POLITECNICO MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE E DELL'INFORMAZIONE

Final Project

Voronoi Tessellation for Functional Anomaly Detection

Scientific computing tools for advanced mathematical modelling

Authors: Federica Botta, Simone Colombara, Michele Di Sabato

Academic year: 2021-2022

---

1. Mathematical formulation of the problem

Our project takes into consideration a particular dataset composed of a very large number of spatial points belonging to the left atrium (LA) of the heart.

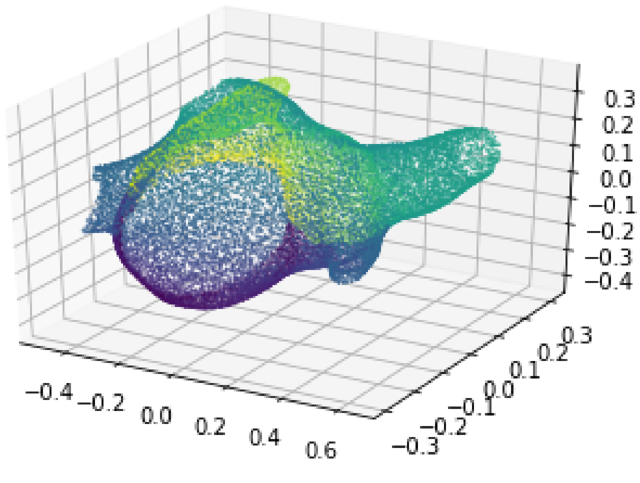


Figure 1: LA formed by the points of the dataset

The main activity of the heart is the mechanical contraction and expansion, these have their origin in the electrical activation of the cardiac cells. Indeed, at each heartbeat, myocytes are activated and deactivated following a characteristic electrical cycle. If a cell doesn't work in the right way could cause some cardiac problems, such as arrhythmias.
To check the health of the heart of a patient, many studies have been conducted on the propagation of the impulse of the potential through the heart and the propagation of this signal in space and time is recorded.

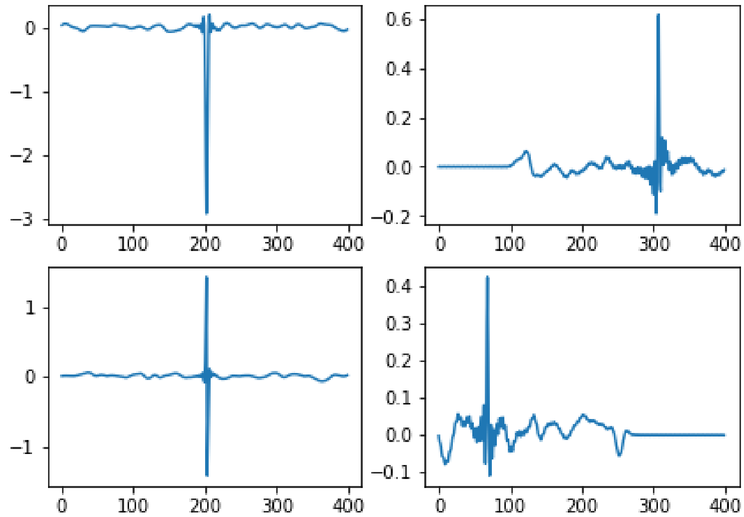


Figure 2: Four signals of the dataset

Figure 2 shows a sample of four signals from our dataset, which is composed of more than 140000 functions. It's clear that some plots seem to be clearly different from the rest. The goal of our project is to apply unsupervised clustering algorithms to classify each signal as normal or as anomalous.
Our dataset contains for each point:

- (x, y, z): spatial coordinates of the location of each point;
- (UAC1, UAC2) universal atrial components: used for the visualization of the system;
- 400 values of the electrical potential (400 time steps): value of the electrical potential in one point of the heart as a function of time;
- IIR value: Infinite Impulse Response;

Remark 1. *The impulse is propagating like a wave, consequently close points should have similar behavior. If a signal is different from the ones in its neighborhood, probably it can be classified as an anomaly.*

Remark 2. *The IIR value can be used to check the goodness of the classification: indeed, if a signal has IIR value greater than 1.22 it might be an anomaly.*

1.1. Exploratory Data Analysis

At first, we wanted to understand better what is an anomalous signal and its meaning, therefore some statistical studies were done, also we relied on Remark 2 to classify it.
Given $\vec{s}$ the signal for each fixed point and $\vec{s_d} = \{s_d(i) = s(i+1) - s(i) \quad \forall i = 0, \dots, N\}$ the derivative of $\vec{s}$, we calculated:

- max amplitude of the signal: $max(\vec{s})$;
- min amplitude of the signal: $min(\vec{s})$;
- variance of the signal: $Var(\vec{s}) = E[(\vec{s} - \mu)^2]$;
- max amplitude of the derivative of the signal: $max(\vec{s_d})$;
- min amplitude of the derivative of the signal: $min(\vec{s_d})$;
- position of the max amplitude of the signal: $\arg\max_i s(\vec{i})$;
- position of the min amplitude of the signal: $\arg\min_i s(\vec{i})$;
- kurtosis[1]: $E[(\frac{X-\mu}{\sigma})^4]$;
- skewness[2]: $E[(\frac{X-\mu}{\sigma})^3]$.

[1] A statistical measure that defines how heavily the tails of a distribution differ from the tails of a normal distribution, with μ its mean and σ its standard deviation

[2] A statistical measure of the asymmetry of the probability distribution of a real-valued random variable about its mean, with with μ its mean and σ its standard deviation

Plotting the values obtained in relation to the IIR value, the following plots are extracted:

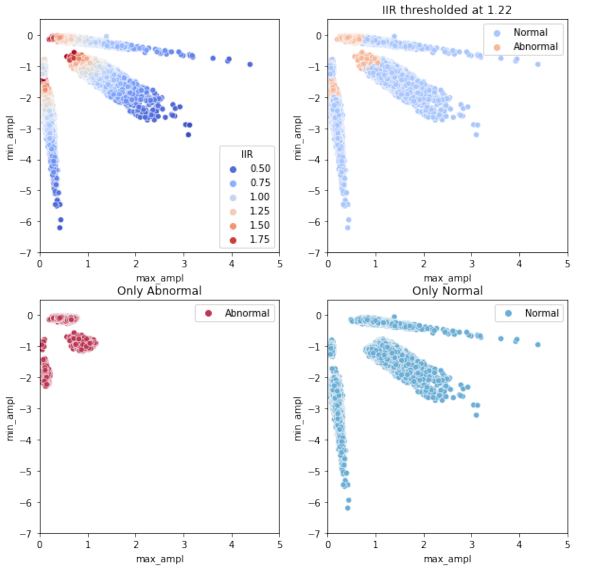


Figure 3: Relation between maximum amplitude and minimum amplitude

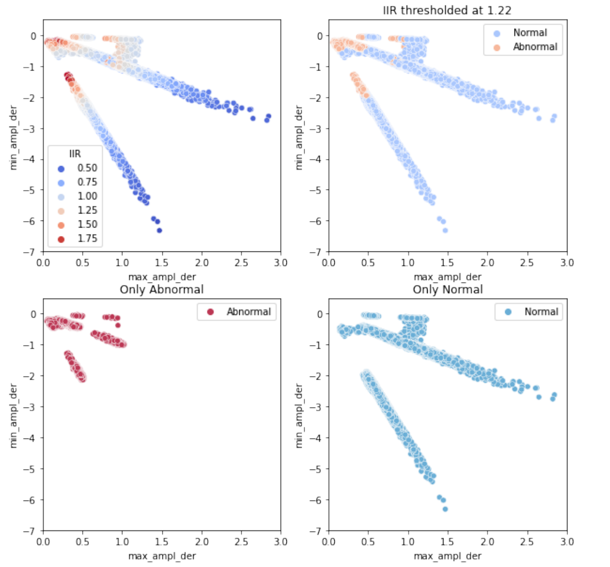


Figure 4: Relation between maximum derivative amplitude and minimum derivative amplitude

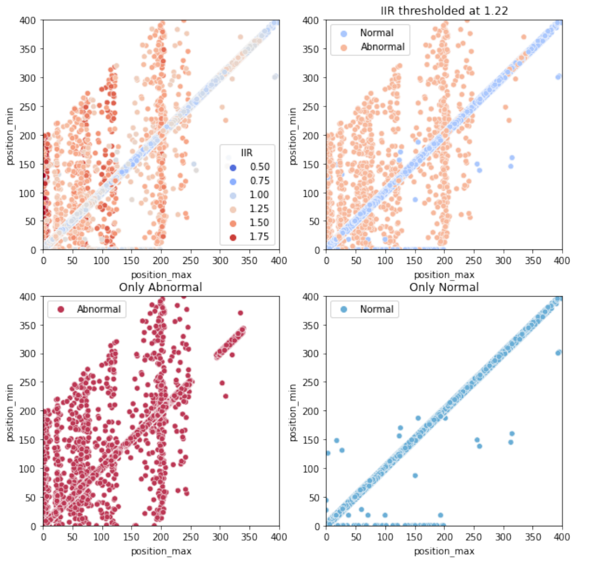


Figure 5: Relation between the position of maximum amplitude and the position of minimum amplitude

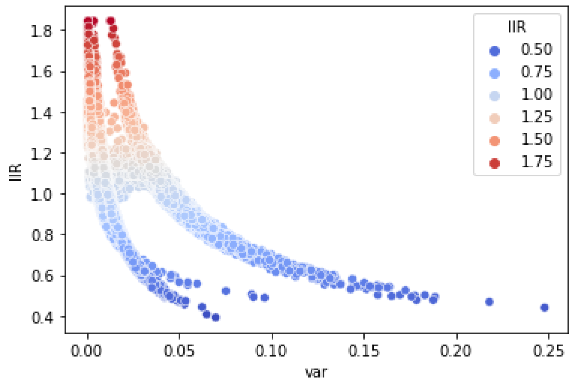


Figure 6: Relation between variance and IIR

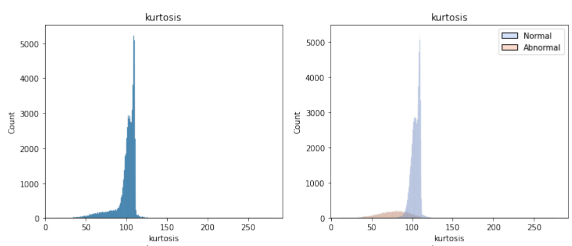


Figure 7: Distribution of kurtosis

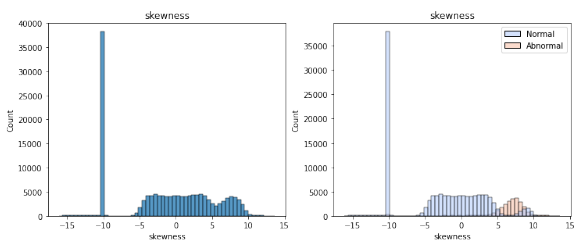


Figure 8: Distribution of skewness

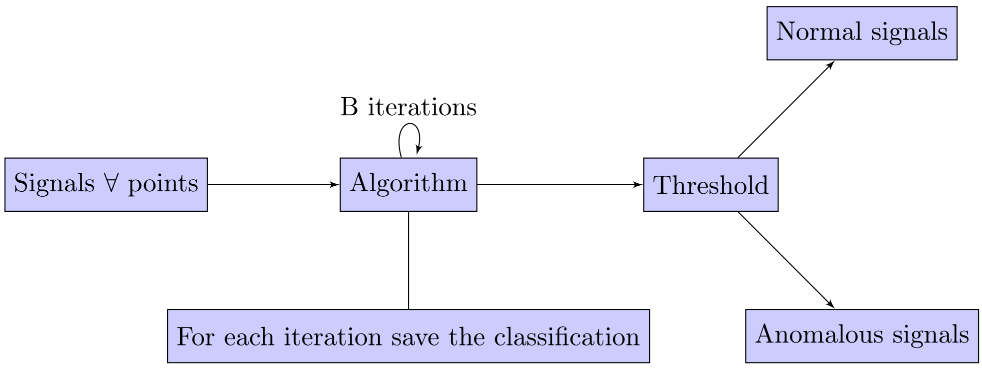
In general, we can observe that[3]:

- Anomalous signals have both their maximum amplitude and minimum amplitude small (Figure 3). The same holds for the derivative's amplitude (Figure 4);
- Normal signals seem to have peaks very close to each other, indeed they are distributed on a linear line (Figure 5);
- Anomalies have a small variance of the signal (Figure 6); peaks with smaller values around them, instead the abnormal ones tend to have lower kurtosis and so lower peaks (according to small variance) (Figure 7);
- The skewness suggests that abnormal signals always have a mean amplitude value larger than their median amplitude value, while it's not true for normal signals (Figure 8).

From these plots, we are able to identify the features most related to the anomalous signals, indeed they seem to be closer to flat signals with small amplitude peaks and small variance.

2. Methods

Our framework is the following:



Remark 3. *The threshold indicates the frequency of times in which (out of the B iterations) a point needs to be classified as an anomaly by the algorithm to confidently say that that point is an anomaly. For example if the threshold is 0.5 and $B = 100$, it means that a point is eventually classified as an anomaly if and only if it has been flagged as such in 51 iterations.*

Moreover, as Remark 1 suggests, we have to consider a spatial correlation between the points, for this reason for each iteration we divide the entire dataset using the Voronoi tessellations.

Definition 1 (Voronoi tessellation)**.** *Given a set $P = \{p_1, \dots, p_n\}$ of sites (called nuclei), a Voronoi tessellation is a subdivision of the points in a space into n groups $\{S_i\}_{i=1}^n$, one for each site in P, such that $\{q \in S_i \iff d(q, p_i) < d(q, p_j) \quad \forall i \neq j\}$.*

An example of a Voronoi tessellation is shown in the following image:

[3] by "normal" and "abnormal" we refer to signals having an IIR less or greater than 1.22, respectively.

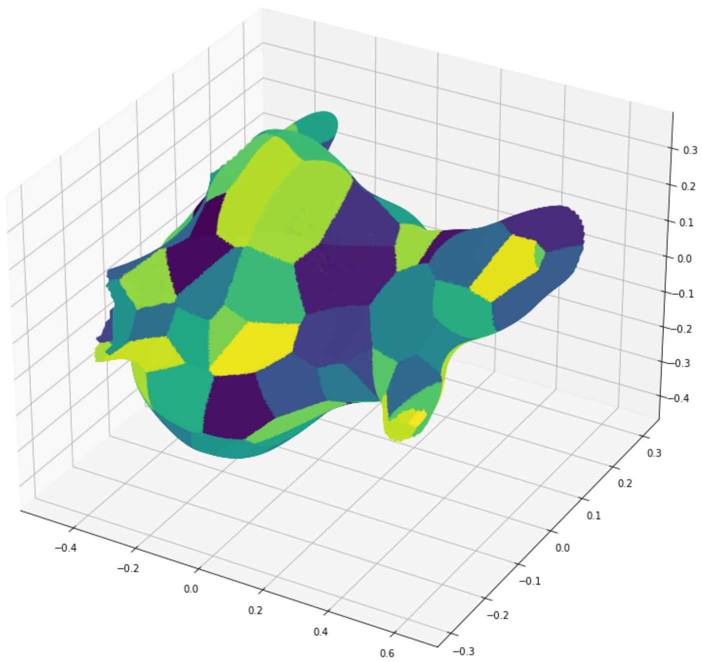


Figure 9: Random Voronoi tesselation for n = 100

An important consequence is the introduction of the hyperparameter **n** (number of tessellations and nuclei).

Furthermore, the signals receive the impulse in different time values, as result, the peak is at different times, this could negatively influence the performance because it could concentrate the method on the difference of the translation, instead of the differences between anomalies and normals. So the signals could need to be pre-processed to align the peak, it will be shifted to the middle of the time interval and padding with zeros wherever needed. The following plots show the results after the shift and the padding.

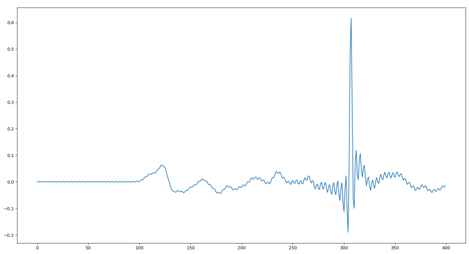


Figure 10: Original signal with peak at the end

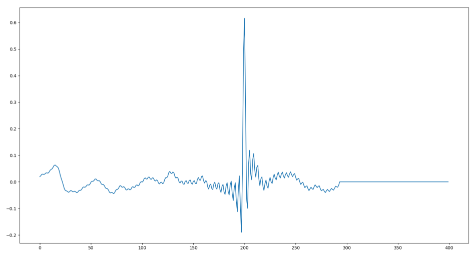


Figure 11: Translated signal with peak at the end

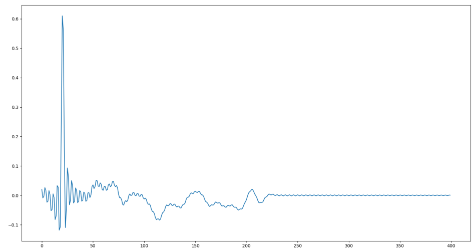


Figure 12: Original signal with peak at beginning

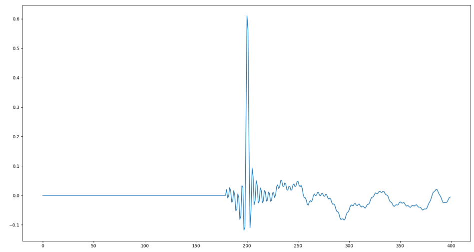


Figure 13: Translated signal with peak at beginning

These will be called **aligned signals**, to differentiate them to the original ones.

Finally, we developed three algorithms: **Voronoi - FPCA algorithm** (Section 2.1), **Voronoi - dimensionality reduction algorithm** (Section 2.2), **Voronoi - dictionary learning algorithm** (Section 2.3) and **Voronoi - weighted H^1 distance** (Section 2.4).

2.1. Voronoi - FPCA algorithm

The Voronoi algorithm is an unsupervised spatial clustering algorithm that uses Functional Principal Components Analysis to reduce the dimensionality: the infinite dimensional space which the functions belong to is condensed into the $\mathbb{R}^p$ vector space.

Definition 2 (Functional principal component analysis)**.** *Functional principal component analysis (FPCA) is a statistical method for investigating the dominant modes of variation of functional data. Using this method, a random function is represented in p eigenbasis, which is an orthonormal basis of the Hilbert space L^2 that consists of the eigenfunctions of the autocovariance operator.*

Then, it uses K-means to cluster the p-dimensional vectors. In our case **K** (number of clusters) is equal to 2: class 0 contains the normal signals, class 1 the anomalies. Also, the dataset used is the aligned signals one.
The method is composed of two steps: a boostrap step and an aggregation step. The first one is used to cluster each point and the second is used to gather the results of the bootstap step.

Algorithm 1 Bootstrap step

Initialize n, p, B.
for $b = 1, \cdots, B$ **do**

Randomly generate a set of n nuclei $\Phi_n^b = \{\boldsymbol{Z}_1^b, ..., \boldsymbol{Z}_n^b\}$ using the sites in S_0, such that $\boldsymbol{Z}_i^b \overset{i.i.d.}{\sim} \mathcal{U}(S_0)$

Obtain a random Voronoi tessellation of S_0, called $\{V(\boldsymbol{Z}_i^b|\Phi_n^b)\}_{i=1}^n$, by assigning each site $\boldsymbol{x} \in S_0$ to the nearest nucleus $\boldsymbol{Z}_i^n$, according to the Euclidean distance

for $i = 1, ..., n$ **do**

For each point $\boldsymbol{x}$ in the i-th patch compute its weight $w_{\boldsymbol{x}}^i$ as the evaluation of the probability density functon of a gaussian distribution evaluated at $\boldsymbol{x}$ with mean $\boldsymbol{Z}_i^b$ and variance Σ: $w_{\boldsymbol{x}}^i = \mathcal{N}(\boldsymbol{x}|\boldsymbol{Z}_i^b, \Sigma) \quad \forall i = 1, \ldots, n$

Compute g_i^b (local representative) by computing the weight average: $g_i^b = \frac{\sum_{j \in V_i} w_{\boldsymbol{x_j}}^i \vec{s}_j}{\sum_{j \in V_i} w_{\boldsymbol{x_j}}^i}$, where V_i is short for $V(\boldsymbol{Z}_i^b|\Phi_n^b)$, i.e. the set of points in the i-th tessellation.

end for

Perform dimensionality reduction of $\{g_1^b, ..., g_n^b\}$ by projecting these functions on $span\{p - \boldsymbol{FPCA}\}$, thus generating the p-dimensional spaces vectors $\{\boldsymbol{g}_1^b, ..., \boldsymbol{g}_n^b\}$

$\{\boldsymbol{g}_1^b, ..., \boldsymbol{g}_n^b\}$ are then clustered in K groups according to a suitable unsupervised method (K-means)

Assign the anomalous label for the group with fewer point

Update the classification matrix for each tessellation

end for

Remark: All the points of a same tessellation have the same label.
Remark: The covariance matrix Σ used when computing the weights of the representative is $\sigma^2 \mathbb{1}_3$, where $\sigma^2 = d_{max}/d_{min}$, d_{max} and d_{min} are the maximum and minimum distance between two nuclei of the tessellation and $\mathbb{1}_3$ is the identity matrix in $\mathbb{R}^3$.

In the end, we perform the second step: the aggregation one, where all the results of the precedent step are post-processed.

Algorithm 2 Aggregation step

for $\boldsymbol{x} \in S_0$ **do**

Compute the frequencies of assignment of the site to each of the K clusters along iterations, i.e. $\pi_{\boldsymbol{x}}^k = \#\{b \in \{1, ..., B\} \ : \ \boldsymbol{x} \in C_k^b\}/B \quad \forall k = 1, ..., K$

Compute spatial entropy $\eta_{\boldsymbol{x}}^k$ for each site $\boldsymbol{x} \in S_0$

end for

Remark: $\eta_{\boldsymbol{x}}^k$ is used for the evaluation of the quality of the final classification.
Remark: C_k^b is the set of $\boldsymbol{x} \in S_0$ whose label is equal to k.

2.2. Voronoi - dimensionality reduction algorithm

After the analysis presented in Section 1.1, we decided to manually perform the dimensionality reduction using specifically features that might be able to help to distinguish between normal and abnormal signals. Note that the signals, with this algorithm, must be the original ones to keep all the characteristics unvaried.

Algorithm 3 Manual dimensionality reduction algorithm

Create specific datasets with: ▷ Most influential features for anomalous signals
- the difference of amplitude in modulus of the signals: ΔA;
- the difference of amplitude derivative in modulus of the signals: ΔA_d;
- the difference of position of the max value and the min value in modulus: ΔP;
- the kurtosis: k.

Initialize n, B, $\epsilon = 0.1$
for $b = 1, \dots, B$ **do**
  Randomly generate a set of n nuclei $\Phi_n^b = \{\boldsymbol{Z}_1^b, ..., \boldsymbol{Z}_n^b\}$ using the sites in S_0, such that $\boldsymbol{Z}_i^b \overset{i.i.d.}{\sim} \mathcal{U}(S_0)$
  Obtain a random Voronoi tessellation of S_0, called $\{V(\boldsymbol{Z}_i^b|\Phi_n^b)\}_{i=1}^n$, by assigning each site $\boldsymbol{x} \in S_0$ to the nearest nucleus $\boldsymbol{Z}_i^n$, according to the Euclidean distance
  for $i = 1, \dots, n$ **do**
    Compute the mean $(\vec{\mu_{\Delta A}}, \vec{\mu_{\Delta A_d}}, \vec{\mu_{\Delta_p}}, \vec{\mu_k})$ and the covariance matrix $(S_{\Delta A}, S_{\Delta A_d}, S_{\Delta_p}, S_k)$ for the specific region
    Compute the Mahalanobis distance[1] of each score to the respective mean value and covariance matrix
    Define the threshold as the q_j=$(1-\epsilon)$ - quantile of the distance for $j = \Delta A, \Delta A_d, \Delta P, k$
    for $x_0 \in \{\text{tessellation } i\}$ **do**
      if $(\Delta A(x_0) > q_{\Delta A}$ **or** $\Delta A_d(x_0) > q_{\Delta A_d}$ **or** $\Delta P(x_0) > q_{\Delta P}$ **or** $k(x_0) > q_k)$ **then**
        x_0 is an anomaly ▷ Update the classification matrix
      end if
    end for
  end for
end for

Definition 1: Given a probability Q on $\mathcal{R}^N$ with mean $\vec{\mu}$ and positive-definite variance matrix S, the Mahalanobis distance of a point $\vec{x}$ from Q is: $d(\vec{x}, Q) = \sqrt{(\vec{x} - \vec{\mu})^T S^{-1} (\vec{x} - \vec{\mu})}$.
Remark: In each tessellation the points could have different labels.

To conclude: the algorithm classifies as anomalies the signals that frequently happen to have a large Mahalanobis distance from the average of the group.

2.3. Voronoi - dictionary learning algorithm

Another possible way to avoid computing an unsupervised dimensionality reduction is the dictionary learning method. This algorithm is used in the Image Processing field, but since images are usually treated as a composition of different rectangular patches and the patches are usually unrolled and treated as signals, it can be applied for our task.

Definition 3 (Dictionary learning)**.** *Dictionary learning is a representation learning method which aims at finding a sparse representation of the input data (also known as sparse coding) in the form of a linear combination of basic elements as well as those basic elements themselves. In particular, given the input dataset $X = [x_1, \dots, x_N]$, we wish to find a dictionary $\mathcal{D} \in \mathbf{R}^{dxn}$ and a representation $R = [r_1, \dots, r_N]$ such that $||X - DR||^2$ is minimized and the representations are sparse enough.*

The idea is to divide the dataset in Voronoi tessellations and train a dictionary on them. The dictionary will learn to represent accurately and sparsely the most frequent shapes that are assumed to be the Normal ones, while the more rare, beeing poorly reconstructed, will be classified as Anomalies.
Aligned signals will be used, since the reconstruction error $||X - DR||^2$ is very sensible to the position of the peak.

Algorithm 4 Dictionary learning method

Initialize n, $\epsilon = 0.1$, B and hyperparameters for dictionary learning algorithm
for $b = 1, \dots, B$ **do**
  Randomly generate a set of n nuclei $\Phi_n^b = \{\boldsymbol{Z}_1^b, ..., \boldsymbol{Z}_n^b\}$ using the sites in S_0, such that $\boldsymbol{Z}_i^b \overset{i.i.d.}{\sim} \mathcal{U}(S_0)$

  Obtain a random Voronoi tessellation of S_0, called $\{V(\boldsymbol{Z}_i^b|\Phi_n^b)\}_{i=1}^n$, by assigning each site $\boldsymbol{x} \in S_0$ to the nearest nucleus $\boldsymbol{Z}_i^n$, according to the Euclidean distance

  for $i = 1, \dots, n$ **do**
    Create a dictionary $\mathcal{D}$ using K number of components and 1 as sparsity of the representation we want to achieve

    Define a subgroup of the current tessellation with signals $\vec{s_a}$ and fit the dictionary on it

    Define another subgroup with signals $\vec{s_b}$ and compute the representation $\tilde{s_b}$ based on the dictionary learned

    Compute the L^1-norm of the representation ($norm_{L^1}^b = ||\tilde{s_2}||_{L^1}$), the reconstruction error ($err_{rec}^b = ||\tilde{s_b} \cdot \mathcal{D} - \vec{s_b}||_{L^2}$) and their means ($\mu_{L^1}^b = E[norm_{L^1}^b]$ and $\mu_{rec}^b = E[err_{rec}^b]$) and covariances ($S_{L^1}^b = Cov[err_{L^1}^b]$ and $S_{rec}^b = Cov[err_{rec}^b]$)

    Define the latest subgroup with signals $\vec{s_c}$, compute the new representation $\tilde{s_c} = \vec{s_c} \cdot \mathcal{D}$, and, as before, the L^1-norm ($norm_{L^1}^c$) and the reconstruction error (err_{rec}^c)

    Compute the Mahalanobis distance[1] using the calculated previously $[\mu_{L^1}^b, \mu_{rec}^b, S_{L^1}^b, S_{rec}^b]$ and the recent $norm_{L^1}^c, err_{rec}^c$

    Set the $q = 1 - \epsilon$ - quantile of the distance as a threshold

    Compute the representation $\tilde{s}$ for all signals $\vec{s}$ of the current tessellation

    Compute the final L^1-norm ($norm_{L^1}$) and the reconstruction error (err_{rec})

    Compute the Mahalanobis distance[1] using the calculated previously $[\mu_{L^1}^b, \mu_{rec}^b, S_{L^1}^b, S_{rec}^b]$ and the recent $norm_{L^1}, err_{rec}$

    for $x_0 \in \{\text{tessellation } i\}$ **do**
      if distance(x_0) > q **then**
        x_0 is an anomaly ▷ Update the classification matrix
      end if
    end for
  end for
end for

Definition 1: Given a probability Q on $\mathcal{R}^N$ with mean $\vec{\mu}$ and positive-definite variance matrix S, the Mahalanobis distance of a point $\vec{x}$ from Q is: $d(\vec{x}, Q) = \sqrt{(\vec{x} - \vec{\mu})^T S^{-1} (\vec{x} - \vec{\mu})}$.
Remark: In each tessellation the points could have different labels.
Remark: The idea is that the signals classified as anomalous will be the ones that badly represented by the respective dictionary. We try to assess the quality of the reconstruction by the L^1 - norm of the representation and the reconstruction error.

A visualization of the subgroups for each tessellation could be the following graph, supposing these are all the signals in a specific tessellation i:

$\vec{s^1}$	$\vec{s^2}$	$\vec{s^3}$	$\vec{s^4}$	$\vec{s^5}$	$\vec{s^6}$	$\vec{s^7}$	$\vec{s^8}$	$\vec{s^9}$	$\vec{s^{10}}$	$\vec{s^{11}}$	$\vec{s^{12}}$	$\vec{s^{13}}$	$\vec{s^{14}}$	$\vec{s^{15}}$	$\cdots$	$\vec{s^N}$

where:
- $\vec{s_a}$ is the set of blue cells $\{\vec{s^1}, \vec{s^7}, \vec{s^{13}}, \vec{s^{14}}, \dots\}$
- $\vec{s_b}$ is the set of red cells $\{\vec{s^3}, \vec{s^6}, \vec{s^9}, \vec{s^{12}}, \dots\}$
- $\vec{s_c}$ is the set of green cells $\{\vec{s^2}, \vec{s^5}, \vec{s^{10}}, \vec{s^{11}}, \dots\}$
- $\vec{s}$ is the set of all cells $\{\vec{s^1}, \dots \vec{s^N}\}$

These subgroups are formed randomly.

2.4. Voronoi - weighted H^1 distance

The ultimate version of the Voronoi algorithm, so far the best one in terms of results, has been obtained simplifying the Voronoi - FPCA algorithm of Section 2.1. Similarly to the original one, it's still based on the Voronoi tessellation and it uses aligned signals, but:

- The randomly sampled point becomes the representative of the tessellation (similarly to algorithm in Section 2.1) with covariance matrix $\Sigma \approx 0$ for the gaussian weights;
- The dimensionality reduction is not performed anymore, indeed the algorithm treats the set of representatives as a graph model where the nodes are the signals and the edges are the weighted H^1 distances[4] between the nodes, which are treated as vectors in $\mathbb{R}^{400}$ (since the signals are composed of 400 time stamps). Finally the algorithm computes a measure of the **centrality** of each node in the graph and classifies as anomalous the farthest nodes (it avoids the clustering with Kmeans).

Algorithm 5 Bootstrap step

Initialize n, ϵ, B
for $b = 1, \cdots, B$ **do**

Randomly generate a set of n nuclei $\Phi_n^b = \{\boldsymbol{Z}_1^b, ..., \boldsymbol{Z}_n^b\}$ using the sites in S_0, such that $\boldsymbol{Z}_i^b \overset{i.i.d.}{\sim} \mathcal{U}(S_0)$

Obtain a random Voronoi tessellation of S_0, called $\{V(\boldsymbol{Z}_i^b|\Phi_n^b)\}_{i=1}^n$, by assigning each site $\boldsymbol{x} \in S_0$ to the nearest nucleus $\boldsymbol{Z}_i^n$, according to the Euclidean distance

Compute the weighted H^1 Distance[1] $D(i,j)$ between the representatives Φ_i^b and Φ_j^b $\forall(i,j)$

Compute the centrality measure of each representative signal as $C(i) = \frac{1}{\sum_j D(i,j)}$

Compute $\tau := \epsilon$ - *quantile of the distribution of the centrality measures* and classify as anomalous the representatives Φ_i^b s.t. $C(i) \leq \tau$ and as normal the others

Update the classification matrix for each tessellation assigning $\forall$ signal $\mathbf{x}$ the same label as the nearest representative

end for

Remark 1: The weighted $H1$ distance is to be intended as distance between vectors in $\mathbb{R}^{400}$ and not between their 3D coordinates.
Remark: All the points of the a same tessellation get the same label.

---

[4] i.e. $\text{dist}(f_1, f_2) = ||f_1 - f_2||_{L2}^2 + \theta||\partial_t f_1 - \partial_t f_2||_{L2}^2$, where θ is used to tune the relevance of the first derivative of the signals.

3. Numerical results

In this section, the numerical results of the methods previously explained will be presented using a scatterplot, where the x-coordinate is the index of the signal and the y-coordinate is the frequency of times the model classified the observation as an anomaly. Finally, the points are colored based on the clustering obtained with the thresholding of the IIR index, such that the red ones have the IIR values greater than 1.22, instead the blue ones lower. A key point is that the more the points with different colors are separated (the points red should be at the top, instead the blue ones at the bottom), the better the performance of the model is.

Voronoi - FPCA algorithm:

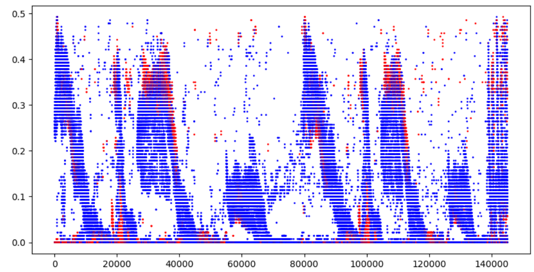


Figure 14: Attempt performed with $n = 200$

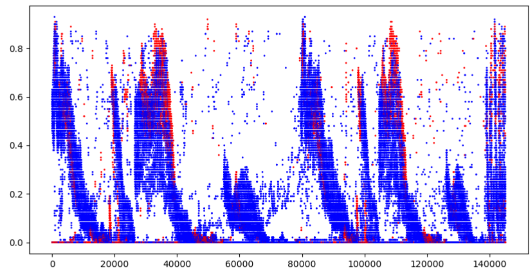


Figure 15: Attempt performed with $n = 500$

As these plots clearly show, the clustering performed by the algorithm is very far away from the correct IIR thresholding distribution, but these plots display the robustness of this algorithm with respect to the choice of the number of nuclei (n).

Voronoi - dimensionality reduction algorithm:

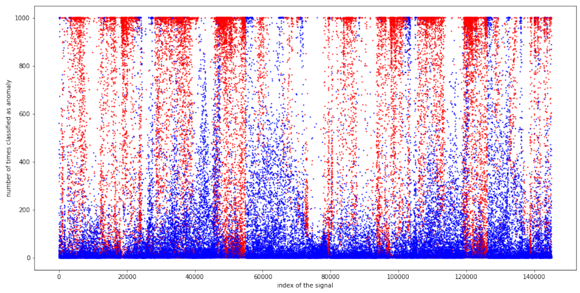


Figure 16: Attempt performed with $n = 5$

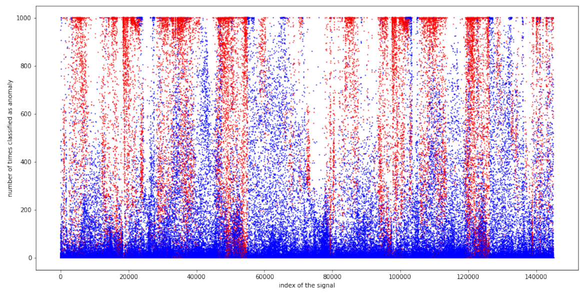


Figure 17: Attempt performed with $n = 10$

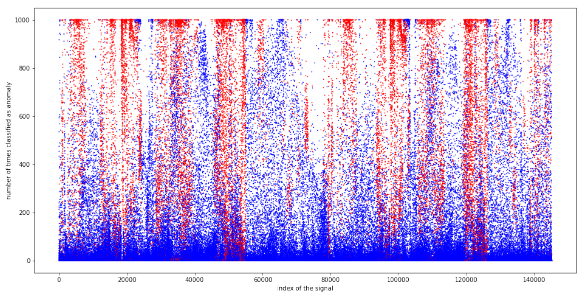


Figure 18: Attempt performed with $n = 15$

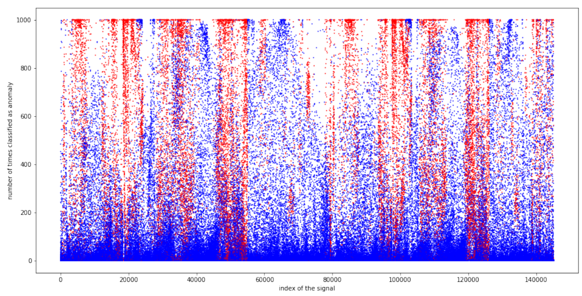


Figure 19: Attempt performed with $n = 20$

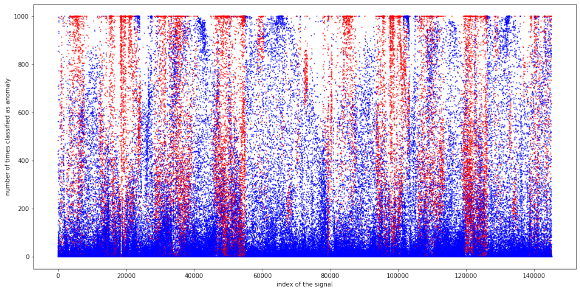


Figure 20: Attempt performed with $n = 25$

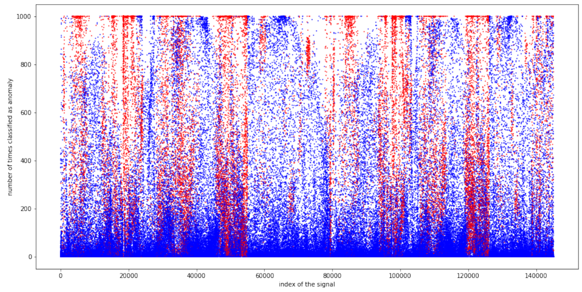


Figure 21: Attempt performed with $n = 30$

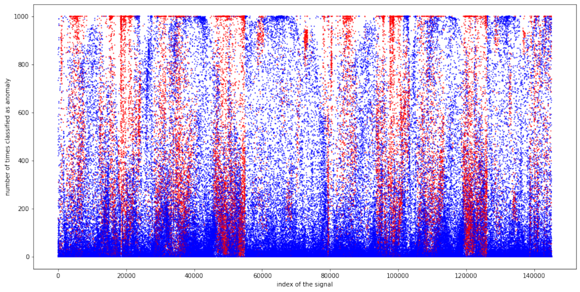


Figure 22: Attempt performed with $n = 40$

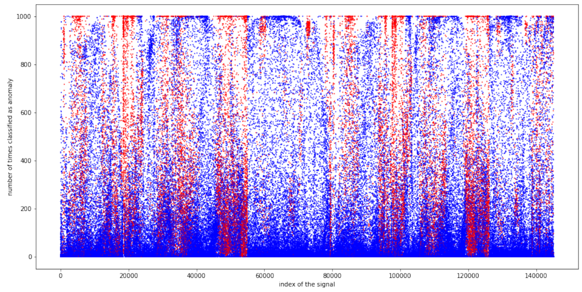


Figure 23: Attempt performed with $n = 50$

The results are satisfying for small values of n, indeed the uncertainty of the classification increases with this hyperparameter, as we can see from Figure 16 to Figure 23.

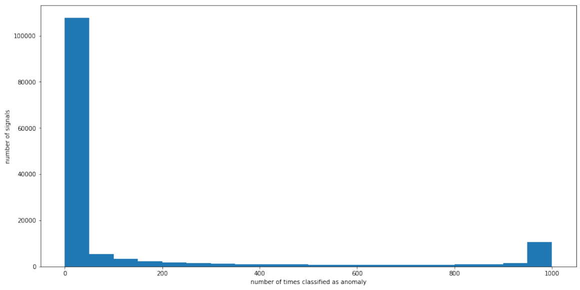


Figure 24: Histogram of assignments, $n = 5$

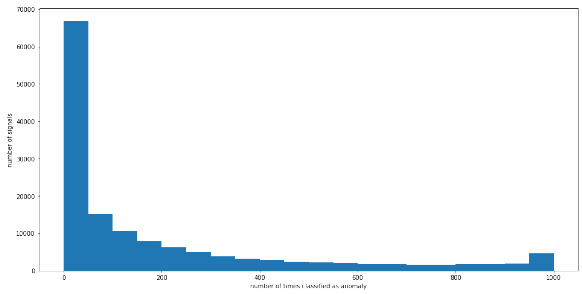


Figure 25: Histogram of assignments, $n = 50$

The ideal histogram should present only two peaks: one located near zero and the second one far away from it. If this were the case, then it would mean that the model was able to classify most of the points as normal, except for a relatively small subset, which would be characterized by containing points classified a reasonably high number of times as anomalies. The same consideration holds for the histograms that are shown in the following sections.
As Figures 24 and 25 show, if we pass from n equal to 5 to n equal to 50, the Voronoi model with dimensionality reduction becomes more unsure since the histogram is more spread.
Finally, the following plot provides an additional proof of the fact that when n increases, the model becomes more unsure, indeed the average normalized entropy grows as n grows, meaning that the clusters found are more "heterogeneous":

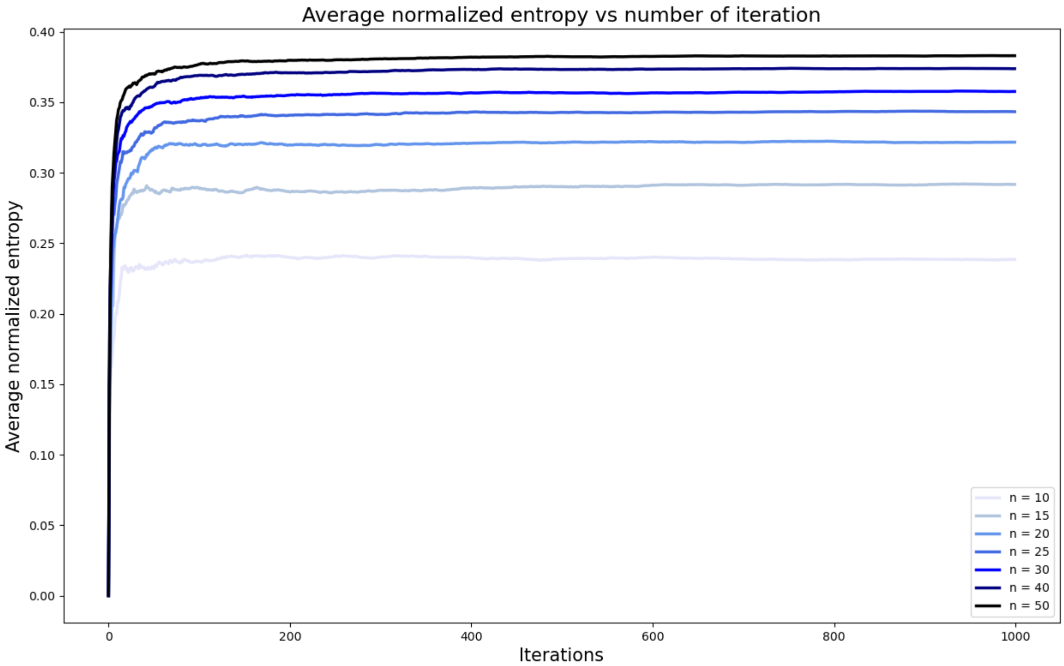


Figure 26: Average normalized entropy vs number of iterations

Voronoi - dictionary learning algorithm:

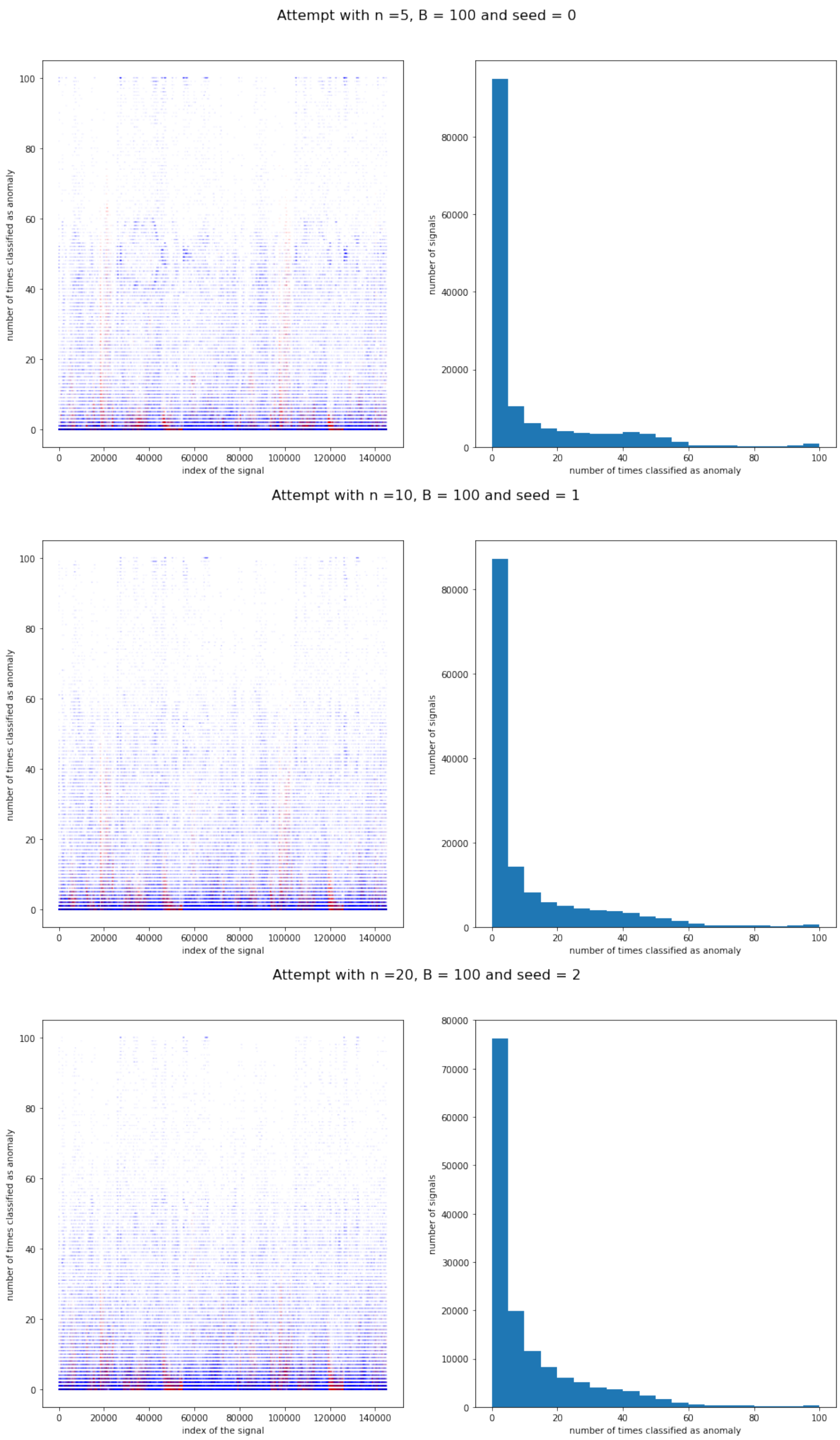


Figure 27

Unfortunately, the results are very poor because, as we can see, the algorithm is quite uncertain, indeed there are no signals which are classified almost all the time as anomalies and it is also inconsistent with respect to the IIR thresholding.

Voronoi - weighted H^1 distance:
The numerical results for this algorithm present an interesting dependence on the number of nuclei n and the threshold chosen to cluster the anomalies, which is used as explained in Section 2.4.

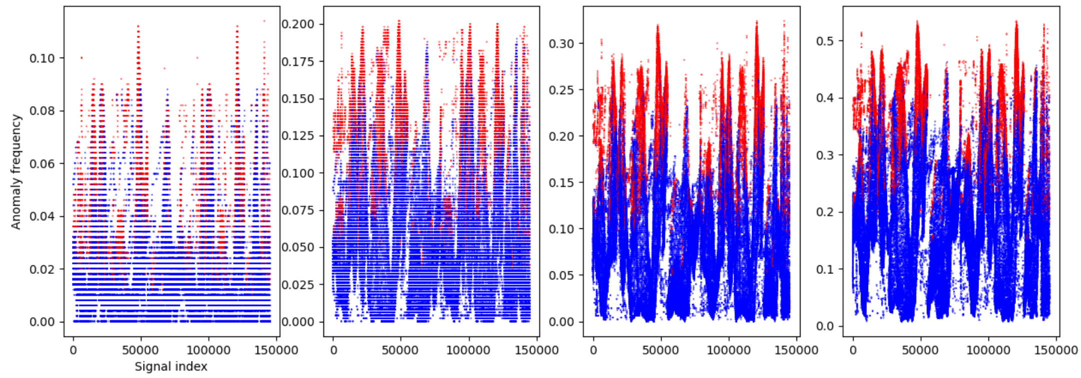


Figure 28: Attempt with $n = 50$ and threshold from left to right 0.01, 0.05, 0.1, 0.2

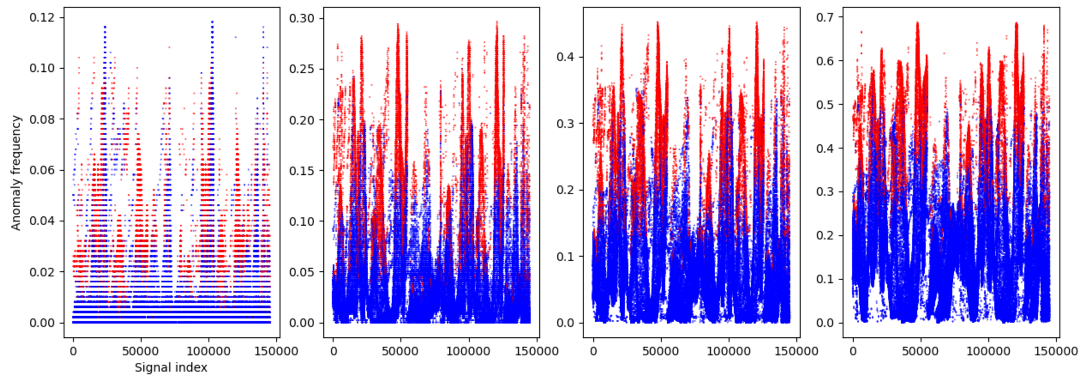


Figure 29: Attempt with $n = 100$ and threshold from left to right 0.01, 0.05, 0.1, 0.2

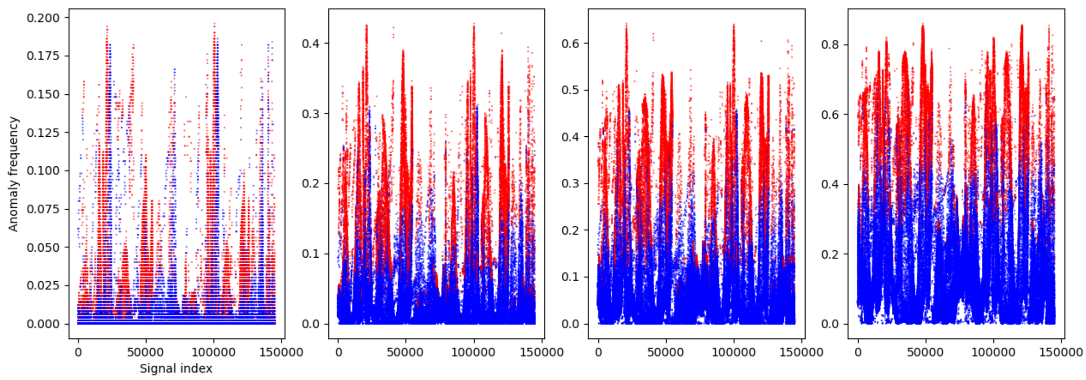


Figure 30: Attempt with $n = 200$ and threshold from left to right 0.01, 0.05, 0.1, 0.2

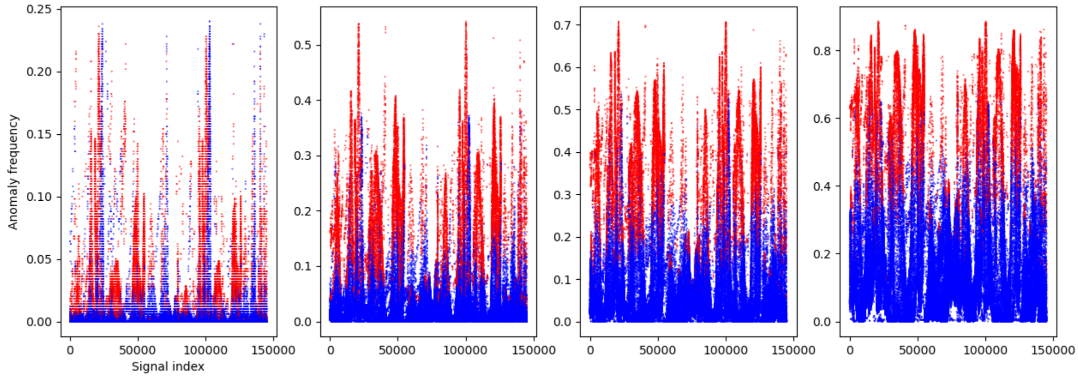


Figure 31: Attempt with $n = 300$ and threshold from left to right 0.01, 0.05, 0.1, 0.2

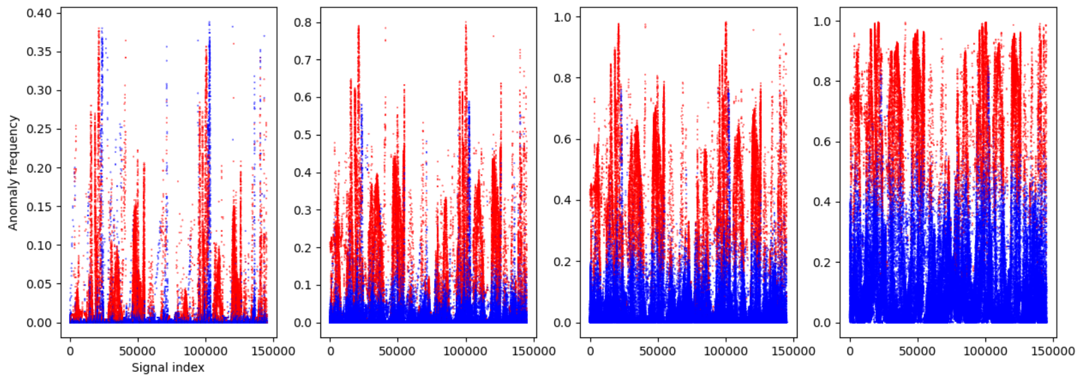


Figure 32: Attempt with $n = 1000$ and threshold from left to right 0.01, 0.05, 0.1, 0.2

It is clear that as the threshold parameter increases, not only the range of assignment to the anomaly cluster increases (for example, the range passes from $(0, 0.25)$ to $(0, 0.85)$ in Figure 31), but it also becomes necessary to increase the anomaly frequency limit to capture the red points. For example, in reference to Figure 32, when using a threshold equal to 0.01, if we define as anomalies all points which have been classified 10% of the time by our algorithm, we would capture almost only the red dots. Instead, if we use as threshold 0.2, using as limit 10% would be inappropriate, since we would classify as anomalies a huge number of blue points (using 50% might be a more suitable choice). This consideration holds especially when the number of nuclei is high, but it's less evident in the case of, for example, n equal to 50.

In conclusion, out of all our models, this seems the most coherent concerning the IIR index, even though it is not based on the IIR feature at all.

4. Bibliography

The paper we studied about the Voronoi tessellation coupled with FPCA is [2]. Moreover, the dictionary learning algorithm is found in [1].

5. Conclusions

Throughout this project, we not only confirmed the importance of the IIR index through some statistical analysis and data exploration (section 1.1), but we also generalized this analysis with the goal of taking into account the spatial correlation among the functions. In particular, to model the correlation, our algorithms are based on two different assumptions about the anomalies:

1. anomalies are grouped and close to each other

2. anomalies are isolated points, which should stand out because of the different shapes of their signals with respect to the nearby signals.

Moreover, we remark that even though the IIR index seems a perfect thresholding feature to cluster the signals when they are considered independent from each other, when dealing with spatial correlation this feature might not be as important.

According to the numerical results in Section 3, our best model is Voronoi with weighted H^1 distance, as explained in Subsection 2.4.

Voronoi - FPCA algorithm:

Advantages

- The algorithm is robust: it produces similar results at each run. This is due to the fact that all points belonging to one single element of the tessellation are clustered according to their representative.
- It especially takes into account how close the points are, thanks not only to the tessellation but also to the covariance matrix of the gaussian weights used to compute the patch representative.
- It works well if we assume that anomalous points are close to each other. This clearly also depends on the number of nuclei chosen.

Disadvantages

- The computational cost is high.
- It seems that the clusters found by this algorithm differ from the ones obtained with a naive IIR-based partition.
- The choice of K-means as the final step of this model might be a possible bottleneck. Possible alternative unsupervised models for classifying vectors may be taken into consideration.
- Because of the computational load, some modeling choices are difficult to tune: in particular, the covariance structure of the gaussian weights used to compute the representative for each patch should be chosen to guarantee to properly factor all the points in.

Voronoi - dimensionality reduction algorithm:

Advantages

- It works well comparing the results with the IIR index thresholding.
- It is robust with respect to the choice of the seed.
- It is fast since the dataset on which it works is already reduced.
- There are few hyperparameters to tune.

Disadvantages

- It works under the assumption of isolated anomalies and increasing n we enforce this assumption.
- Increasing n the results tend to be different from IIR thresholding as you can see from Figure 16 to 25. This is because the IIR value seems to be continuous on the LA surface, meaning that signals closed together have similar IIR (Figure 33).

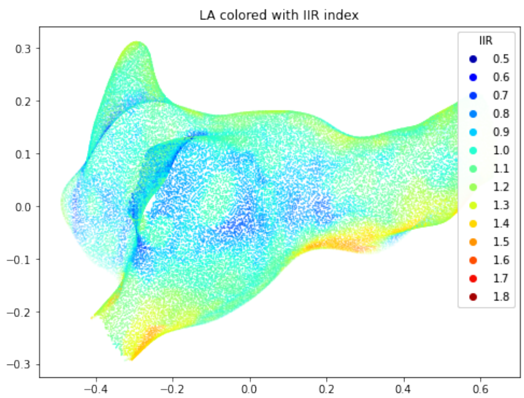


Figure 33: Distribution of the IIR on the LA surface

- The features we chose for the dimensionality reduction are correlated with the IIR index, but in reality we don't know if the IIR actually clusters the signals.

Voronoi - dictionary learning algorithm:

Advantages

- It is completely independent of the IIR index.
- The speed of the algorithm can increase if we reduce the *dimension of all the subgroups.* We do not recommend decreasing it too much since it may result in meaningless dictionaries learned.

Disadvantages

- The signals detected as anomaly are just the ones that are badly described by the dictionary, this means that it can work properly if the anomalies are isolated and rare in order to be always the less frequent group for each voronoi tessellation.
- The algorithm is slow.
- There are a lot of hyperparameters related to the dictionary learning process that are difficult to tune since they can vary in a large range and the algorithm is slow.

This algorithm ended up failing even if it is very valid in the image processing field, probably because the image patches (that are rectangular regions of the image), when unrolled, become "signals" very different from one another, and when a dictionary is trained, it learns the most recurrent patterns among the patches. On the contrary, our signals were very similar to each other (being impulses) and there were only few features able to discriminate anomalies from normal signals, so the algorithm ended up considering most of the signals as normal and assigning the anomalous label with lots of uncertainty as clearly shown in Figure 27.

Voronoi - weighted H^1 distance:

Advantages

- Entirely independent from the IIR.
- It has a good performance when compared to the naive IIR thresholding model.
- Computationally efficient.
- Robust with respect to perturbations in the parameter n.

Disadvantages

- It is difficult to interpret the results: while the other algorithms classify anomalies based on the statistics of the signals, this algorithm uses the shape of the signals themselves, hence there is no way of motivating an assignment based on, for example, the position of the peak, its intensity or the average value of the first derivate.

Using this method we can obtain the best confusion matrix so far, as we can see in the following figure, indeed the misclassified signals (prediction 1 and IIR < 1.22, prediction 0 and IIR > 1.22) are very small in comparison to the right classified (prediction 1 and IIR > 1.22, prediction 0 and IIR < 1.22):

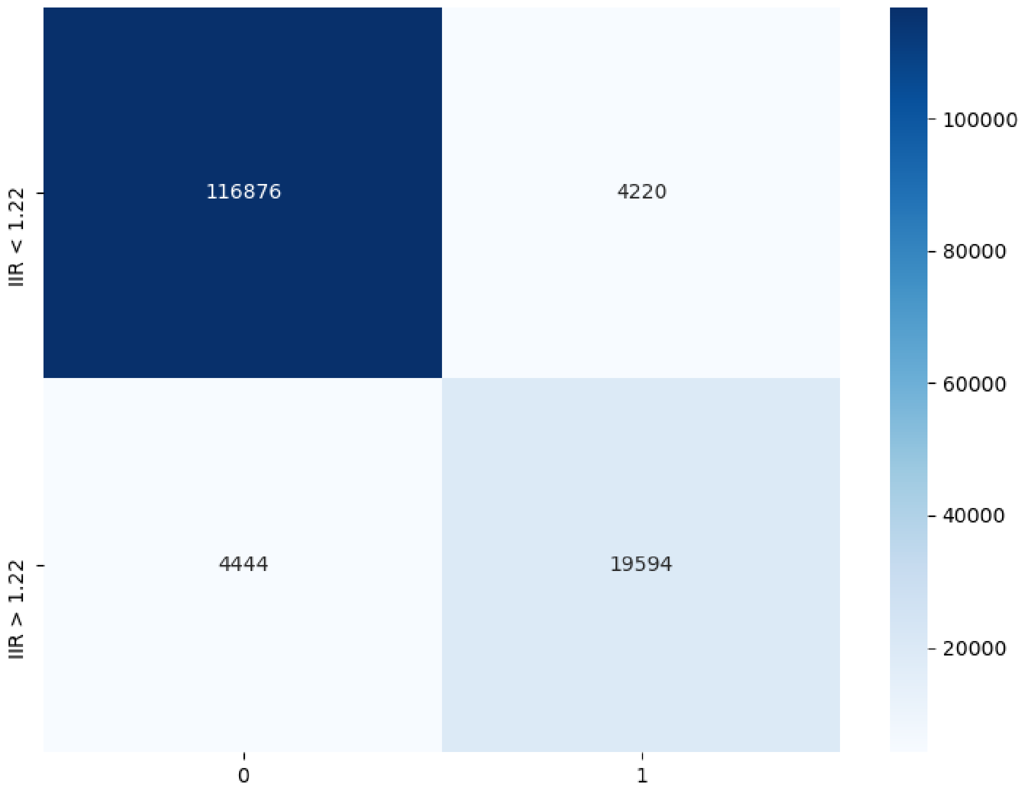


Figure 34: Confusion matrix using Voronoi with weighted H^1 distance with $n = 300$, threshold $\epsilon = 0.05$ and 0.05 as limiting anomaly detection frequency for clustering.

References

[1] Diego Carrera, Fabio Manganini, Giacomo Boracchi, and Ettore Lanzarone. Defect detection in sem images of nanofibrous materials. *IEEE Transactions on Industrial Informatics*, 13(2):551–561, 2017.

[2] Piercesare Secchi, Simone Vantini, and Valeria Vitelli. Bagging voronoi classifiers for clustering spatial functional data. *International Journal of Applied Earth Observation and Geoinformation*, 22:53–64, 2013. Spatial Statistics for Mapping the Environment.