Mini-Project (ML for Time Series) - MVA 2024/2025

Alexis Marouani alexis.marouani@polytechnique.edu
Grégoire Béchade gregoire.bechade@polytechnique.edu

December 17, 2024

1 Introduction and contributions

The paper we studied aims to introduce a novel method to perform anomaly detection in large time series, based on the introduction of a "normal behaviour" of the time series. Anomaly detection can be defined in several manners. It can be interpreted as "outlier detection", where the objective is to detect single anormal points (which can for instance correspond to sensor failures). The paper focuses on the second type of task, which aims to detect anormal subsequences of the time series. In this case, several consecutive points behave anormaly (for instance, an anormal heartbeat or an anormal day of electricity consumption). The authors decided to consider that an anomaly sequence is a sequence that can be observed several times in the time-series, in opposition to another definition that would consider an anomaly as a unique sequence. The method aims to define the expected behaviour of the time series, and defines anomalies as subsequences that are too far from this expected behaviour. The real contribution of this new method is not an improvment in terms of accuracy, but in computational time. Indeed, the approximation of the expected behaviour prevents to compute the distances between all pairs of subsequences.

We decided to reproduce the algorithm described in the paper, to implement some variations, and test it on new data.

We decided to split the work as follows : Grégoire made the data extraction and vizualization and Norma1. Alexis did Norma2 and our personal variation.

As the source code was not available, we reimplemented everything ourselves.

Experiments: Several algorithms from the original papers were run, and our own variation using k-means instead of hierarchical clustering was implemented. A new method to sample the subsequence was also tested and significantly improves the results.

2 Method

Classical methods for anormal sequences detection rely on the fact that anormal sequences are considered as unique sequences in the time-series. These methods label as anomalies the sequences that are far away from every other subsequence of the time series, for a given metric (euclidian, DTW for instance). However, as explained in the article, one can be looking for anomalies that are not unique, like anormal heartbeats in a ECG. The method proposed introduces the **normal model** (N_M), which is a sequence of fixed length (here $3 \times l$, with l the length of the anomalies we are looking for), that represents the "normal behaviour" of the model. The proposed method to determine the N_M is :

1. Extract the subsequences of length $3 \times l$ (randomly or with motif selection).
2. Perform a hierarchical clustering of the subsequences.
3. Select the cluster c in $\mathbb{C}$ (the set of clusters) that maximises the following quantity : $N(c) = \frac{frequency(c)^2 \times coverage(c)}{\Sigma_{x \in \mathbb{C}} dist(center(c), center(x))}$
   With $frequency(c)$ the number of sequences in the cluster, $coverage(c)$ the time lag between the first and the last sequences in the cluster, and $center(c)$ the barycenter of the cluster. This quantity aims to select the cluster with the most sequences, and that is close to all the other clusters.
4. Finally, N_M is defined as the barycenter of the cluster c.

The distance of a subsequence to the normal model is then defined as the minimal (euclidian) distance between a subsequence os size l and all the subsequences in the normal model of size l (as a recall, the normal model is of size $3 \times l$). The authors describe several metric to label a subsequence as *anormal*. Selecting the k subsequences with the biggest minimal distance to the normal model, or selecting the ones that are above a certain threshold can be methods to determine the anomalies.

The key point of the method is that instead of comparing an anomaly to all the subsequences of the time-series and then comparing it to a very large number of identical pattern, the clustring step enables to drastically reduce the number of comparisons. Indeed, each sequence of size l is compared to $2 \times l$ sequences in the normal model, instead of $T - 2 \times l$ in the whole time series (with the condition of non overlap).

We decided to implement the algorithm from scratch to try to reproduce the results from the paper.

A first implementation was tested, trying to reproduce exactly the algorithm from the paper (**Norma1 without seasonal sampling**).

Seasonal sampling :
We noticed that the sampling was not optimal, as sequences beginning at different hours were compared with an euclidian distance, which did not seem relevant. For the other algorithms described above, we decided to perform a samplig which took into account the seasonality of the time series as follows : We randomly selected approximately 50 subsequences of size $7 \times 48 = 336$ from the original time-series. Indeed, as the final step of the clustering consists in determining a barycenter, and as there is a strong seasonality in the time series, we don't want to sum a subsequence beginning at 7:00 and a subsequence beginning at 23:00, as it would probably converge toward a flat distribution. The sampling was then performed on days, in order to have subsequences beginning at the same time.

We decided to fix a size of 7 days for the normal model to catch a "normal week".

Three algorithms were tested with this seasonal sampling :

1. First, we tried to reproduce the algorithm from the paper, with hierarchical clustering and selecting the centroïd of the most representative cluster as the normal model, as explained above (**Norma1 with seasonal sampling**).

2. Then, we tried another algorith from 5, which follows the same method but decides to define as the normal model the average mean of the centroids of each cluster, with the score of each cluster as weights (**Norma2 with seasonal sampling**). This approach aims at better catching the normal behaviour of the time series. Indeed, if there is two normal patterns with a very different shape in the time series, the first approach would label as anomaly every pattern of the second size, while the second approach integrates this information.

3. Finally, we implemented our own variations, with the use of k-means instead of hierarchical clustering. The first variation gets the centroid of the most representative cluster as the normal model (**K-means variation 1 with seasonal sampling**), as it is done in Norma1. The second variation defines the normal model as the averaged sum of the centroids of each cluster with respect to the criterion defined above (**K-means variation 2 with seasonal sampling**), as it is done in Norma2.

To evaluate our model, we computed the score of each subsequence of the origin time-series, defined as the minimal distance between this subsequence and each subsequence in the normal model, and selected the 10 most anormal subsequences.

3 Data

We decided to run the algorithm on the NYC taxis dataset, which represents the number of calls to taxis in New York City every 30 minutes between 01/07/2014 and 30/01/2015, available here. The dataset has 5 anomalies highlighted, that we aim to detect. This dataset seemed performant because of its big size : 10320 points. Its mean is 15137.57, with a standard deviation of 6931.92. The figure 1 shows a day of the time-series.

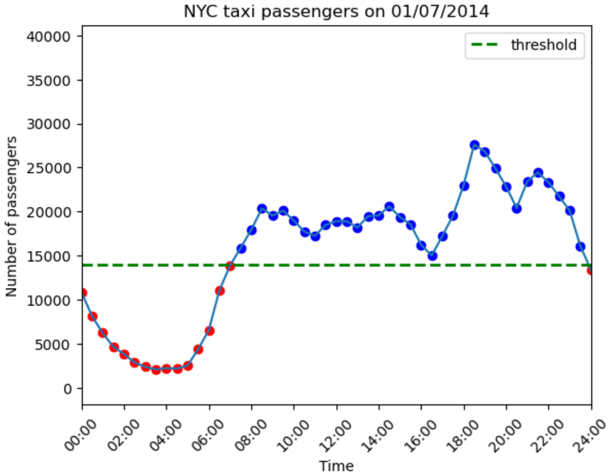
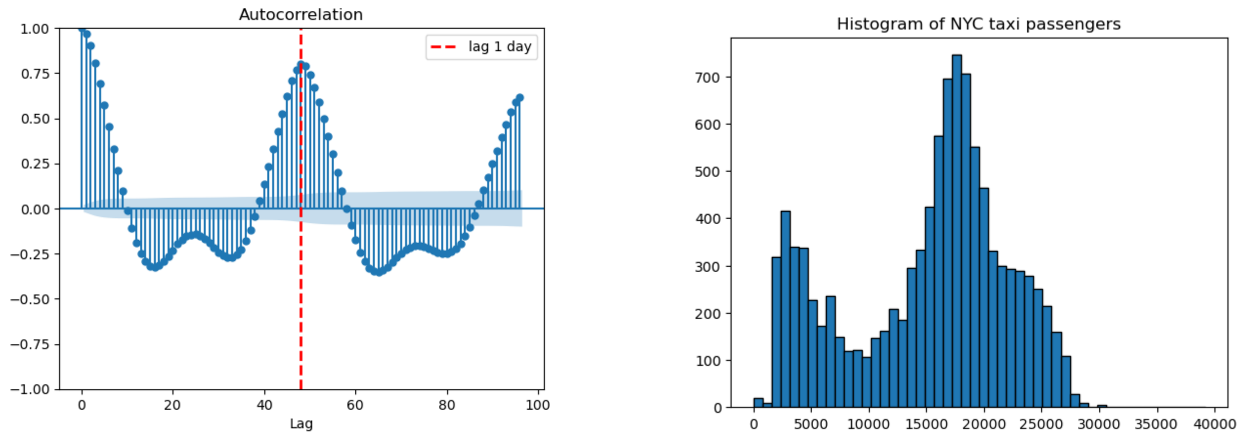


Figure 1: A day of the NYC taxis dataset



(a) Autocorrelation plot

(b) Histogram of values

Figure 2: Data analysis

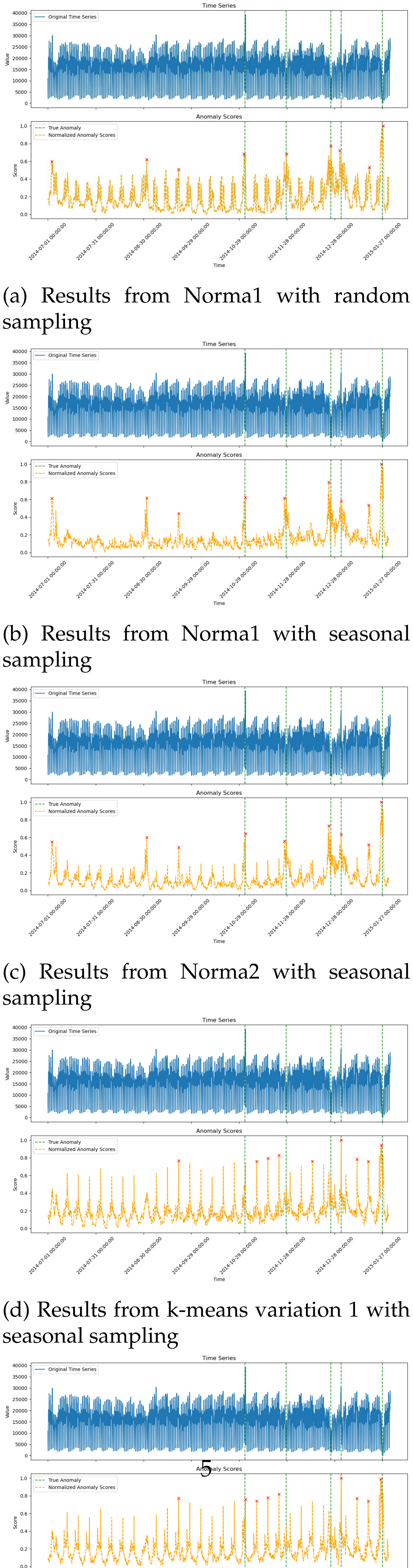
Augmented Dicky-Fuller and KSS tests showed p-values of respectively 2.5e-19 and 0.06, which confirms the stationarity of the time-series. The periodicity of the time-series was confirmed by an autocorrelation plot (see Fig.2a), with a peak every 48 points, corresponding to 24 hours time-lag. An histogram of the values taken by the time-series shows a bimodal distribution with few outliers, separated by the 14000 value (see Fig.2b). Values below 14000 correspond almost exactly to hours between 00:00 and 07:00, which is coherent with the physical meaning of the time-series (see figure 1).

4 Results

The three algorithms described in the methods were run on the taxi dataset. 10 anomalies were extracted, and the goal was to detect the 5 known anomalies in the dataset. First, one can notice that Norma1 has better results with seasonal sampling rather than random sampling, as can be noticed on the score curve, whose variance diminishes with seasonal sampling. Norma 1 and Norma 2 with seasonal sampling returns roughly the same results, and identify correctly the 5 anomalies which correspond to aldready detected anomalies in the time-series (Thanksgiving and Christmas leading to a huge increase in the number of calls to join the family at unexpected hours and New-York marathon or a snow storm leading to a decrease in the number of calls). New anomalies were also detected by those two algorithms. 5 also pointed out these new anomalies, and observed that they corresponded to very rainy days. The method seems relevant, as it enabled to detect new anomalies. One can also notice that the anomaly score is less regular in the Norma1, which is a good thing, as sundays should not be more anormal than mondays. The k-means variations algorithms do not manage to catch the expected anomalies. To conclude, Norma1 with seasonal variations algorithm outperforms the other algorithms tested.

5 References

Boniol, P., Linardi, M., Roncallo, F., Palpanas, T., Meftah, M., & Remy, E. (2021). Unsupervised and scalable subsequence anomaly detection in large data series. The VLDB Journal, 30(6), 909-931.



(a) Results from Norma1 with random sampling

(b) Results from Norma1 with seasonal sampling

(c) Results from Norma2 with seasonal sampling

(d) Results from k-means variation 1 with seasonal sampling