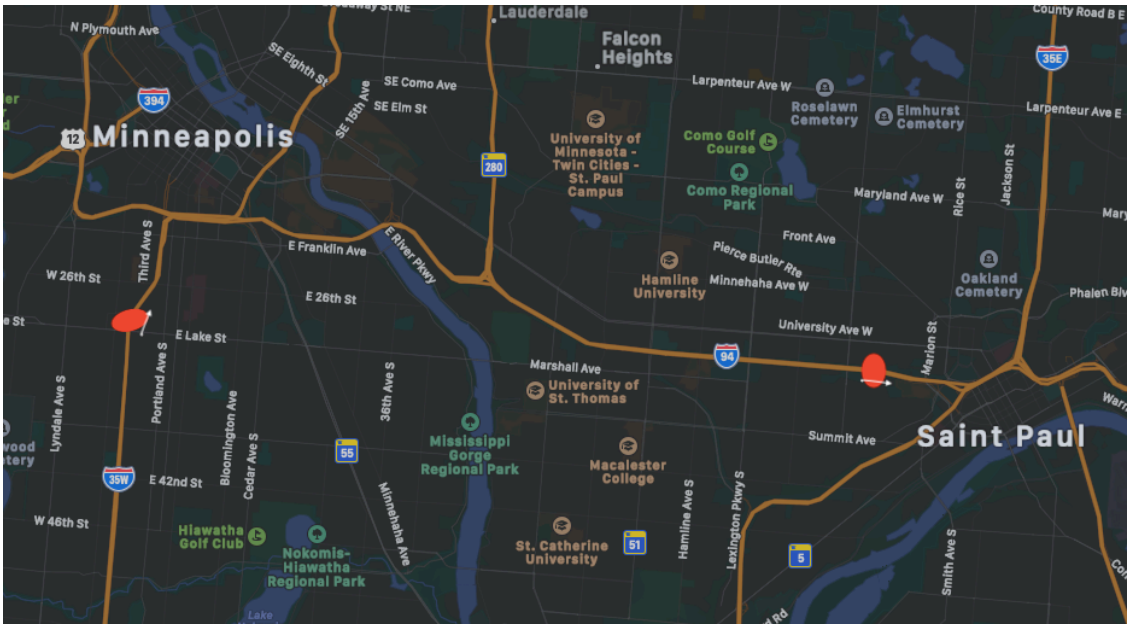

Figure 1.1 Satellite image of I-35 and I-94

The two detectors considered are S60 on I-35W north direction and S1816 on I-94 east direction. Despite the detectors are in different roads they share the same number of lanes 5 and they have the similar boundary levels in term of flow, density and speed in the 2013 year. In addition both segments do not present ramps 500 meters before and after the detectors. The flow, density and speed measurements are downloaded using [1]. For both detectors all lanes are aggregated together and the 30-minute averages from 01/01/2013 00:00 to 31/12/2013 23:30 of the dimensions is computed (17650 observations). The procedure is then repeated also with 6-minute averages (87600 observations).

S60 detector boundary levels 30-minute averages 2013:

- $\rho(t,x) : MIN = 0.37 \;\; veh/km \quad MAX = 91.84 \;\; veh/km$
- $q(t,x) : MIN = 74 \;\; veh/h \quad MAX = 6074 \;\; veh/h$
- $v(t,x) : MIN = 7.02 \;\; km/h \quad MAX = 98.65 \;\; km/h$

S1816 detector boundary levels 30-minute averages 2013:

- $\rho(t,x) : MIN = 0.7216 \;\; veh/km \quad MAX = 90.5739 \;\; veh/km$
- $q(t,x) : MIN = 112 \;\; veh/h \quad MAX = 7266 \;\; veh/h$
- $v(t,x) : MIN = 8.509 \;\; km/h \quad MAX = 86.79 \;\; km/h$

For S60 detector, the centroids calculated on data of the 2013 year are also tested on data from 01/04/2014 to 31/07/2014 of the same traffic road. As mentioned before the procedure is repeated for both 30 and 6-minute averages.

[1] http://data.dot.state.mn.us/datatools/

S60 detector boundary levels 30-minute averages 2014:

- $\rho(t,x) : MIN = 0.8309 \;\; veh/km \quad MAX = 97.47 \;\; veh/km$
- $q(t,x) : MIN = 144 \;\; veh/h \quad MAX = 5870 \;\; veh/h$
- $v(t,x) : MIN = 13.53 \;\; km/h \quad MAX = 87.32 \;\; km/h$

Due to the long period of time considered, from 01/01/2013 to 31/12/2013 for both detectors are present missing values. To reduce their impact on the normalization procedure and on the clustering algorithm missing values are replaced with substituted values. Generally missing values are created when detectors are out of order, no measure of flow and density are registered for a certain period. To overcome that, an imputation process is performed. If, for example, data on 14/03/2013 at 9:30 are missing the imputation is done by taking the median of the data on 01/03, 07/03, 21/03 and 28/03 at 9:30. By considering the median of the four observations closest to the missing one, with the respect to the temporal shift, the possible impact of the public holidays in the imputation procedure is reduced.

Time series, flow and density, are preprocessed using normalization over all period (17520 observations for 30-minute averages and 87600 observations for 6-minute averages). This scaler is such that each output time series is in the range [0,1] allowing to have identical scales for time series with originally different scales (veh/h and veh/km):

$$\rho_{norm}(t,x) = \frac{\rho(t,x) - MIN}{MAX - MIN}$$

.

$$q_{norm}(t,x) = \frac{q(t,x) - MIN}{MAX - MIN}$$

.

The MinMaxScaler function from the Python machine learning library **Scikit-learn** [2] trasforms each values of the time series proportionally within the range [0,1] preserving the shape. Density and flow scaled series are not amplitude invariant, they do not have the same standard deviation (reached instead by using standardization). By looking forward to the clustering procedure density and flow scaled series could not have the same importance in explaining the variance within the cluster. A single modality could be responsible for a large part of the variance inside a specific cluster. The inverse_transform method of MinMaxScaler, that undo the scaling of a data point according to feature_range,

$$\rho(t,x) = \rho(t,x)_{norm} * (MAX - MIN) + MIN$$

$$q(t,x) = q(t,x)_{norm} * (MAX - MIN) + MIN$$

is used in the centroids representations to have a better comprehension of paths, no more in the [0,1] range.

Despite the $MIN \;\; MAX$ for $\rho(t,x)$ and $q(t,x)$ are defined over the 2013 year, the clustering procedure is applied to the 365 daily multivariate time series with 48 daily observations for 30-minute averages and 240 observations 6-minute averages. The choice of scaling the two varaibles with respect to the entire year time and not with respect to the single 365 daily time series is done

[2] Pedregosa et.Al "Scikit-learn: Machine Learning in Python"

to preserve variability with respect to different days of the week. For example I simple assume that the MAX density reached on Sunday or Saturday is really low compared to MAX density reached on another working day.

To create the 365 daily normalized multivariate time series a simple **reshape** procedure is applied:

$[ts * n, 2] \implies [ts, n, d]$ where $ts = 365$ is the number of time series, $n =$ is the number of daily observations (48 or 240 depending on minute aggregation averages) and $d = 2$ is the dimensionality of the multivariate time series (flow and density)[3].

3 Methodology

3.1 Soft-Dynamic Time Warping

After having treated time series and pre-processed them in order to have a multivariate time series for every day of the year, I have to define a strategy in order to compare different series both for assign each series in a cluster and update the centroids in the K-means algorithm. Dynamic Time Warping is a technique to measure similarity between two temporal sequences considering not only the temporal alignment but every binary alignment of the two series. For example a similar traffic condition based on flow and density could be recognized in different hours of the day in two different series. The calculation of the DTW similarity involves a dynamic programming algorithm that tries to find the optimum warping path between two series under certain constraints.

Given two multivariate series that corresponds to two different days:

$x \in R^{2\times n}$ and $y \in R^{2\times n}$ valued in R^2 (flow and density).

Consider a function in order to compare different points of the two series ($x_i \in R^2$ and $y_j \in R^2$) $d : R^2 \times R^2 \Rightarrow R$, such as $d(x_i, y_j) = (\sum_{i,j=1}^{n} (|x_i - y_j|^p)$, where usually $p = 2$ and $d(x_i, y_j)$ is the quadratic Euclidean distance between two vectors.

A matrix of similarity is computed:

$\Delta(x, y) := [d(x_i, y_j)]_{i,j} \in R^{2\times n}$

$\Delta(x, y)$ can also be defined as local cost matrix, such a matrix must be created for every pair of series compared.

The DTW algorithm finds the path that minimizes the alignment between x and y by iteratively stepping through $\Delta(x, y)$, starting at $[d(x_i, y_j)]_{1,1}$ and finishing at $[d(x_i, y_j)]_{n,n}$, and aggregating the cost [4]. At each step, the algorithm finds the direction in which the cost increases the least under the chosen constraints. These constraints typically consist in forcing paths to lie close to the diagonal of the local cost matrix [5].

By considering $A_{n,n} \subset \{0, 1\}^{n,n}$ all binary alignment matrices the DTW similarity measure reads as follow:

[3]The Pre-process strategy is repeated separately for the three sets considered S1816 2013, S60 2013 and S60 2014 to avoid data leakage

[4]Sardà-Espinosa:"Comparing Time-Series Clustering Algorithms in R Using the dtwclust Package" chapter 2

[5]Sakoe,Chiba: "Dynamic programming algorithm optimization for spoken word recognition," IEEE Transactions on Acoustics, Speech and Signal Processing, vol. 26(1), pp. 43–49

$$DTW(x,y) = \min_{A \in A_{n,n}} \langle A, \Delta(x,y) \rangle \tag{1}$$

This creates a warped "path" between x and y that aligns each point in x to the nearest point in y.

However I can not define dynamic time warping as a distance because does not satisfy the triangular inequality, moreover it is not differentiable everywhere due to the min operator.

Soft-Dynamic Time Warping is a variant of DTW that is differentiable. It uses the log-sum-exp formulation [6]:

$$DTW^{\gamma}(x,y) = -\gamma \log \sum_{A \in A_{n,n}} exp(-\frac{\langle A, \Delta(x,y) \rangle}{\gamma}) \quad where \quad \gamma \geq 0 \tag{2}$$

Despite considering all alignments and not just the optimal one, soft-DTW can be computed in quadratic time $O(2 \times n^2)$ as DTW, however as DTW soft-DTW does not satisfy the triangular inequality. Soft-DTW is a symmetric similarity measure, it supports multivariate series as DTW, and it can provide differently smoothed results by means of a user-defined parameter γ.

The "path" created between x and y is smoother than the one created with DTW. Soft-DTW depends on a hyper-parameter γ that controls the smoothing. As showed in Figure 2.1 and in equation (3) DTW corresponds to the limit case when γ =0.

$$DTW^{\gamma}(x,y) = \begin{cases} \min_{A \in A_{n,n}} \langle A, \Delta(x,y) \rangle, & \gamma = 0 \\ -\gamma \log \sum_{A \in A_{n,n}} exp(-\frac{\langle A, \Delta(x,y) \rangle}{\gamma}), & \gamma \geq 0 \end{cases} \tag{3}$$

By default the γ hyperparameter is set to 1.

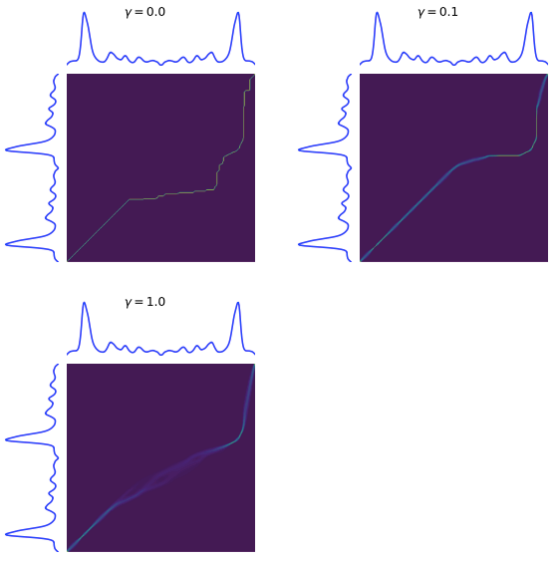

Figure 2.1 Soft-DTW hyperparameter behaviour

[6] Cuturi,Blondel:"Soft-DTW: a Differentiable Loss Function for Time-Series"

3.2 K-means algorithm

Partitioning methods, such as K-means and K-medoids use an iterative way to create the clusters by moving data points from one cluster to another, based on a distance measure, starting from an initial partitioning. [7].

Stepping into time series clustering, since the procedure has been applied to 30-minute averages and 6-minute averages, even if the number of data points is substantially large (365 multivariate time series with 240 daily observations for 6-minute averages) K-means remains computationally attractive. The complexity of each iteration of the K-means algorithm performed on 365 time series is $O(k\times365)$. This linear complexity is one of the reasons for the popularity of the k-means clustering algorithm [8].

The soft-DTW is used in K-means algorithm to assign the series to the clusters and to upload the centroids of the cluster. Centroids in a cluster correspond to the multivariate time series that minimizes the sum of the similarity measures between that time series and all time series inside the cluster. Given the 365 multivariate time series each of them composed by daily observations (48 with 30-minute averages, 240 with 6-minute averages) for both flow and density dimensions the algorithm work as follow:

Algorithm $k-means clustering\ \ (T,k)$
$Input: T=(t_1,t_2,...,t_{365})$
$Input: k\ \ the\ \ number\ \ of\ \ clusters$
$Output: c_1,c_2,...,c_k\ \ (set\ \ of\ \ cluster\ \ bi-dimensional\ \ centroids)$
$p=0$
$Randomly\ \ choose\ \ k\ \ objects\ \ and\ \ make\ \ them\ \ as\ initial\ \ centroids\ \ (c_1^{(0)},c_2^{(0)},...,c_k^{(0)})$
repeat
$Assign\ \ each\ \ data\ \ point\ \ to\ \ the\ \ cluster\ \ with\ \ the\ \ nearest\ \ centroid\ \ using\ \ soft-DTW$
$p=p+1$
// **Centroid update**
for $j=1\ \ to\ \ k$ **do**
$Update\ \ the\ \ centroid\ \ c_j^{(p)}\ \ of\ \ each\ \ cluster\ \ using\ \ soft-DTW$
end for
until
$c_j^{(p)}\approx c_j^{(p-1)}\quad j=1,2,...,k$
$Return\ \ c_1,c_2,...,c_k$

Sometimes different initializations of the centroids lead to very different final clustering results. To overcome this problem the K-means algorithm is run 5 times with different centroids randomly placed at different initial positions. The final results will be the best output of the 5 times consecutive runs in terms of inertia.

[7] K-medoids clustering is very similar to the K-means clustering algorithm. The major difference between them is that while a cluster is represented with its center in the K-means algorithm, it is represented with the most centrally located data point in a cluster in the K-medoids clustering

[8] Soheily-Khah:" Generalized k-means based clustering for temporal data under time warp" Chapter 3

3.3 Number of Optimal Clusters, K

K-means requires number of clusters K, as clustering parameter. Getting the optimal number of clusters is very significant in the analysis. If,for example, K is too high each time series starts representing a own cluster.

There is no a unique approach for finding the right number of clusters, I take in account a clustering quality measure, the soft-DTW similarity measure between nearest centroids and an empirical method to fix a minimum number of time series in each cluster.

Since Soft-DTW is differentiable it could be also used as a function to evaluate the cohesion inside each cluster and the separation with respect to the nearest cluster. The silhouette coefficient is a measure of how similar a time series is to its own cluster (cohesion) compared to other clusters (separation). The silhouette can be computed with the soft-DTW metric, it takes values in the range [-1, 1] [9].

Assume that the time series have been clustered via K-means. For time series $t_i \in C_k$ (time series i in the cluster C_k) $a(t_i)$ is the mean distance between time series t_i and all other time series in the same cluster:

$a(t_i) = \frac{1}{|C_k|-1} \sum_{j \in C_k} DTW^{\gamma}(t_i, t_j)$.

$b(t_i)$ is defined as the mean dissimilarity of the time series $t_i \in C_k$ to the nearest cluster C_z (where $C_k \neq C_z$) as the mean of the distance from the time series t_i to all time series $\in C_z$:

$b(t_i) = \min_{k \neq z} \frac{1}{|C_z|} \sum_{j \in C_z} DTW^{\gamma}(t_i, t_j)$.

Finally the silhouette coefficient for a time series is computed as follow:

$s(t_i) = \frac{b(t_i)-a(t_i)}{max\{a(t_i),b(t_i)\}} \quad if \quad |C_k| > 1$

The coefficients for every time series are averaged to have a global measure. The Mean Silhouette Coefficient for all time series is computed for the different number of clusters considered in the K-means algorithm to see how the number of clusters afflicts the analysis.

I consider also the soft-DTW similarity measure between the nearest clusters, by taking in account their centroids and applying equation (2) :

- When $K = 2$ becomes $DTW^{\gamma}(C_1, C_2)$ where C_1 and C_2 are centroids of the clusters.
- When $K = d$ where $d \geq 2$, $DTW^{\gamma}(C_j, C_z)$ is applied to all possible binary combinations of centroids forming a symmetric matrix of dimension $(d \times d)$ in which the minimum similarity between two centroids is selected.

As the number of clusters K increases the nearest clusters become closer each other until values of soft-DTW are stabilazed 17.

In addition as the number of clusters increases is more difficult to interpret the traffic behaviour captured by each cluster. To deal with this problem, following an empirical method [10], I fixed a lower bound for the minimum number of observations in each cluster approximately at $\sqrt{\frac{3}{2} * N}$. Where N is the number of daily time series (365). If the number of time series within a cluster is

[9] Rousseeuw: "Silhouettes: a Graphical Aid to the Interpretation and Validation of Cluster Analysis" pag 53-65

[10] Zhang et.Al: "An empirical study to determine the optimal k in Ek-NNclus method" chapter 1

lower than the fixed bound the cluster does not generalize well a particular traffic path, thus the interpretability of the cluster is too difficult.

The code is implemented with the Python machine learning library for time series **tslearn** [11].

4 Result

In this chapter graphical results are showed. To generalize the traffic dynamics within a cluster the centroids [12] using inverse_transform method} of the train set are plotted. To identified traffic patterns over the year a graphical tool is used: **calplot** library [13]. In addition to verify the path in different period of the year and to validate the representation of the centroids weekly series of flow and density are illustrated.

The centroids trained on a particular detector are both used on train data (flow and density of that detector) and on test data (flow and density of the other detector). In addition the centroids estimated on S60 2013 data are further validated on S60 2014 data. As the number of cluster increases the generalization on the test data becomes more difficult thus I only report the grapichal results on test set for two clusters and three clusters.

The procedure is repeated both for 30-minute averages and 6-minute averages, a general tendency is found for both detectors: Until three clusters K-means algorithm with 30-minute averages and 6-minute averages data gives the same results (same days in same cluster). As the number of clusters is greater than three the two aggregation times give different results. For 6-minute averages only these lasts results are presented.

4.1 S60 detector 30-minute averages

For the S60 detector the number of clusters showed is from 2 to 5. The plot of the soft-DTW similarity measure between nearest clusters (nearest centroids) in relation with the number of cluster is reported to confirm that after 5 clusters, the algorithm creates clusters that do not generalize well the traffic path over the year. Furthermore one of them contains a number of days lower than the fixed lower bound.

4.1.1 Two clusters

For two clusters on S60 detector the algorithm separates well the working days and no-working days including weekends and Public holidays. From the centroids in Figure 4.1 I can recognize free flow in the first cluster $k = 0$ and peaks in the morning (7-9) and in the afternoon (16:30-18:30) in the density centroid of the second cluster $k = 1$.

[11] Tavernard et.Al: "Tslearn, A Machine Learning Toolkit for Time Series Data"
[12] $q(x,t)$ in (veh/100)/h and $\rho(x,t)$ in (veh/Km)
[13] https://pypi.org/project/calplot/

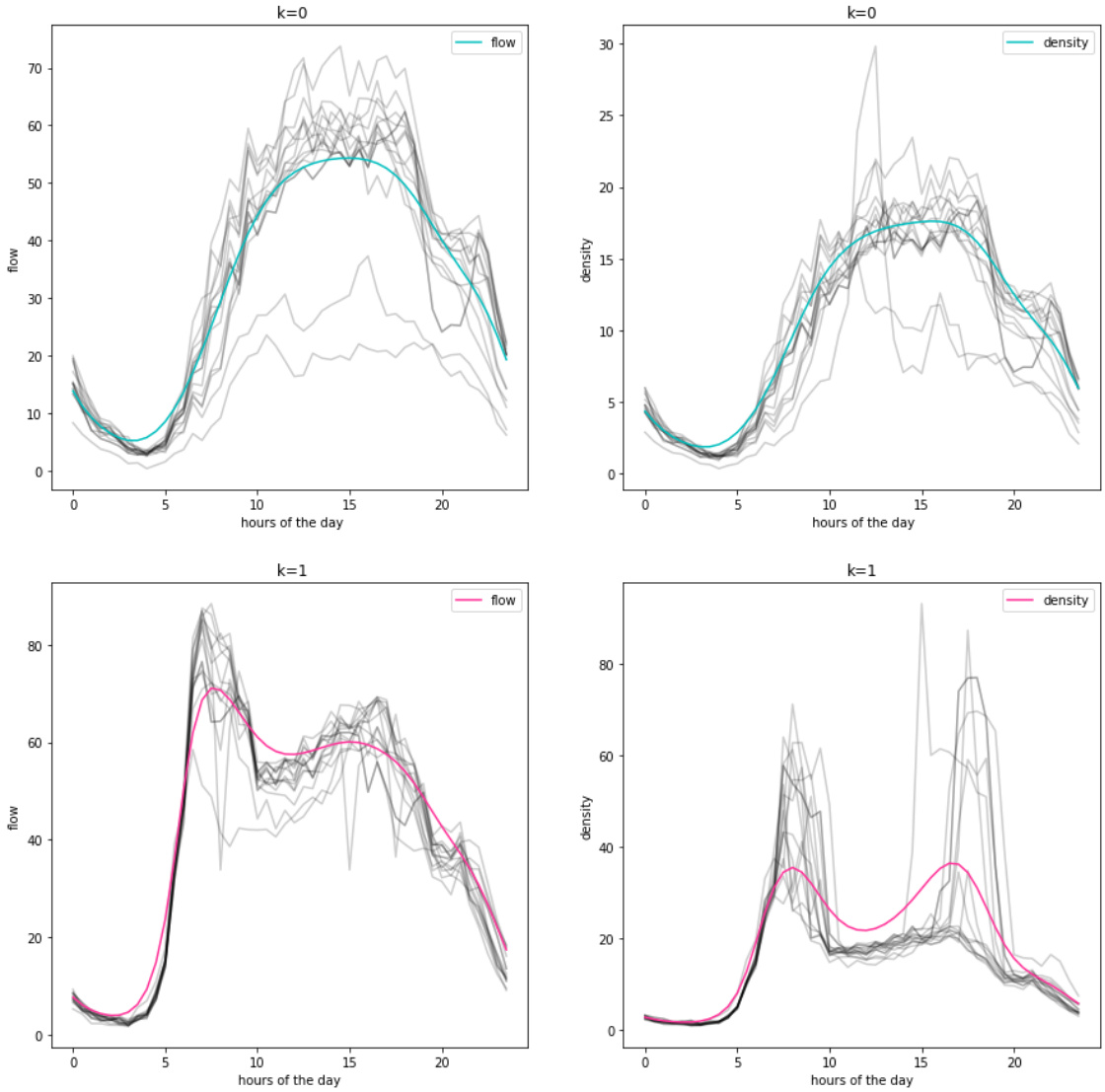

Figure 4.1 Centroids of S60 detector two clusters

As showed in Figure 4.2 I clearly see this path over the year, despite some working days classified as no-working days and Saturdays as working days (red circle). To check this days a section is created in Appendix(36). In addition Public holidays are marked in (green circle) to better visualize the difference between the two clusters. [14]

[14]01-01-2013 First of the year. 21-01-2013 Martin Luther King's Day. 27-05-2013 Memorial Day. 04-07-2013 Indipendence Day. 02-09-2013 Labor Day. 28-11-2013 Thanksgiving Day. From 23-12-2013 to 31-12-2013 Christmas Holidays.

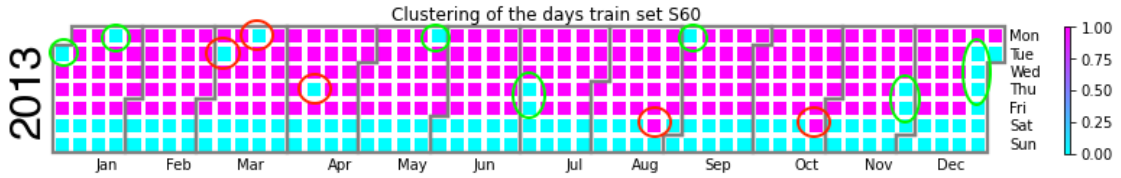

Figure 4.2 Clustering on S60 detector two clusters

The S60 centroids are then applied to test data: S1816 2013 data in Figure 4.3, S60 2014 data in Figure 4.3.5. In general working days and no-working days are well distinguished in both situation. Some working days are classified as no-working days due to a different dynamics of the traffic during working days on the two roads segments as showed later 18. For two clusters the k-means algorithm shows good results for both unseen data that varies or in spatial or in temporal dimension.

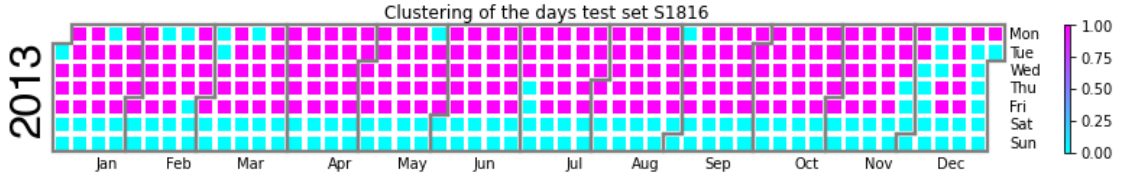

Figure 4.3 Clustering on S1816 detector, with respect to S60 centroids two clusters

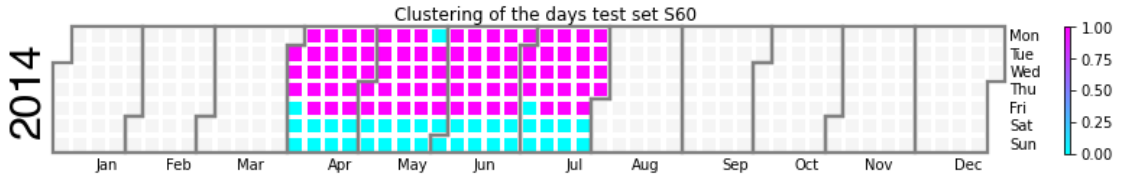

Figure 4.3.5 Clustering on S60 2014 detector, with respect to S60 2013 centroids two clusters

4.1.2 Three clusters

In Figure 4.4, for three clusters on S60 detector the first cluster $k = 0$ represents again a free flow situation in no-working days (weekends and Public holidays).The second cluster $k = 1$ represents a low level of traffic during working days, while the third cluster $k = 2$ represents a greater level of traffic during working days with respect to the second cluster $k = 1$. Despite the second $k = 1$ and third $k = 2$ cluster have similar behaviour in term of flow centroids, they differ mostly for the traffic congestion of the afternoon (16:30-18:30). The peak in the density centroid in the third cluster $k = 2$ is greater than 50 veh/km while the peak in the density centroid in the second cluster $k = 1$ is around 25 veh/km. The second $k = 1$ and third $k = 2$ cluster have similar behaviour in the morning congestion (7-9) where the density centroids reach the same level (35 veh/km).

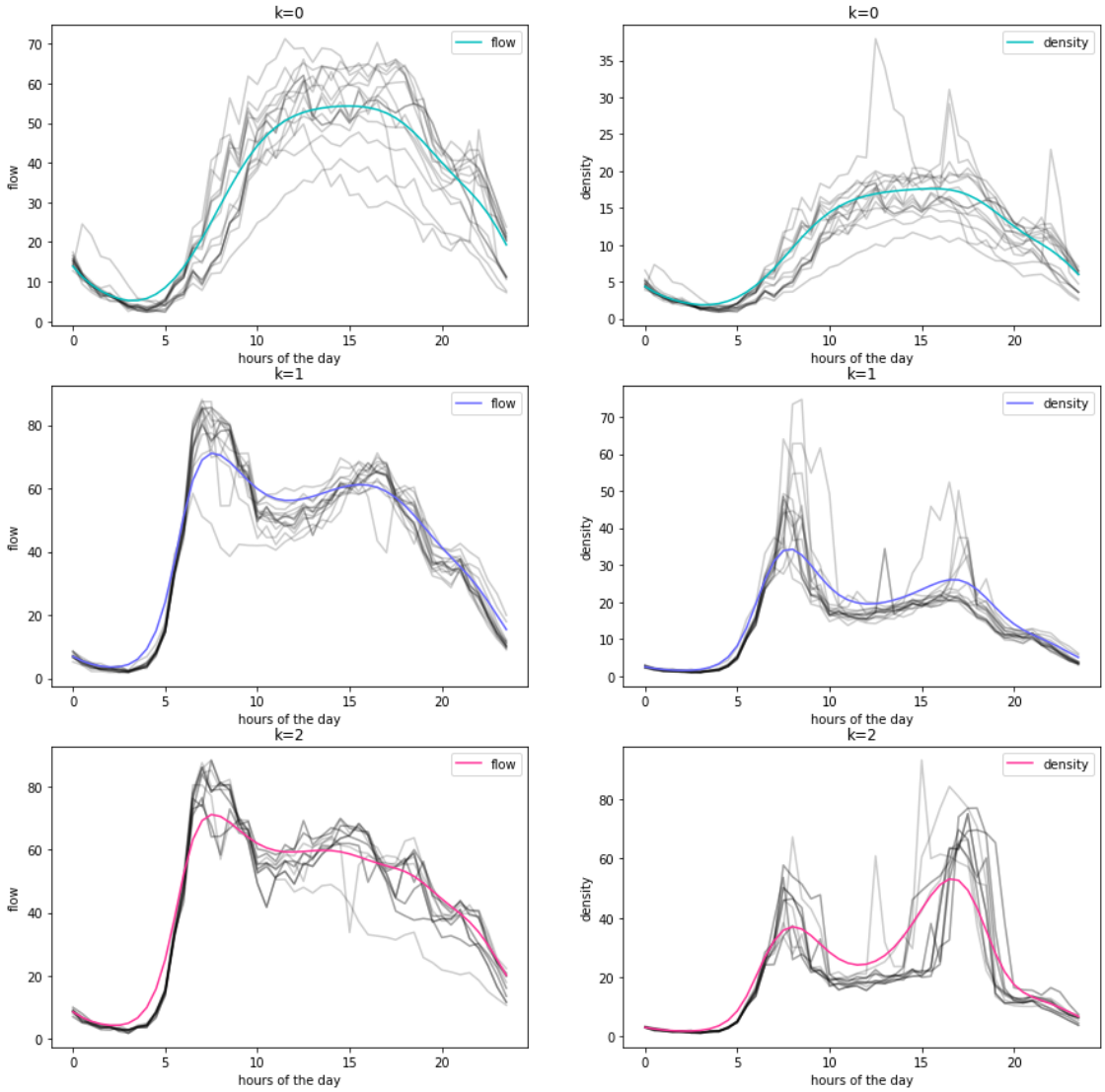

Figure 4.4 Centroids of S60 detector three clusters

In Figure 4.5 I can see the seasonal behaviour of the traffic on S60 detector. According to the K-means algorithm the most trafficated months are June, July, August and December where the majority of the days are assigned to the third cluster $k = 2$. In addition the majority of Mondays are assigned to the second cluster $k = 1$. To further validate the seasonal trends In Figure 4.6 the weekly time series (marked in green) of different months are plotted together.

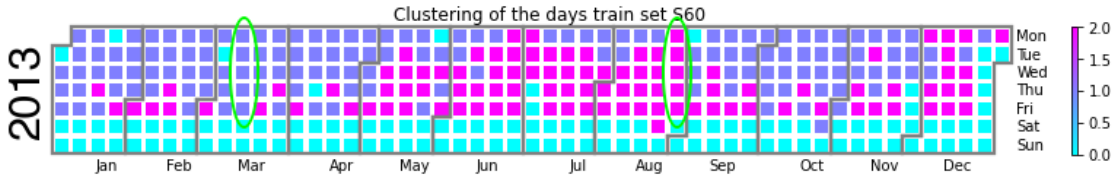

Figure 4.5 Clustering on S60 detector three clusters

In Figure 4.6 the density series from Monday to Friday are compared. The March's series (from 11/03 to 15/03), that belongs to the second cluster $k = 1$, presents an afternoon peaks lower than the one presents in August's series (from 26/08 to 30/08), that instead belong to the third cluster $k = 2$. This behaviour is captured by the algorithm, in fact the main difference between second $k = 1$ and third $k = 2$ cluster is due to the traffic congestion in the afternoon (16:30-18:30).

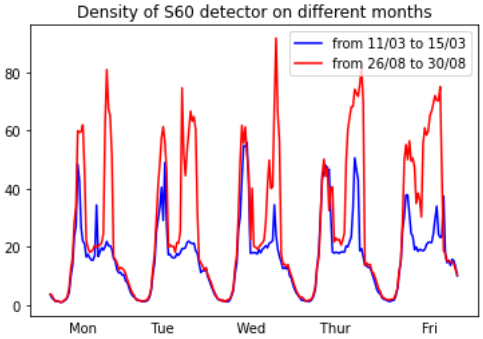

Figure 4.6 Density of S60 detector in a week of March and in a week of August

In Figure 4.7 the S60 centroids are then applied to S60 2014 data. During the working days I can recognize the seasonal behaviour with June and July more trafficated than April and May. In addition the majority of Mondays are assigned to the second cluster $k = 1$. The K-mean algorithm with three cluster gives consistent results when tested on unseen data that varies in the temporal dimension.

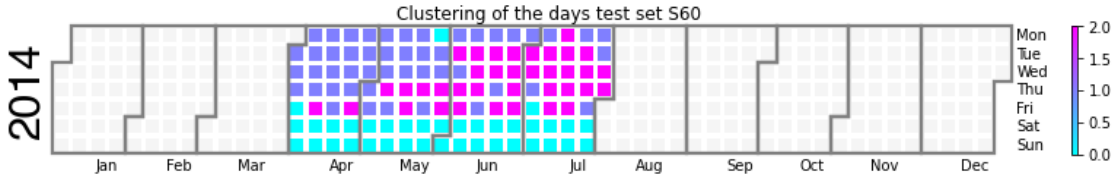

Figure 4.7 Clustering on S60 2014 detector, with centroids of S60 2013 three clusters.

In Figure 4.8 the S60 centroids are also applied to S1816 data. During working days traffic on S1816 is more represented by the third cluster $k = 2$ of S60 detector, only 30 days are classified inside the second cluster $k = 1$. No-working days continue to be correctly classified. Again to test the difference between second $k = 1$ and third $k = 2$ cluster of S60 detector on unseen data in Figure 4.9 the weekly time series of S1816 detector (marked in green) of different months are plotted together.

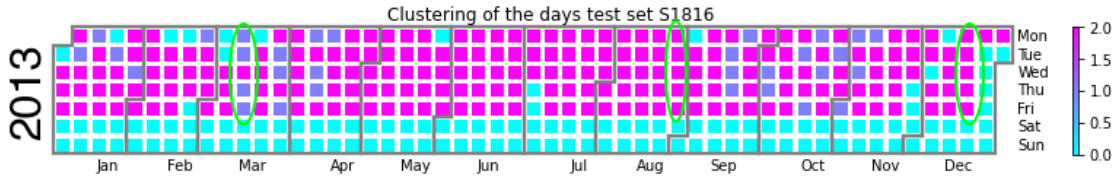

Figure 4.8 Clustering on S60 2014 detector, with respect to S60 2013 centroids three clusters

In Figure 4.9 the density series from Monday to Friday of S1816 are compared. The March's series (from 11/03 to 15/03) presents an afternoon peaks lower than the one presents in August's series (from 26/08 to 30/08), except for Wednesday 13/03 classified in the third cluster $k = 2$. The different behaviour in term of density levels during the traffic congestion in the afternoon (16:30-18:30) between second $k = 1$ and third $k = 2$ clusters is captured also on the test set.

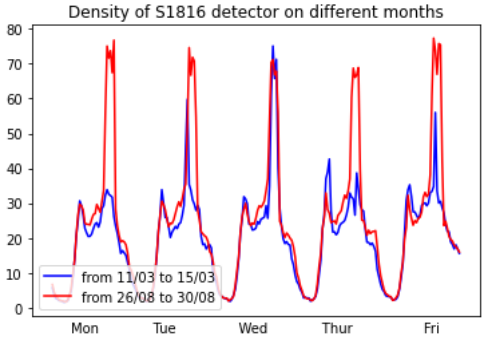

Figure 4.9 Density of S1816 detector in a week of March and in a week of August

4.1.3 Four clusters

In Figure 4.11, for four clusters on S60 detector the first cluster $k = 0$ represents again a free flow situation in no-working days.The fourth cluster $k = 3$ represents a low level of traffic during working days, while the second cluster $k = 1$ and the third ones $k = 2$ (similar each other) represent a greater level of traffic during working days with respect to the fourth cluster $k = 3$. The second $k = 1$, the third $k = 2$ and fourth $k = 3$ cluster have similar behaviour in term of flow centroids, in addition the three clusters have similar behaviour in the morning congestion (7-9) where the density centroids reach the same level (35 veh/km). The fourth cluster $k = 3$ differ mostly from the second $k = 1$ and third $k = 2$ one for the traffic congestion of the afternoon (16:30-18:30). The peak in the density centroid in the fourth cluster $k = 3$ is around 25 veh/km more flatted than density peaks reached in the afternoon by the second $k = 1$ and third $k = 2$ cluster. Despite the second $k = 1$ and third $k = 2$ cluster are the most similar ones they are distinguished by the level of traffic congestion in the afternoon (16:30-18:30) where the density centroid of the second cluster $k = 1$ reaches 40 veh/km while the density centroid of the third cluster $k = 2$ exceeds 50 veh/km.

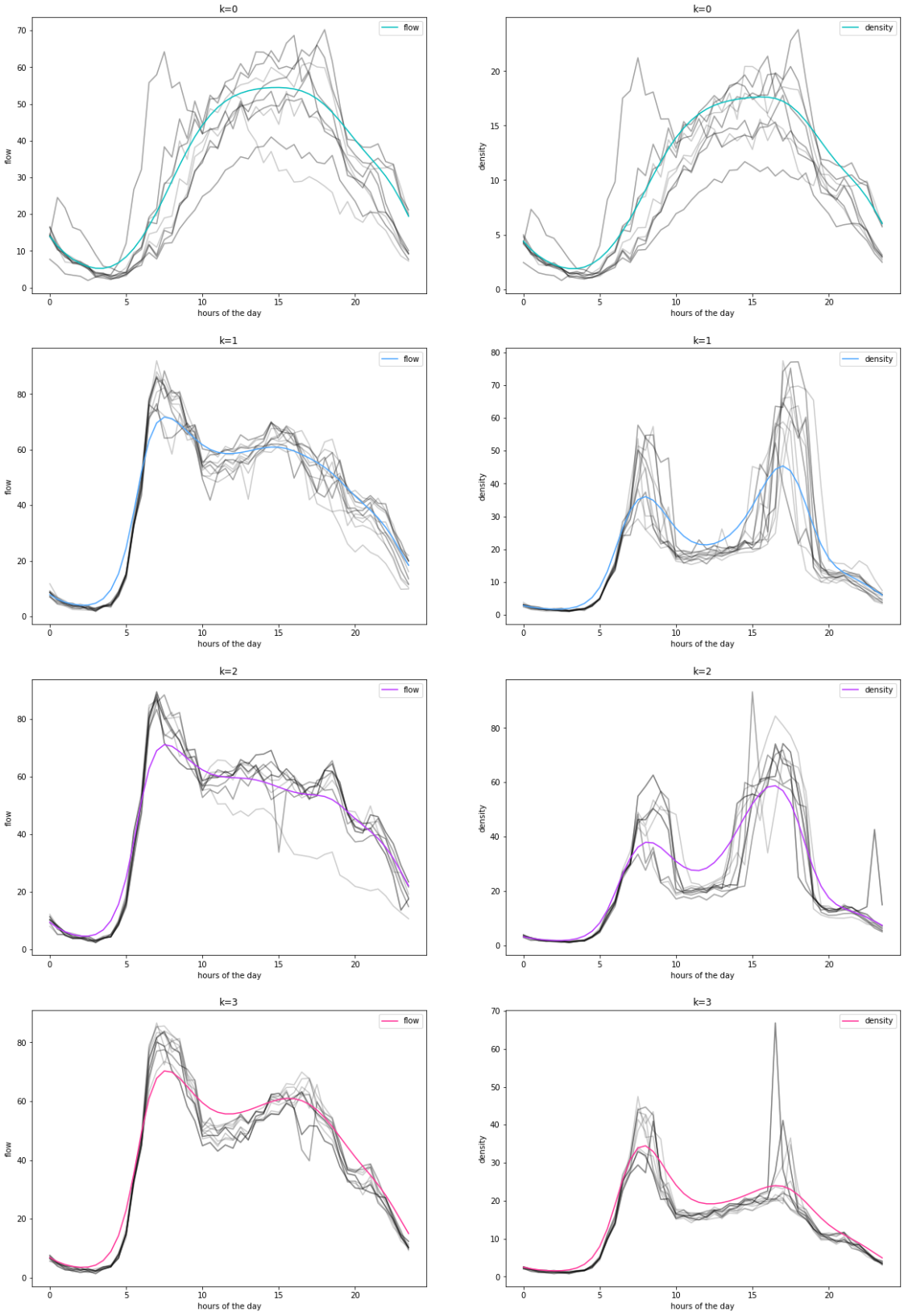

Figure 4.11 Centroids of S60 detector four clusters

In Figure 4.12 I can see the seasonal behaviour of the traffic on S60 detector. May,June, July August and December are mostly marked by the second $k = 1$ and the third $k = 2$ cluster which represent a significant level of traffic. On the opposite side February, March, April and October are mostly signed with the fourth cluster $k = 3$ that represents a lower level of traffic with respect to second $k = 1$ and third $k = 2$ cluster. In addition the majority of Mondays are assigned to the fourth cluster $k = 3$, while Fridays are assigned mostly to the second $k = 1$ and third $k = 2$ cluster. In order to understand better how the algorithms separate the second $k = 1$ and third $k = 2$ cluster (the most similar ones), in Figure 4.13 the four days time series (marked in green) of different months are plotted together.

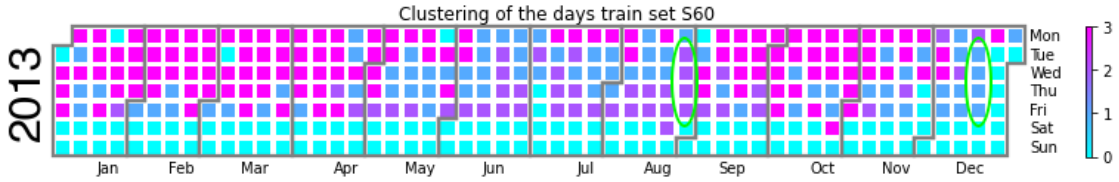

Figure 4.12 Clustering on S60 detector

In Figure 4.13 the density series from Tuesday to Friday of different months are compared. The August's time series (from 27/08 to 30/08) ,that belongs to the third cluster $k = 2$, presents visible higher peaks in the afternoon in relationship with the December's time series (from 17/12 to 20/12), that instead belongs to the second cluster $k = 1$. This behaviour is captured by the K-means algorithm as the main difference between third $k = 2$ and second $k = 1$ cluster. I can see different levels of traffic congestion in the afternoon (16:30-18:30), where the peaks of the third cluster $k = 2$ (magenta) are greater than the one in the second cluster $k = 1$ (blue).

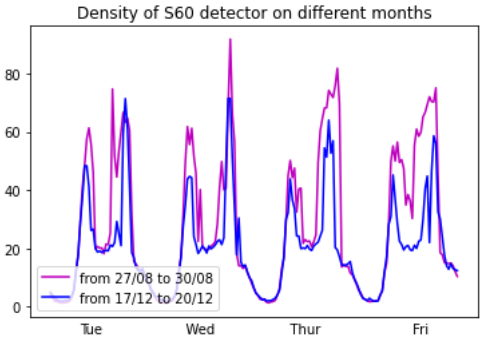

Figure 4.13 Density of S60 detector in a week of August and in a week of December

4.1.4 Five clusters

In Figure 4.14, for five clusters on S60 detector, despite the algorithm continues to separate well no-working days from the other days, the third cluster $k = 2$ contains only 8 days. Following the lower bound rule fixed in section (3.3) the third cluster $k = 2$ does not to generalize a particular traffic dynamic over the year, it is difficult to interpret the traffic path capture by the third cluster $k = 2$.

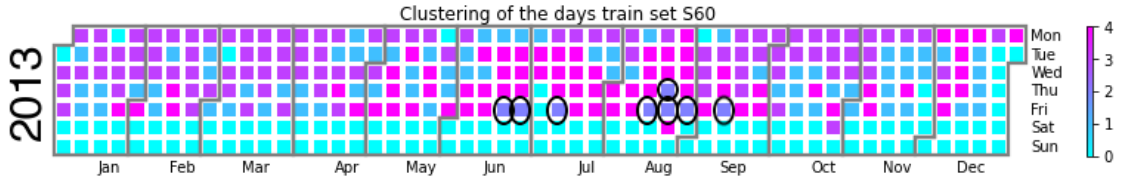

Figure 4.14 Clustering on S60 detector five clusters

In Figure 4.15 the line of the soft-DTW similarity measure between nearest clusters in relation with the number of cluster considered flattens markedly after 5 clusters showing that after this number of clusters the algorithm does not generalize well the traffic dynamic over the period considered.

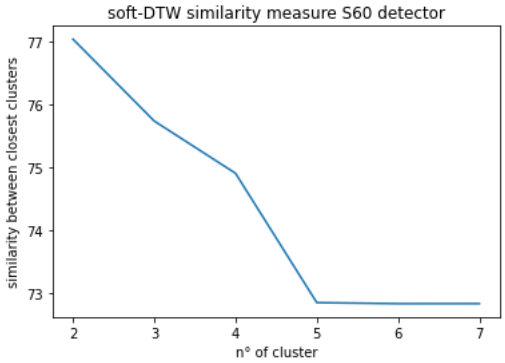

Figure 4.15 Soft-DTW similarity measure between closest cluster in relationship with the number of cluster

4.2 S1816 detector 30-minute averages

For the S1816 detector the number of clusters showed is from 2 to 5. The plot of the soft-DTW similarity measure between nearest clusters (nearest centroids) in relation with the number of cluster flattens significantly after 8 clusters however the lower bound for the minimum number of days in each clusters is reached after 5 clusters.

4.2.1 Two clusters

For two clusters on S1816 detector the algorithm separates again the working days and no-working days including weekends and public holidays. Again in Figure 4.16 the first cluster $k = 0$ shows a free flow traffic situation while in the density centroid of second cluster $k = 1$ I can clearly recognize a peak in the afternoon (16:30-18:30). Considering second clusters $k = 1$ from figure 4.16 (S1816 detector) and from figure 4.1 (3 S60 detector), they present different shape both in term of flow centroid and density centroid, as expected the algorithm captures different traffic dynamics during the working days in the two roads segments (I-35W and I-94).

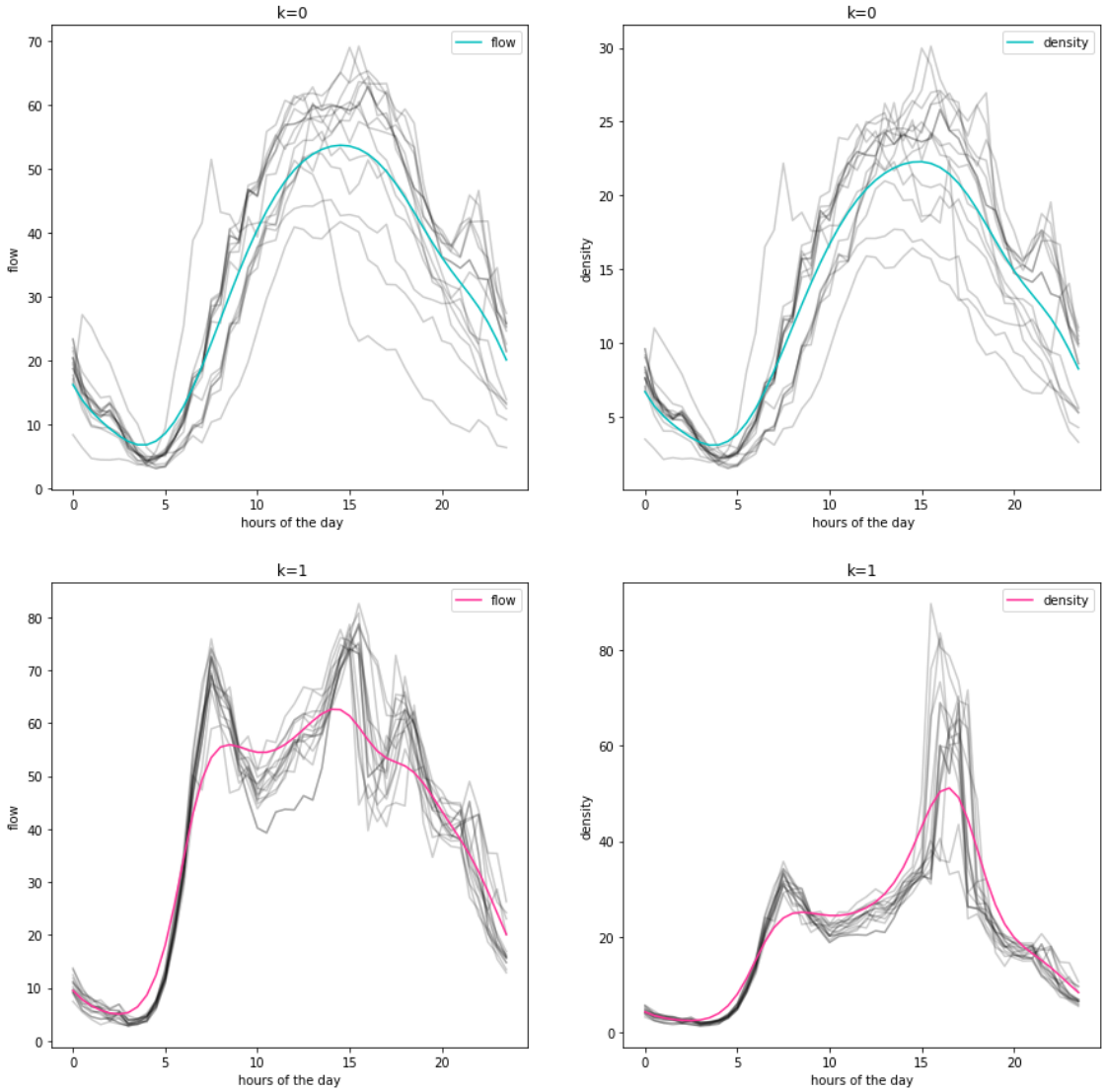

Figure 4.16 Centroids of S1816 detector two clusters

In Figure 4.17 The algorithm clearly divides working days and no-working day over the year, despite some working days classified as no working days (red circle). To check this days a section is created in Appendix{ 41}. The Public holidays are marked (green circle) for a better visualization. [15]

[15] 01-01-2013 First of the year. 21-01-2013 Martin Luther King's Day. 27-05-2013 Memorial Day. 04-07-2013 Indipendence Day. 02-09-2013 Labor Day. 28-11-2013 Thanksgiving Day. From 23-12-2013 to 31-12-2013 Christmas holidays.

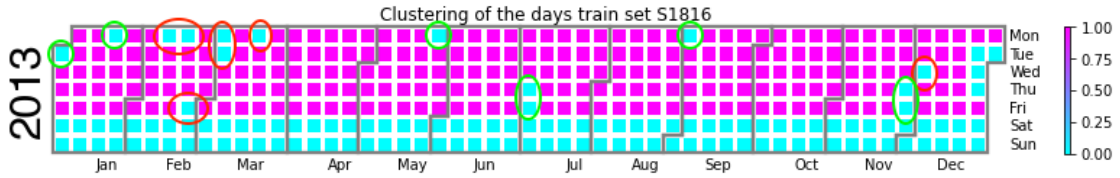

Figure 4.17 Clustering on S1816 two clusters

The centroids trained on S1816 detector are then applied to S60 detector with good result.

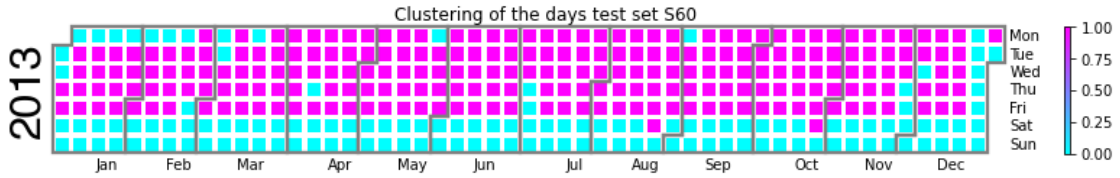

Figure 4.18 Clustering on S60, with S1816 centroids two clusters

4.2.2 Three clusters

In Figure 4.19, for three clusters on S1816 detector the first cluster $k = 0$ represents again a free flow situation in no-working days. The second cluster $k = 1$ represents a low level of traffic during working days, while the third cluster $k = 2$ represents a greater level of traffic during working days with respect to the second cluster $k = 1$. The second $k = 1$ and third $k = 2$ cluster mostly differ for the traffic congestion of the afternoon (16:30-18:30). The peak in the density centroid in third cluster $k = 2$ is around 60 veh/km and at the same time the flow centroid presents an inflection point, while the peak in the density centroid in the second cluster $k = 1$ is lower than 50 veh/km and in at the same hours the flow centroid do not change its curvature. The second $k = 1$ and the third $k = 2$ cluster have similar behaviour in the morning congestion (7-9) where the density centroids reach the same level (25 veh/km).

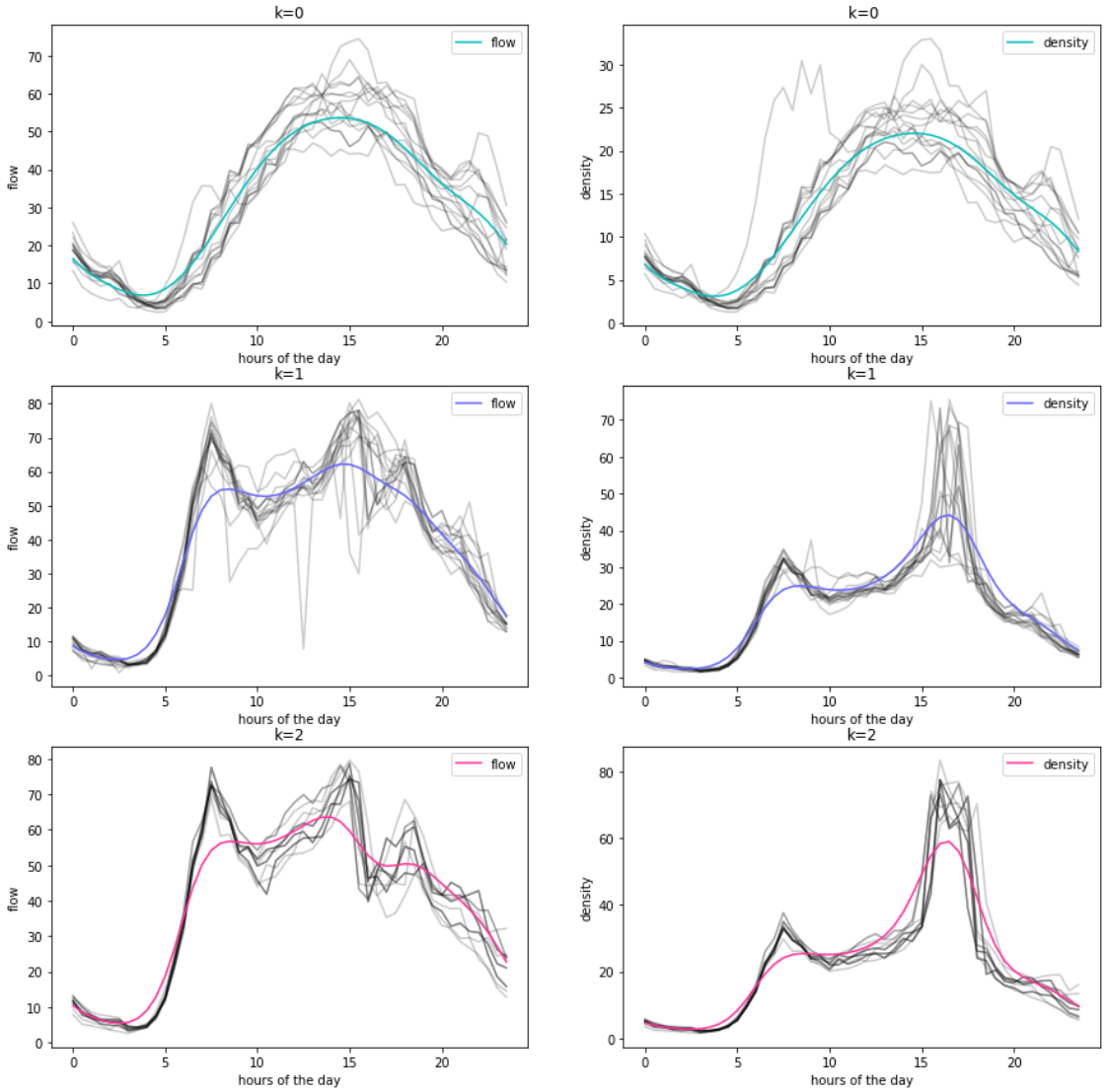

Figure 4.19 Centroids of S1816 detector three clusters

In Figure 4.20 I can see the seasonal behaviour of the traffic on S1816 detector. According to the clustering algorithm the most trafficated periods are from May to September and the month of December where the majority of the days are assigned to the third cluster $k = 2$. To further validate the seasonal trends In Figure 4.21 the weekly time series (marked in green) of different months are plotted together.

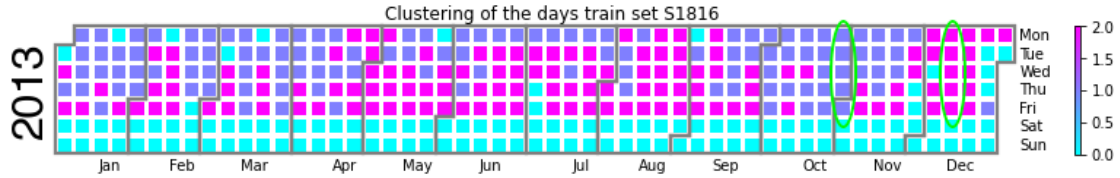

Figure 4.20 Clustering on S1816 detector three clusters

In Figure 4.21 the density series from Monday to Friday of S1816 are compared. The October's series (from 28/10 to 01/11),that belongs to the second cluster $k = 1$, presents an afternoon peaks lower than the one presents in December's series (from 09/12 to 13/12),that belongs instead to the third cluster $k = 2$. The different behaviour in term of density levels during the traffic congestion in the afternoon (16:30-18:30) between second $k = 1$ and third $k = 2$ cluster is captured by the algorithm.

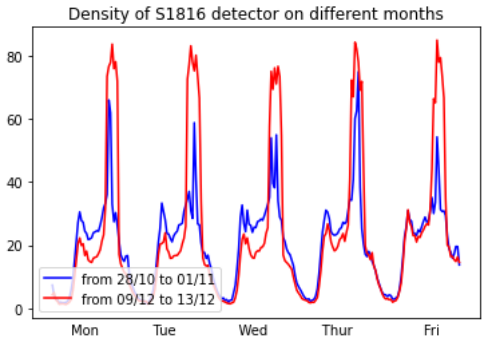

Figure 4.21 Density of S1816 detector in a week of October and in a week of December

In Figure 4.23 The S1816 centroids are also applied to S60 data. Working days traffic on S60 is more represented by the second $k = 1$ cluster of S1816 detector, moreover the majority of monday continue to be classified in the second cluster $k = 1$ as low traffic working days . No-working days (weekends and Public holidays) continue to be correctly classified. The majority of days classified in the third cluster $k = 2$ are in May, June, July, August and December, signed as more trafficated months by the S60 centroids 8.

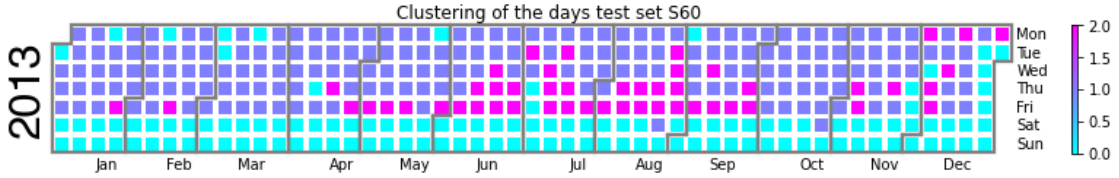

Figure 4.23 Clustering on S60 detector, with S1816 centroids three clusters

4.2.3 Four clusters

In Figure 4.24, for four clusters on S1816 detector the first cluster $k = 0$ represents again a free flow situation in no-working days (weekends and Public holidays).The fourth cluster $k = 3$ represents a low level of traffic during working days, while the second cluster $k = 1$ and the third one $k = 2$ (similar each other) represent a greater level of traffic during working days. The second $k = 1$ and the third $k = 2$ cluster have similar behaviour in term of shape in the flow centroids. Both of them presents a inflection point during the traffic congestion of the afternoon (16:30-18:30), while the density centroids reach the maximum. Despite the same shape of centroids flow, the second cluster $k = 1$ presents a lower level of flow during the day (7-19) than the third cluster $k = 2$. The second $k = 1$, the third $k = 2$ and the fourth $k = 4$ cluster have similar behaviour in the morning congestion (7-9) where the density centroids approximately reach the same level (25 veh/km). The fourth cluster $k = 3$ differ mostly from the second $k = 1$ and third $k = 2$ one for the traffic

congestion of the afternoon (16:30-18:30). The peak in the density centroid in the fourth cluster $k = 3$ is around 40 veh/km more flatted than density peaks reached in the afternoon by the second $k = 1$ and third $k = 2$ cluster. In addition the fourth cluster $k = 3$ does not present any inflection point in the flow centroid during the afternoon peak in the density centroid.

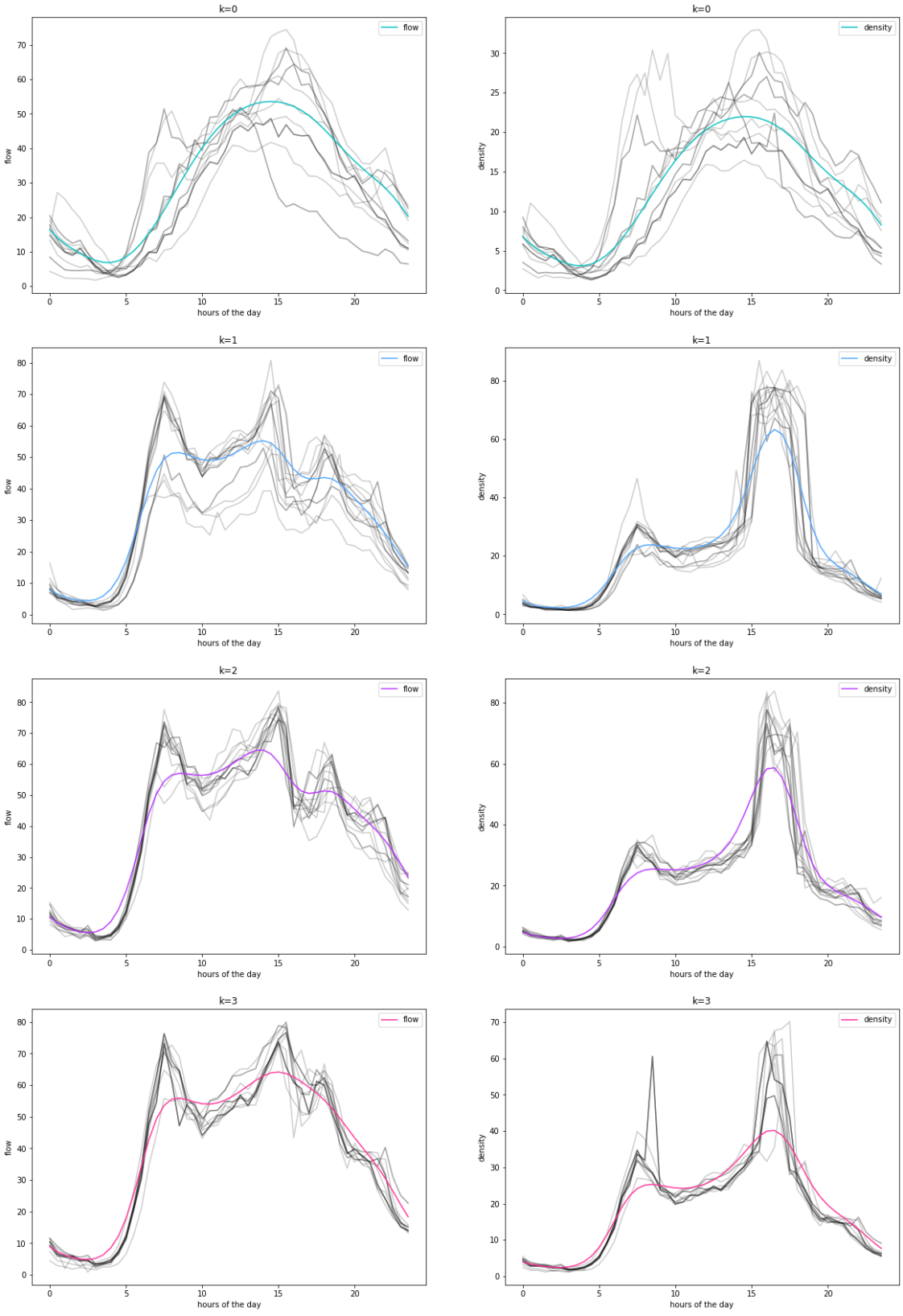

Figure 4.24 Centroids of S1816 detector four clusters

In Figure 4.25 I can see the seasonal behaviour of the traffic on S1816 detector. May, June, July and August are mostly marked by the third cluster $k = 2$. The second cluster $k = 1$ mostly represents December (do not count the last 10 days of Christmas holidays). These two clusters present a higher level of traffic with respect to the fourth cluster $k = 3$ which indeed marks the majority of the working days over the year. In order to understand better how the algorithms separate the second $k = 1$ and third $k = 2$ clusters (the most similar ones), in Figure 4.26 the weekly time series (marked in green) of different months are plotted together.

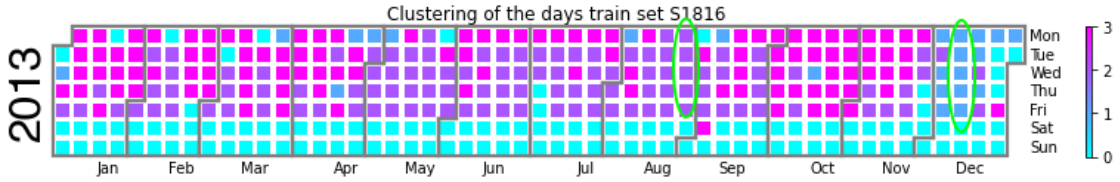

Figure 4.25 Clustering on S1816 detector four clusters

In Figure 4.26 the flow series from Monday to Friday of S1816 are compared. The August's series (from 26/08 to 30/08), that belongs to the third cluster $k = 2$, presents an higher level of traffic flow during the day (7-19) than the one presents in December's series (from 09/12 to 13/12),which belongs instead to the second cluster $k = 3$. The different behaviour in term of flow levels during the day between second $k = 1$ and third $k = 2$ cluster is captured by the K-means algorithm.

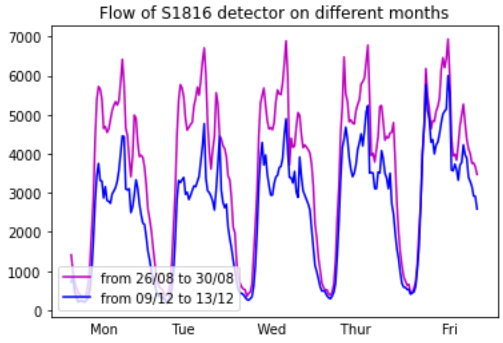

Figure 4.26 Flow of S1816 detector in a week of August and in a week of December

4.2.4 Five clusters

In Figure 4.27, for five clusters on S1816 detector, the algorithm separate Sundays and Public holidays from Saturdays. Altough the first cluster $k = 0$ and the fourth $k = 3$ one have a free flow situation during the day (7-19), the first cluster $k = 0$ (Sundays and Public holidays) have lower values of flow and density than the ones present in the fourth cluster $k = 3$ (Saturdays).The fifth cluster $k = 4$ represents a low level of traffic during working days, while the second cluster $k = 1$ and the third $k = 2$ ones (similar each other) represent a greater level of traffic during working days with respect to the fifth cluster $k = 4$. The second $k = 1$ and the third $k = 2$ clusters are equivalent to the second $k = 1$ and third $k = 2$ ones built with number of clusters equal to four 25. Moreover the fifth one $k = 4$ is the same as the fourth one $k = 3$ built with number of clusters equal to four. Apparently the algorithm passing from 4 to 5 clusters keeps unchanged three clusters.

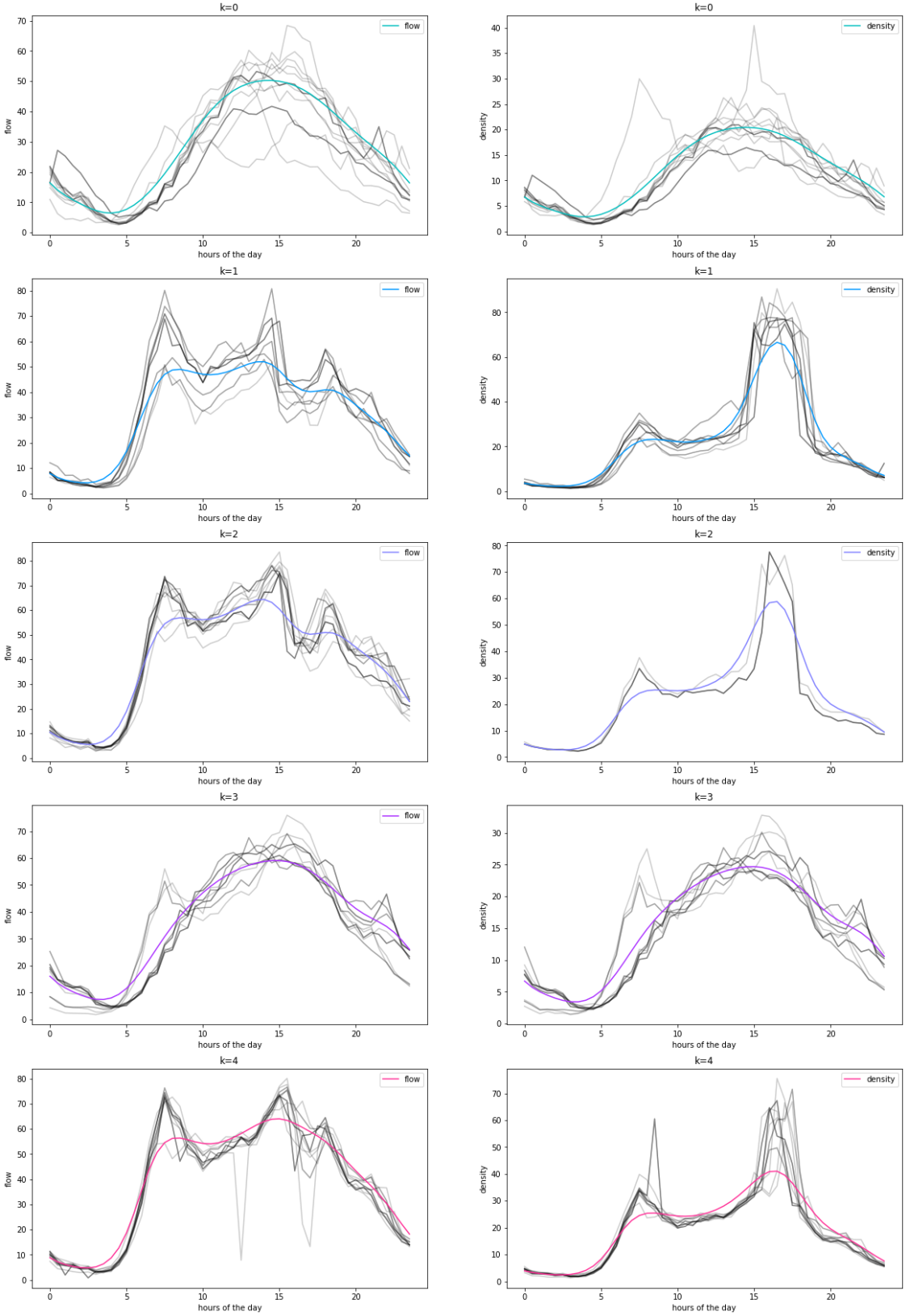

Figure 4.27 Centroids of S1816 detector five clusters

In Figure 4.25 I can see the clustering procedure on S1816 detector. As mentioned before the first cluster $k = 0$ captures the traffic dynamic of Sundays and Public Holidays (green circle). The fourth cluster $k = 3$ marks mostly the Saturdays. The third cluster $k = 2$ represents summer months. The second cluster $k = 1$ (black circle) identifies the month of December (do not count last 10 days of Christmas holidays). The fith cluster $k = 5$ represents low level of traffic during working days with respect to second $k = 1$ and third $k = 2$ clusters. The fifth cluster $k = 4$ is observed mainly in January and October and in Mondays over the years. The main difference between Fig 4.28 (Five clusters) and Fig 4.25 (26 Four clusters) is in the addition of a cluster to capture the traffic path of Saturdays.

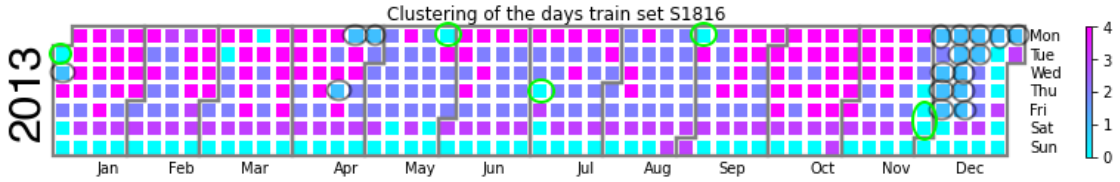

Figure 4.28 Clustering on S1816 detector five clusters

Regarding the second cluster $k = 1$ in Figure 4.28, it contains only 17 days. Since the same order of magnitude of the fixed lower bound (see section 3.3) is reached I do not present K-means algorithm with more than five clusters. By increasing the number of clusters the interpretation of these ones becomes complicated. However by looking at the line plot in Figure 4.29 of the soft-DTW similarity measure between nearest cluster in relation with the number of clusters I can notice that the line flattens after 8 clusters. In S1816 detector the lower bound (minimum number of days in a cluster) is not reached as in S60 detector when the line of soft-DTW similarity measure flattens (see Figure 4.15 17). This behaviour shows that to validate a clustering procedure multiple aspects should be considered such as interpretability of the clusters, distance between them and cohesion inside each of them.

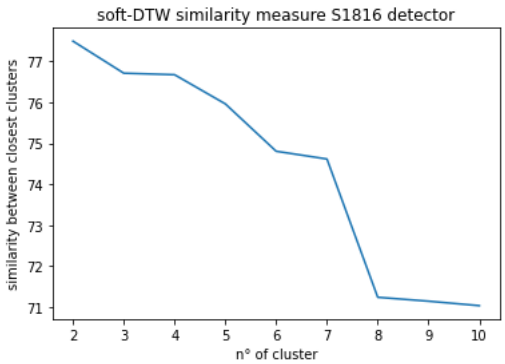

Figure 4.29 Soft-DTW dissimilarity measure between closest cluster in relationship with the number of cluster

4.3 S60 detector 6-minute averages

Although different data from S60 detector are considered (6-minute averages), the K-means algorithm gives same results of the 30-minute averages for two and three clusters (same days in same clusters). In particular for two clusters 6-minute averages differ from 30-minute averages only for

one day (24/08), while for three clusters 6-minute averages differ from 30-minute averages for 7 days. Moreover for two and three clusters, results on S1816 using S60 centroids 6-minute averages are the same (same days same clusters) of the ones obtained using S60 centroids 30-minute averages on S1816 .
When the number of clusters is increased to four, 6-minute averages data gives different results with respect to 30-minute averages data. As showed in Figure 4.30 the seasonal trend is confirmed with May,June, July, August and December most trafficated months (see Appendix {46}). However the fourth cluster $k = 3$ just includes 16 days (lower bound reached) without representing properly a particular traffic path over the year.

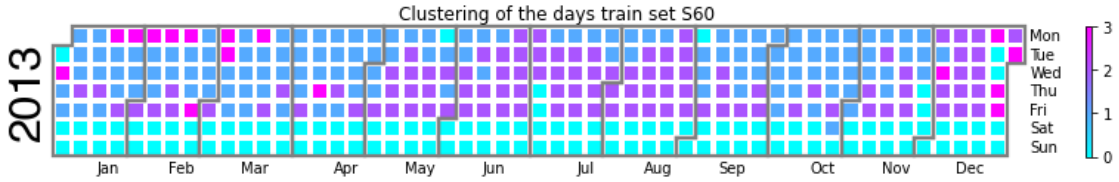

Figure 4.30 Clustering on S60 detector 6-minute averages, four clusters

4.4 S1816 detector 6-minute averages

By considering data from S1816 detector as 6-minute averages, the K-means algorithm gives same results of the 30-minute averages for two and three clusters (same days in same clusters). For two clusters 6-minute averages differ from 30-minute averages only for one day (04/12), while for three clusters 6-minute averages differ from 30-minute averages for 11 days.

Moreover for two and three clusters, results on S60 using S1816 centroids 6-minute averages are the same (same days same clusters) of the ones obtained using S1816 centroids 30-minute averages on S60.

When the number of clusters is increased to four, 6-minute averages data gives different results with respect to 30-minute averages data. As showed in Figure 4.31 the difference between the second cluster $k = 1$ and third cluster $k = 2$ is less marked than in 30-minute averages (see Figure 4.25 26). The seasonal trend is confirmed with May,June, July, August and December most trafficated months (see appendix 47), but the interpretability of the clusters with respect to traffic paths becomes harder.

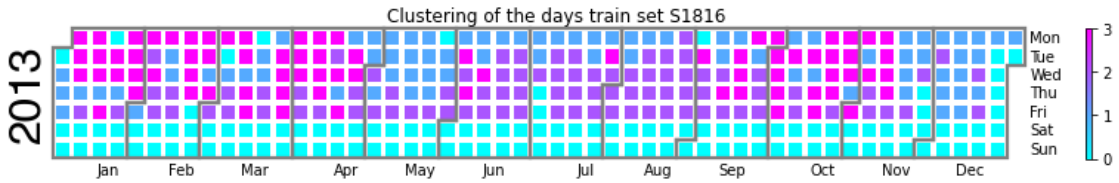

Figure 4.31 Clustering of S1816 detector 6-minute averages, four clusters

When the number of cluster considered is five, 6-minute averages data do not separate Sundays and Public holidays from Saturdays as in 30-minute averages data (see Figure 4.8 11). In Figure 4.32 K-means algorithm put these days inside the same cluster, the first one $k = 0$. Furthermore the

fourth cluster $k = 3$ just represent 11 days (lower bound reached) without representing properly a particular traffic path over the year.

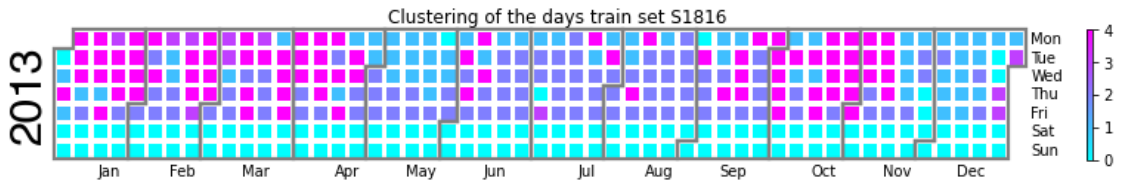

Figure 4.32 Clustering of S1816 detector 6-minute averages, five clusters

4.5 Silhouette Coefficients

To validate the clustering procedure from another prospective the silhouette coefficients are computed in Tab 1,2,3,4. As explained earlier (chapter3) the coefficients represent the averages of all 365 multivariate time series. The coefficients are computed both for train set (detector used to estimate the centroids) and test set (the other detector) for the two aggregation time 6-minute and 30-minute. The number of clusters considered is from 2 to 5 as the ones showed earlier.

In all tables I can distinguish the same trend: as the number of clusters increases the silhoeutte coefficients are reduced.

The greater values of the quality measure are reached with two clusters. By only following the silhouette coefficient and picking up two as the most suitable number for K, in K-means algorithm, I can not separate the seasonal behaviour of the traffic over the year marked with three clusters in both detector.

Comparing 30-minute averages with 6-minute averages I can not confirm that one time aggregation works better the the other, despite as showed before for 6-minute averaged the interpretability of the four and five clusters K-means procedure becomes harder 31. As already mentioned is important to validate the procedure via multiple perspectives.

The final decision on how many clusters consider should be made by the user considering different metrics as for example the ones indicate in this project.

	TRAIN 2013	TEST 2013
K	S1816 30min	S60 30min
2	0.746	0.6614
3	0.5148	0.5692
4	0.5248	0.5708
5	0.4878	0.4567
6	0.3621	0.3905

	TRAIN 2013	TEST 2013	TEST 2014
K	S60 30min	S1816 30min	S60 30min
2	0.6863	0.7291	0.7765
3	0.5884	0.4646	0.6538
4	0.4895	0.3606	0.5485
5	0.432	0.2271	0.477
6	0.405	0.2576	0.4249

Tab 1,2 Silhoeutte coefficients 30-minute averages data for S60 and S1816 detectors

	TRAIN 2013	TEST 2013
K	S1816 6min	S60 6min
2	0.7307	0.6543
3	0.4940	0.5483
4	0.4297	0.4771
5	0.36359	0.4333
6	0.3648	0.4604

	TRAIN 2013	TEST 2013
K	S60 6min	S1816 6min
2	0.6832	0.7300
3	0.5597	0.5127
4	0.5096	0.4887
5	0.5057	0.3144
6	0.4567	0.2608

Tab 3,4. Silhoeutte coefficients 6-minute averages data for S60 and S1816 detectors

5 Conclusion and Outlook

Through this project the K-means algorithm with soft-DTW is preliminary tested on multivariate road traffic time series. After having decide a pre-process strategy the main challenge is to decide the optimal number of clusters. For both detectors analyzed S60 and S1816, and for both aggregation time considered 6-minute and 30-minute three clusters seems to be a good compromise between quality of the clustering procedure and differentiation of traffic dynamics over a year of time. By applying the centroids trained on a detector on the other detector two clusters is the best choice. When instead the centroids trained on S60 detector are applied to the same segment road but in a different time interval (next year) also three clusters works properly. 6-minute averages and 30-minute averages data give same result for two and three clusters, by increasing the number of clusters to four and five, despite silhouette coefficients are in the same order of magnitude, clusters of 6-minute averages become harder to interpret or some of them contain too few days with respect to 30-minute averages. Since the K-means algorithm is highly dependent from the traffic analyzed by changing the period of time considered no more one year but for example 6 months, different results are expected both in term of optimal number of clusters and performance regarding the different aggregation time.

6 References

[1] Pedregosa, Varoquaux, Gramfort, Michel, Thirion. “Scikit-learn: Machine Learning in Python”.

[2] Sardà-Espinosa. “Comparing Time-Series Clustering Algorithms in R Using the dtwclust Package”.

[3] Sakoe, Chiba. “Dynamic programming algorithm optimization for spoken word recognition,” IEEE Transactions on Acoustics, Speech and Signal Processing.

[4] Cuturi, Blondel. “Soft-DTW: a Differentiable Loss Function for Time-Series”.

[5] Saeid Soheily-Khah. “Generalized k-means based clustering for temporal data under time warp”.

[6] Rousseeuw. “Silhouettes: a Graphical Aid to the Interpretation and Validation of Cluster Analysis”.

[7] Zhang, Bouadi, Martin. “An empirical study to determine the optimal k in Ek-NNclus method”.

[8] Tavenard, Faouzi, Vandewiele, Divo, Androz, Holtz, Payne, Yurchak. “Tslearn, A Machine Learning Toolkit for Time Series Data”.

[9] Treiber, Martin, Arne Kesting. “Traffic flow dynamics.”

[10] The code implemented is available at https://github.com/NicolaRonzoni/Multivariate-Time-series-clustering.git

7 Appendix

7.1 S60 detector misclassification check

In this section possible days misclassified by K-means are analyzed. The days are marked in red in Fig 4.2 4. To better understand the reason why some working days are classified as no-working days and some no-working days classified as working days the weekly time series from Monday to Sunday of flow or density is considered.

On 05/03/2013 low level of flow during all day, classified as no-working day.

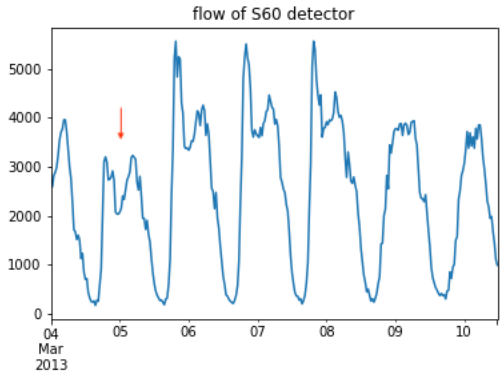

Figure 7.1 Weekly flow time series from 04/03 to 10/03

On 18/03/2013 low level of flow during all day, classified as no-working day.

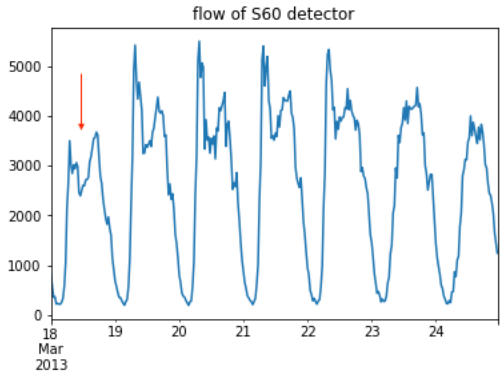

Figure 7.2 Weekly flow time series from 18/03 to 24/03

On 11/04/2013 anomalous peak in the density series in the morning (7-9) followed by low level of density (11-15), classified as no-working day.

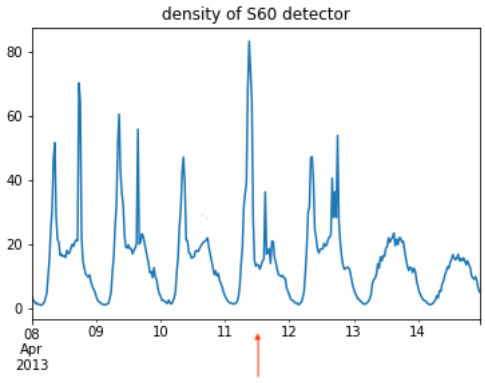

Figure 7.3 Weekly density time series from 08/03 to 14/03

On 24/08/2013 small peak of the flow series during the morning (7-9), classified as working day.

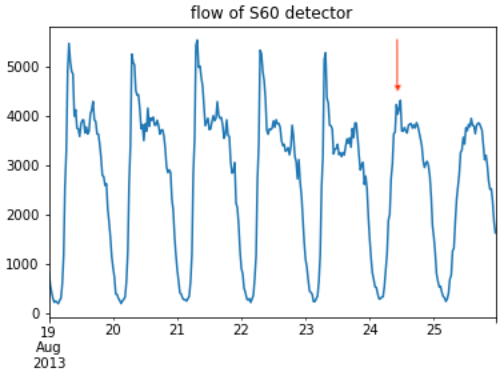

Figure 7.4 Weekly density time series from 19/08 to 25/08

On 26/10/2013 peaks in the density series in the morning (7-9) and in the afternoon (16:30-18:30) as a working day.

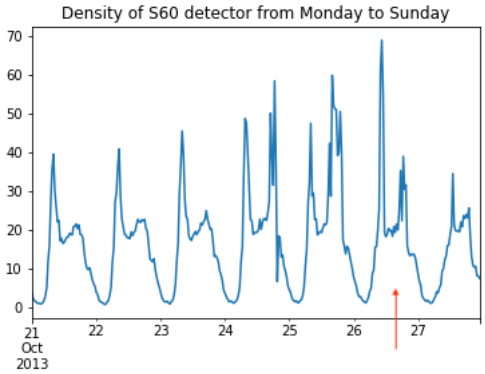

Figure 7.5 Weekly density time series from 21/10 to 27/10

7.2 S1816 detector misclassification check

In this section possible days misclassified by K-means are analyzed. The days are marked in red in Fig 4.17 19. To better understand the reason why some working days are classified as no-working days the weekly time series from Monday to Sunday of flow or density is considered.

On 11/02/2013 density series with the same level of density as a no-working day.

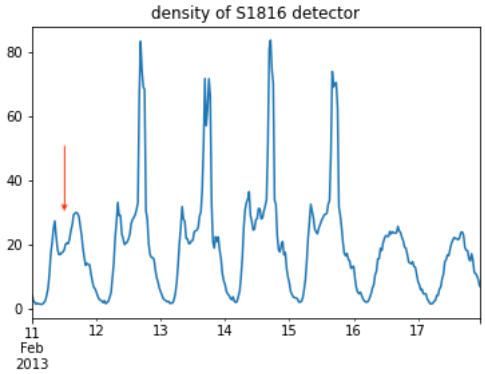

Figure 7.6 Weekly density time series from 11/02 to 17/02

On 18/02/2013 and on 22/02/2013 density series with the same level of density as a no-working day.

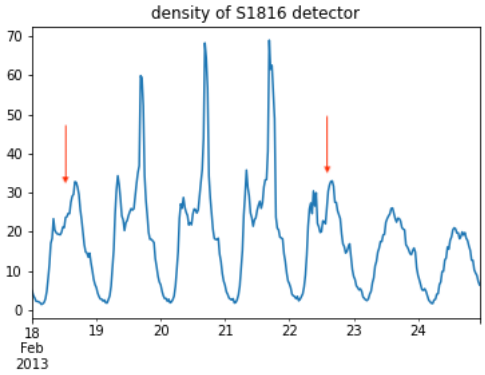

Figure 7.7 Weekly density time series from 18/02 to 24/02

On 04/03/2013 and 05/03/2013 density series with the same level of density as a no-working day.

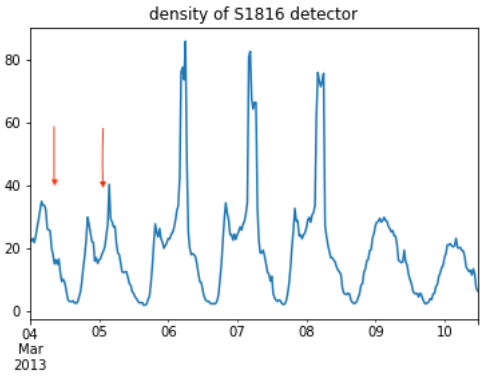

Figure 7.8 Weekly density time series from 04/03 to 10/03

On 18/03/2013 peak of the afternoon (16:30-18:30) in the density series lower than the other working days of the week, classified as no-working day.

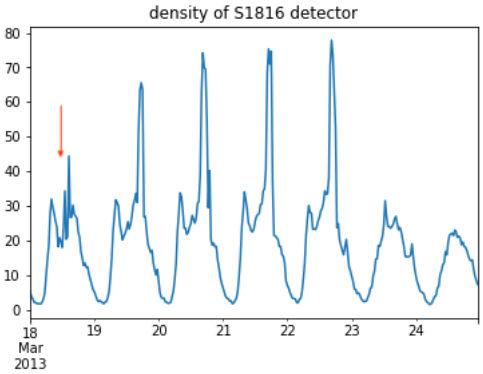

Figure 7.9 Weekly density time series from 18/03 to 24/03

On 04/12/2013 low level in the flow series, classified as no-working days.

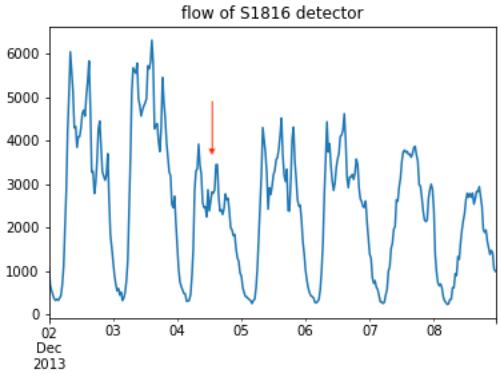

Figure 7.10 Weekly flow time series from 02/12 to 08/02

7.3 S60 detector: paths of the centroids 6-minute averages, four cluster

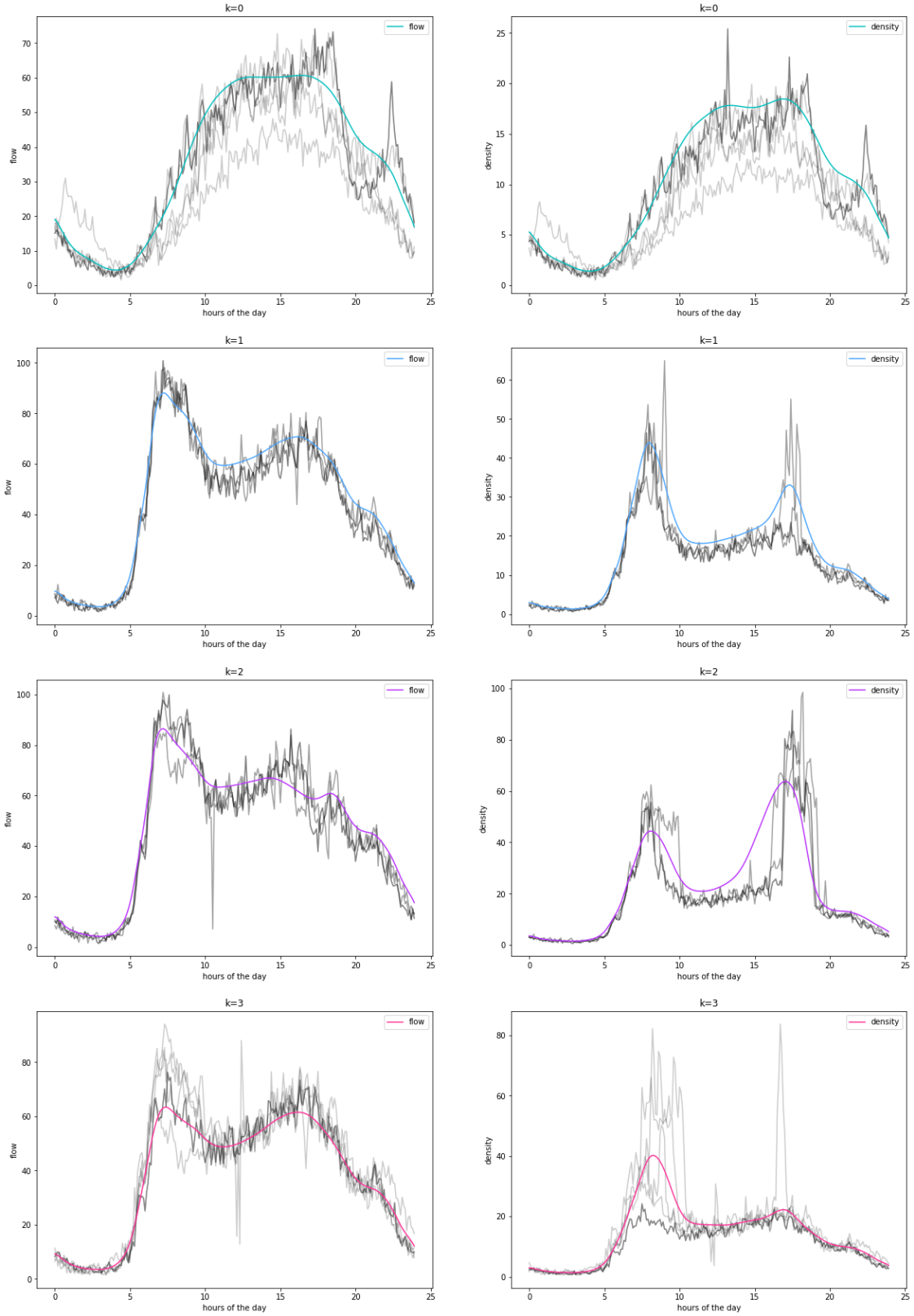

Figure 7.11 Centroids on S60 detector 6-minute averages, four clusters

7.4 S1816 detector: paths of the centroids 6-minute averages, four cluster

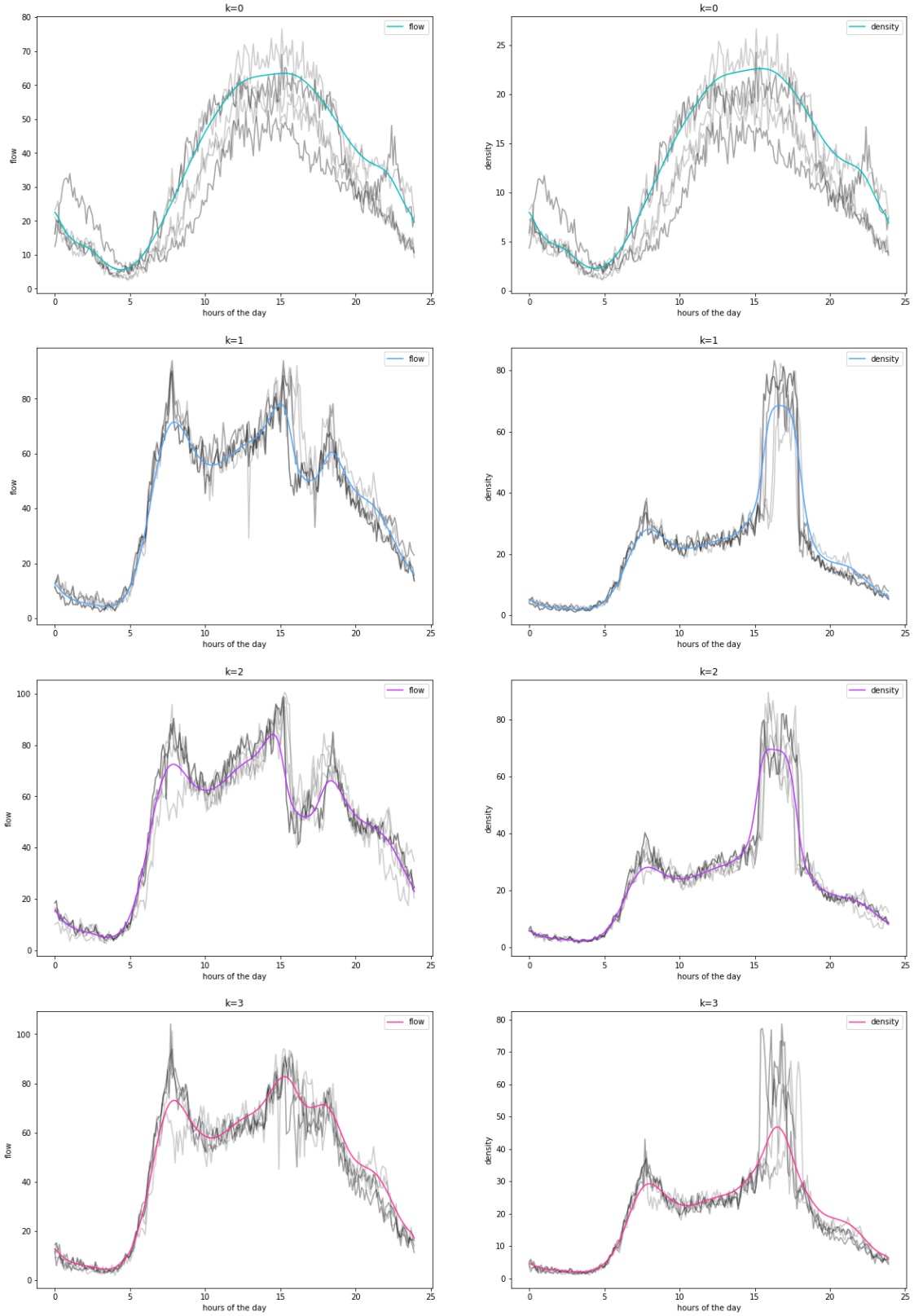

Figure 7.12 Centroids on S1816 detector 6-minute averages, four clusters