Spectra

March 20, 2020

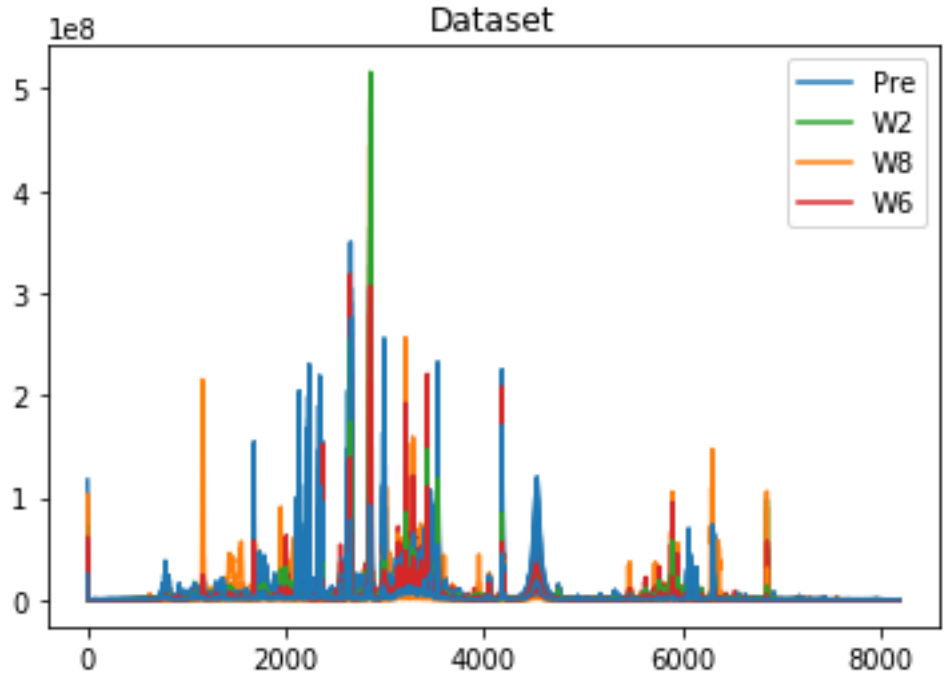
1 NMR spectra investigation

In this document presented an investigation of four classes of multi-component signal with chemical shift.

The first aim of this investigation is to see if unsupervised clusteristion can present convincing results. Second - if it is possible to set up a NN-based signal classifier.

1.1 1. Data overview

The disposable dataset has 59 samples belonging to 4 categories: 'Pre'(19 samples), 'W2'(14 samples), 'W8'(13 samples), 'W6'(13 samples).



Pic.1

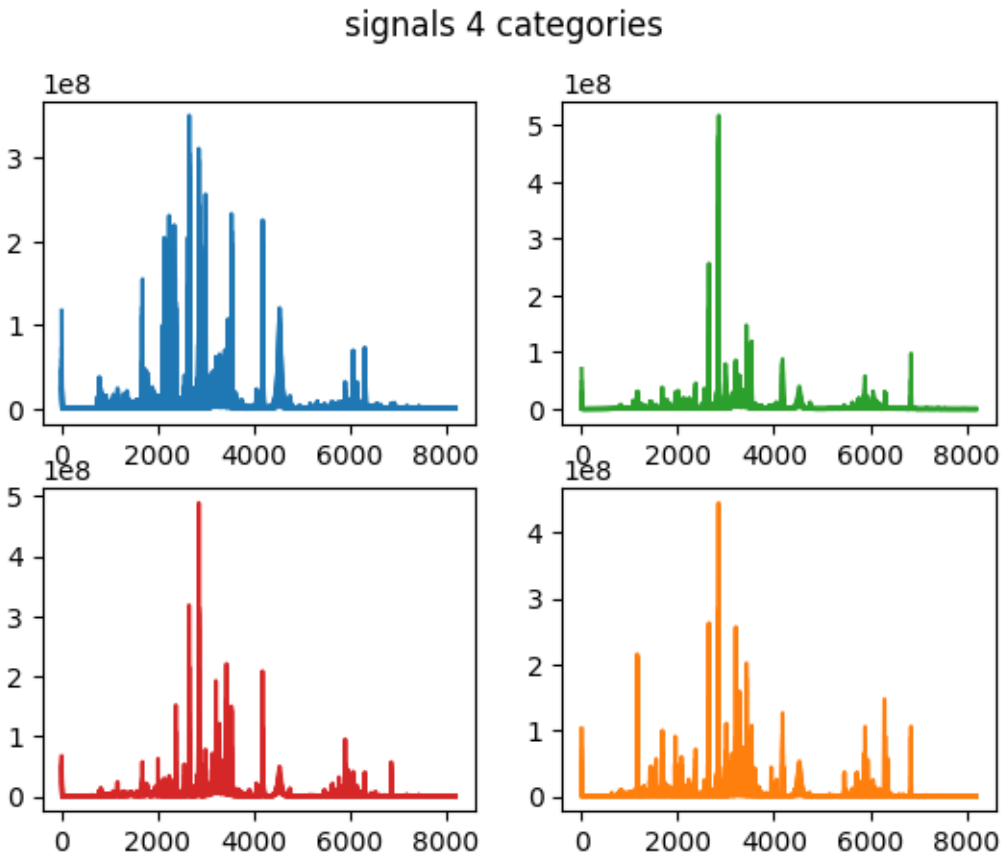
As seen on the Pic.1, there is no clear pattern for any of signal types. Pic.2 and Pic.3 show all the signal groups plotted separated and and a random example of each type of signal respectively.

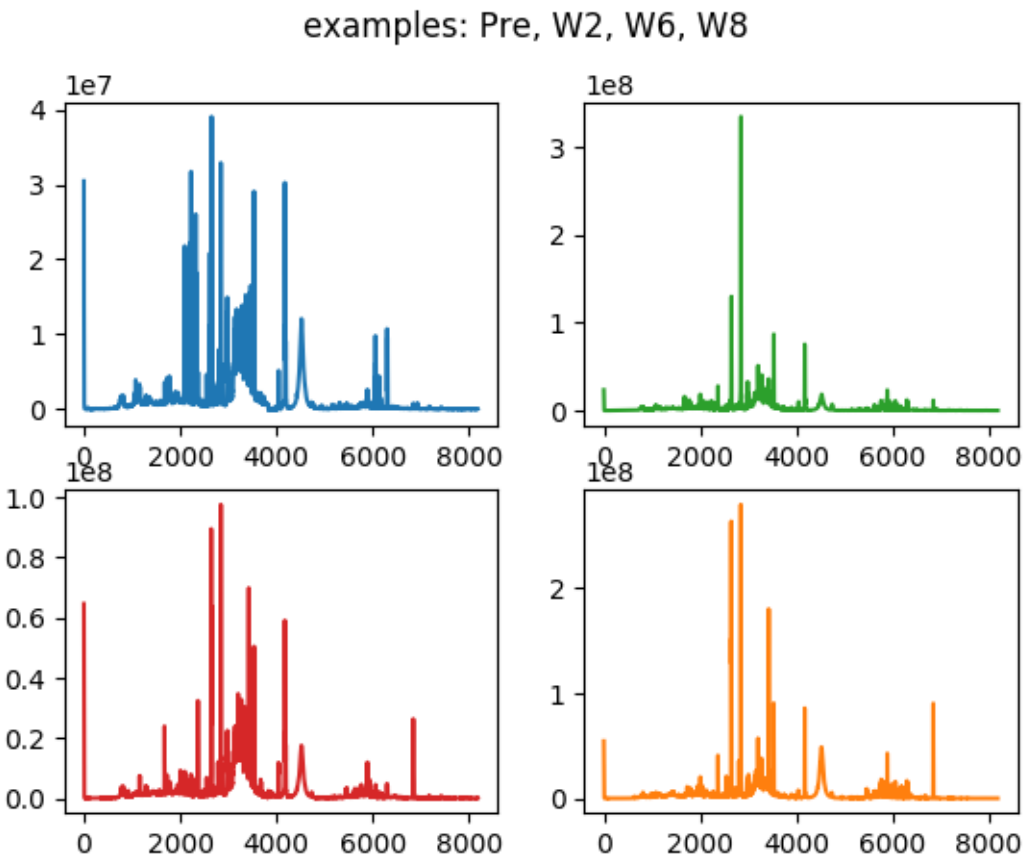
All signals

|

Signals plotted by class

- -





Pic.2

|

Pic.3

As seen, there is no obvious clear pattern which allows visually discern the signals and define which class they belonging to.

1.2 2. Clusterisation

1.2.1 2.1 PCA

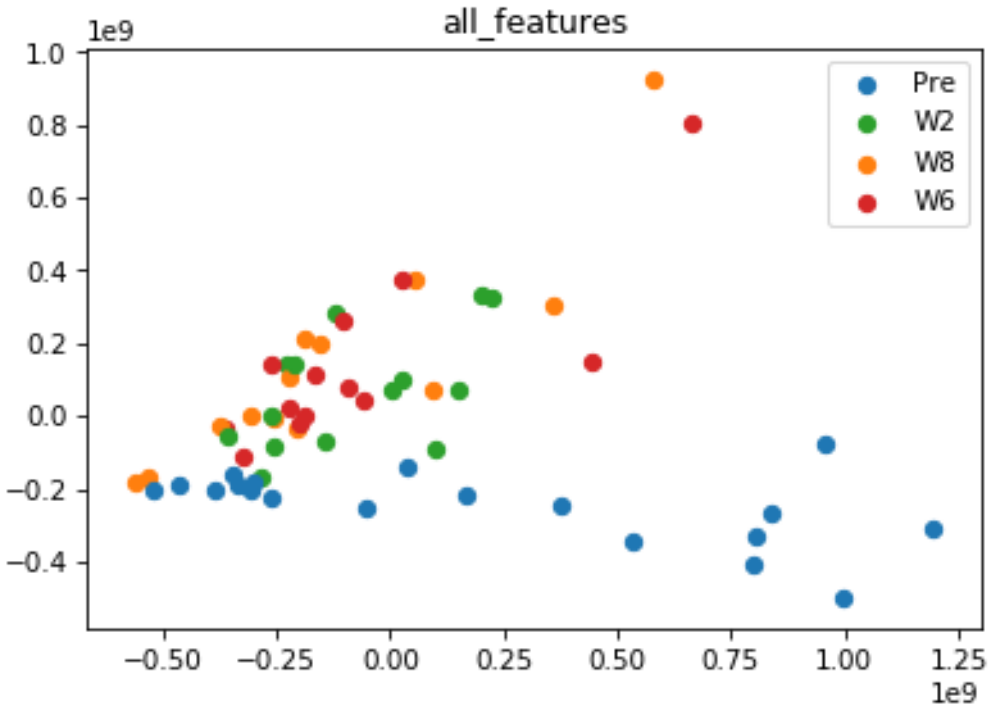
PCA was the first unsuperwised clusterisation method, apllied to the dataset:

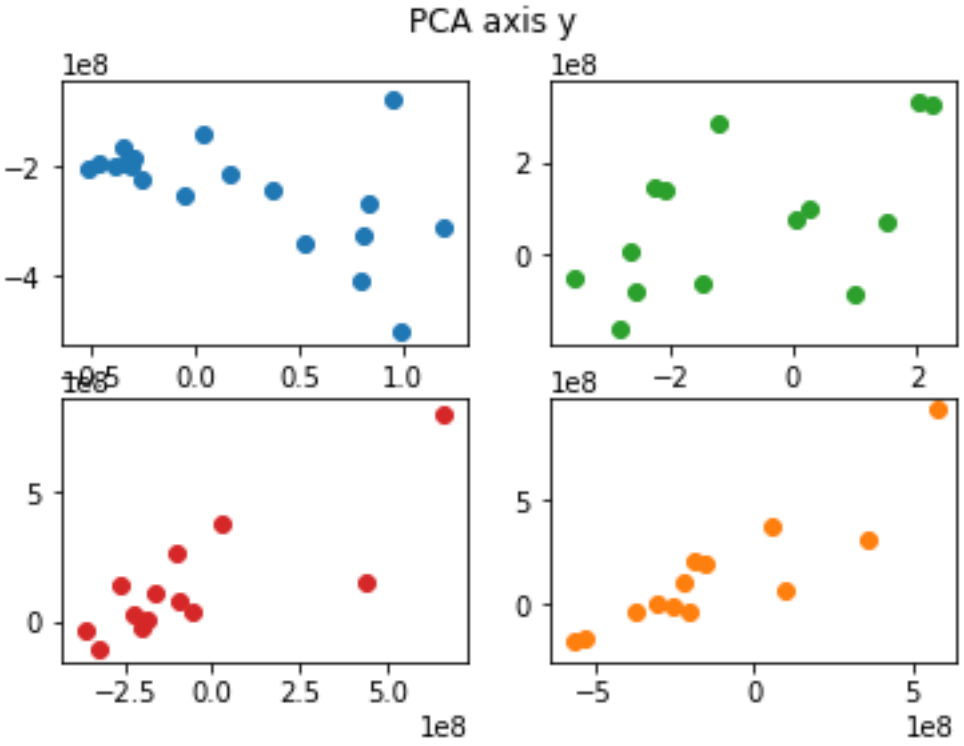
All signals

|

Signals plotted by class

- -



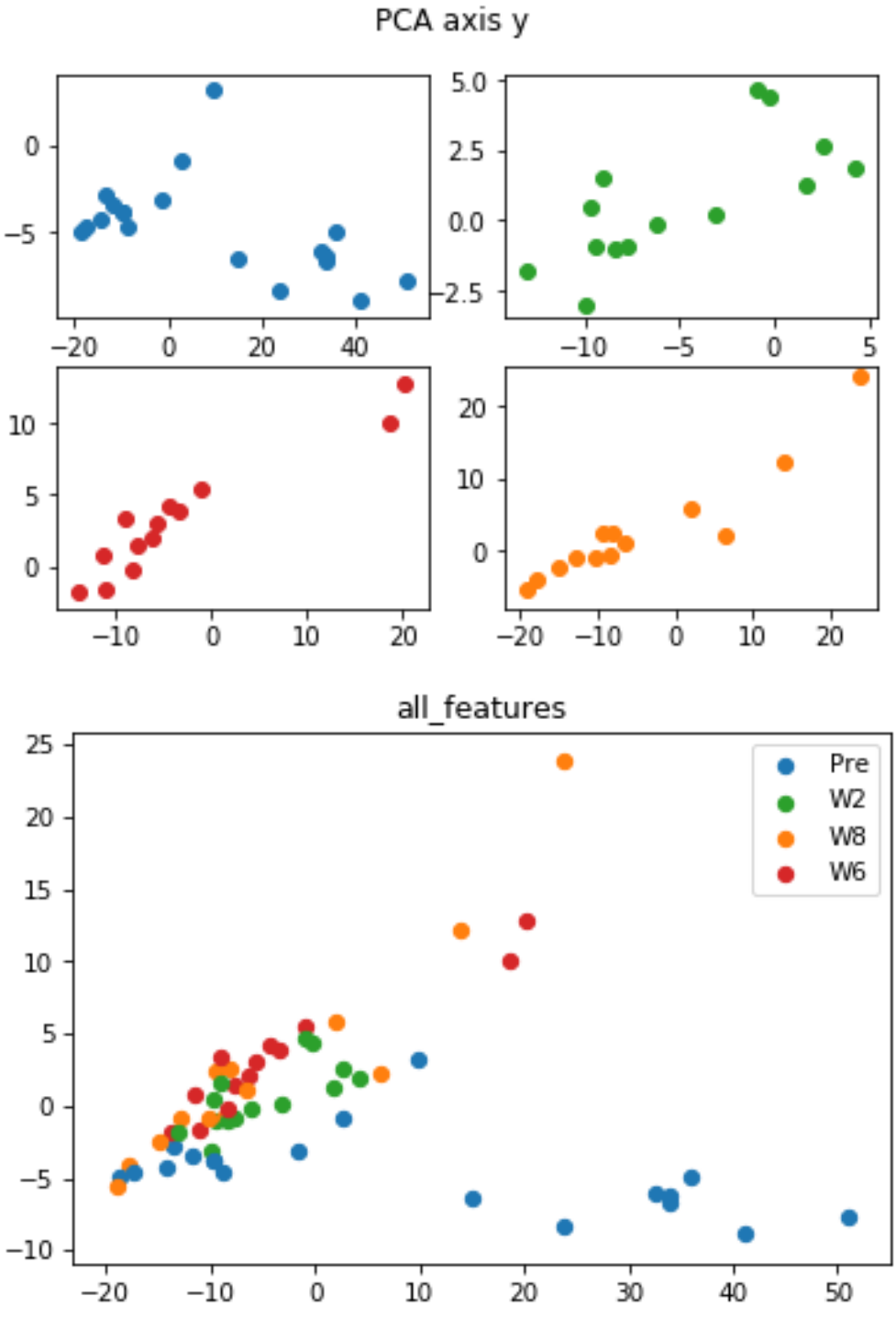


All signals - scaled

|

Signals plotted by class - scaled

- -

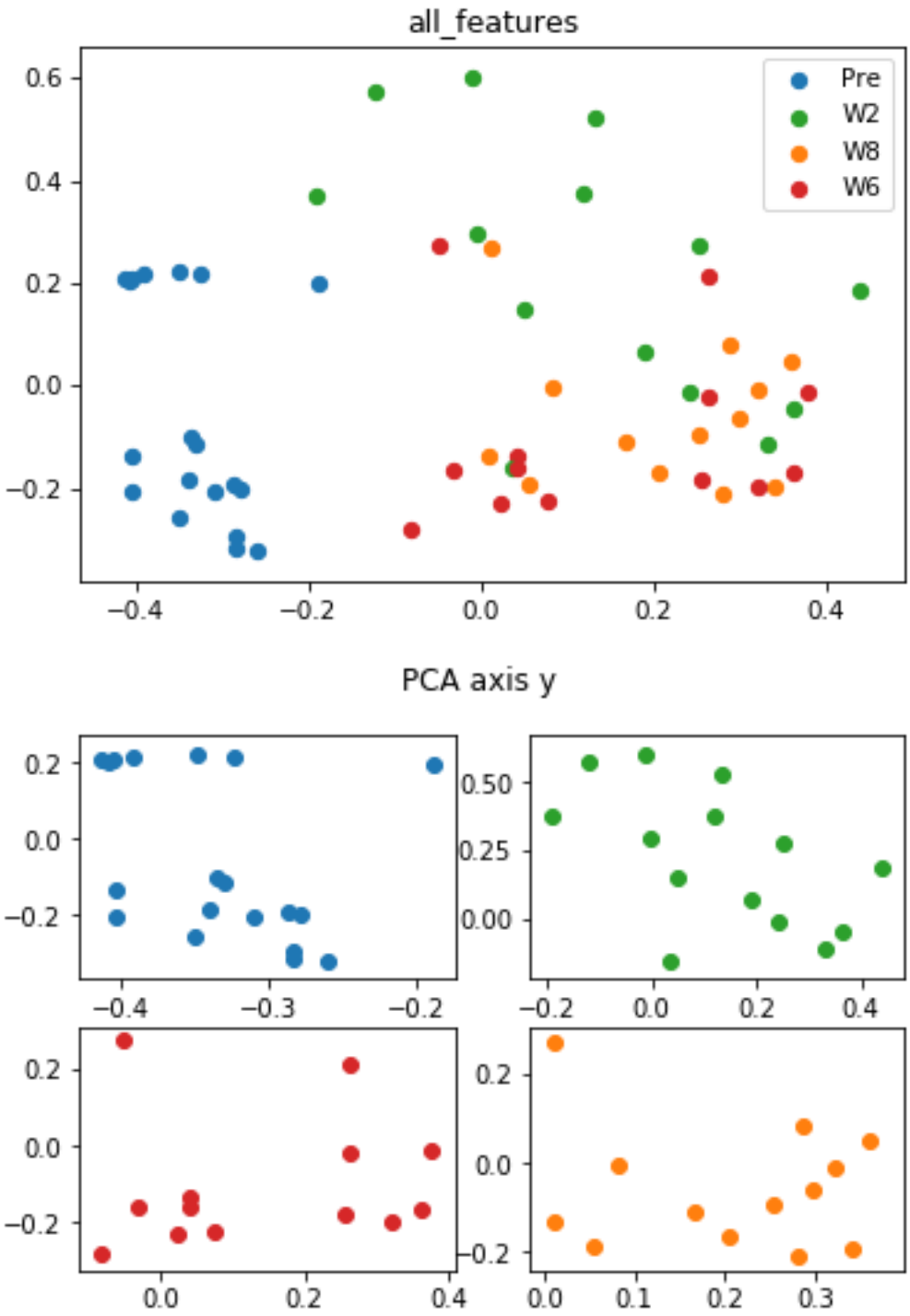


All signals - normalized

|

Signals plotted by class - normalized

- -



Pic.4

Pic.5

Only the class “Pre” shows a clear pattern in PCA. For the other three is impossible to define affiliation to any of other classes with accetable error rate.

1.2.2 2.2 K-Means Neighbours

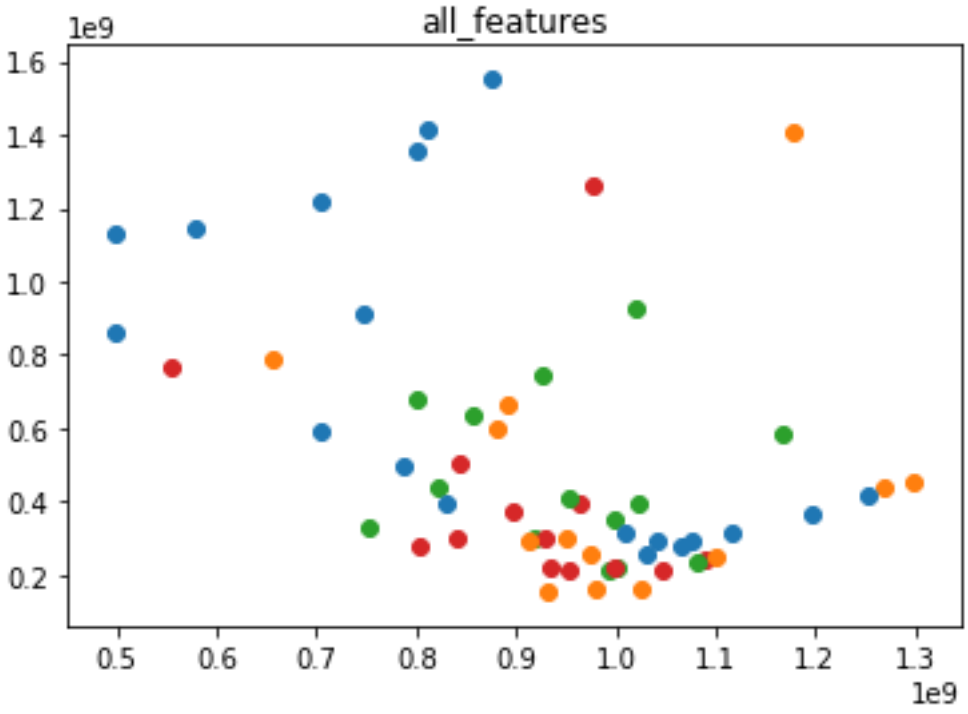
K-Mean neighbours has neither shown satisfied results (Pic.6 and Pic.7)

All signals

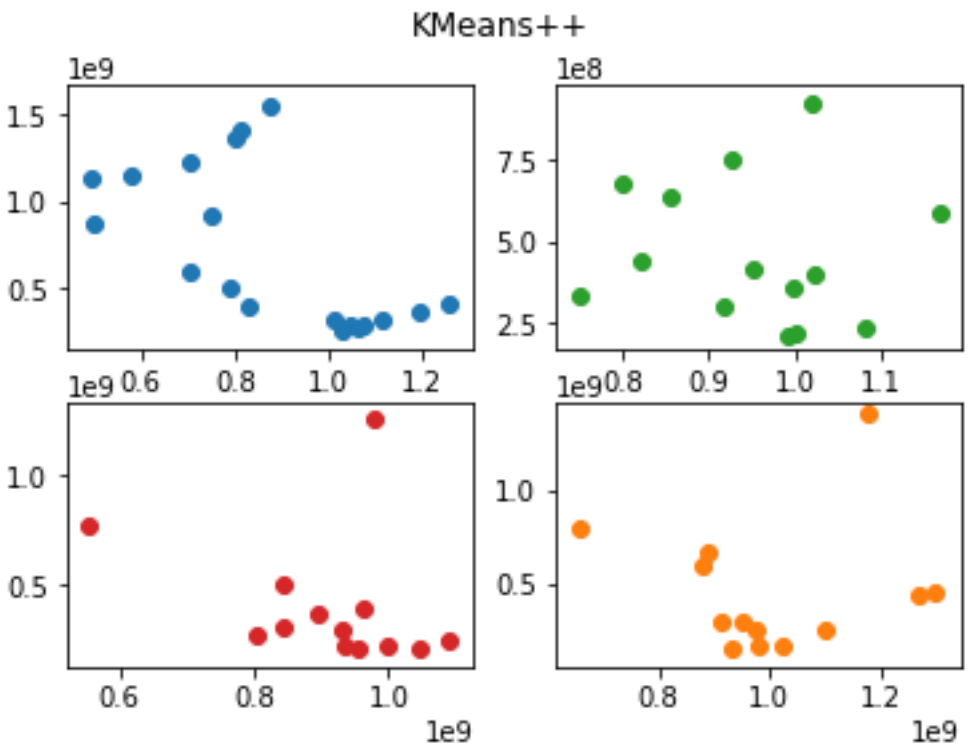
|

Signals plotted by class

- -



|

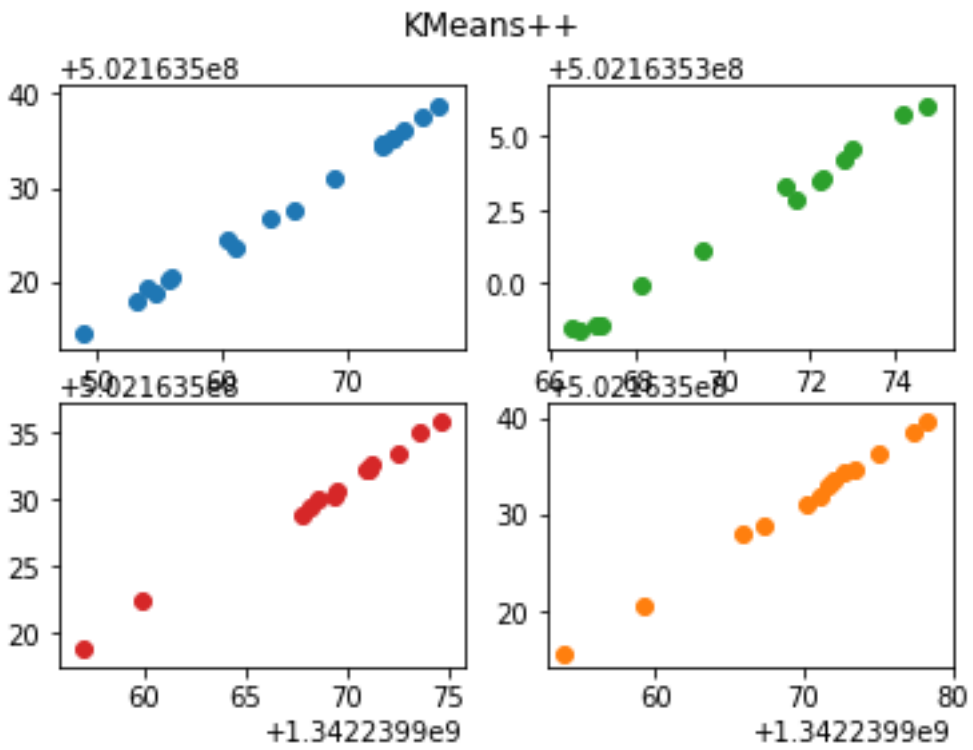


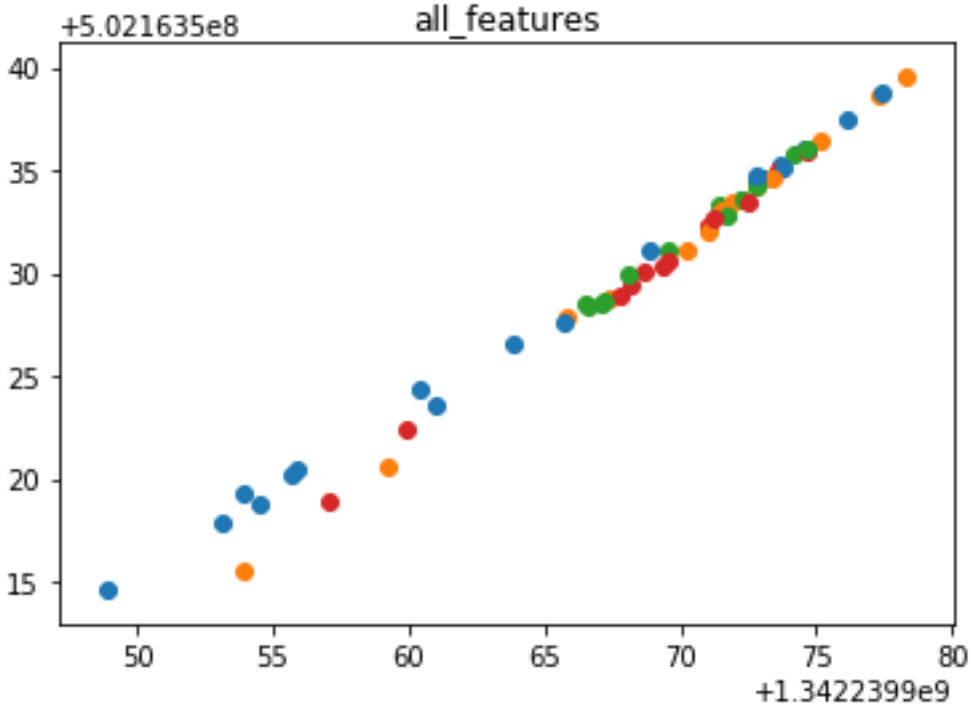
All signals - scaled

|

Signals plotted by class - scaled

- -



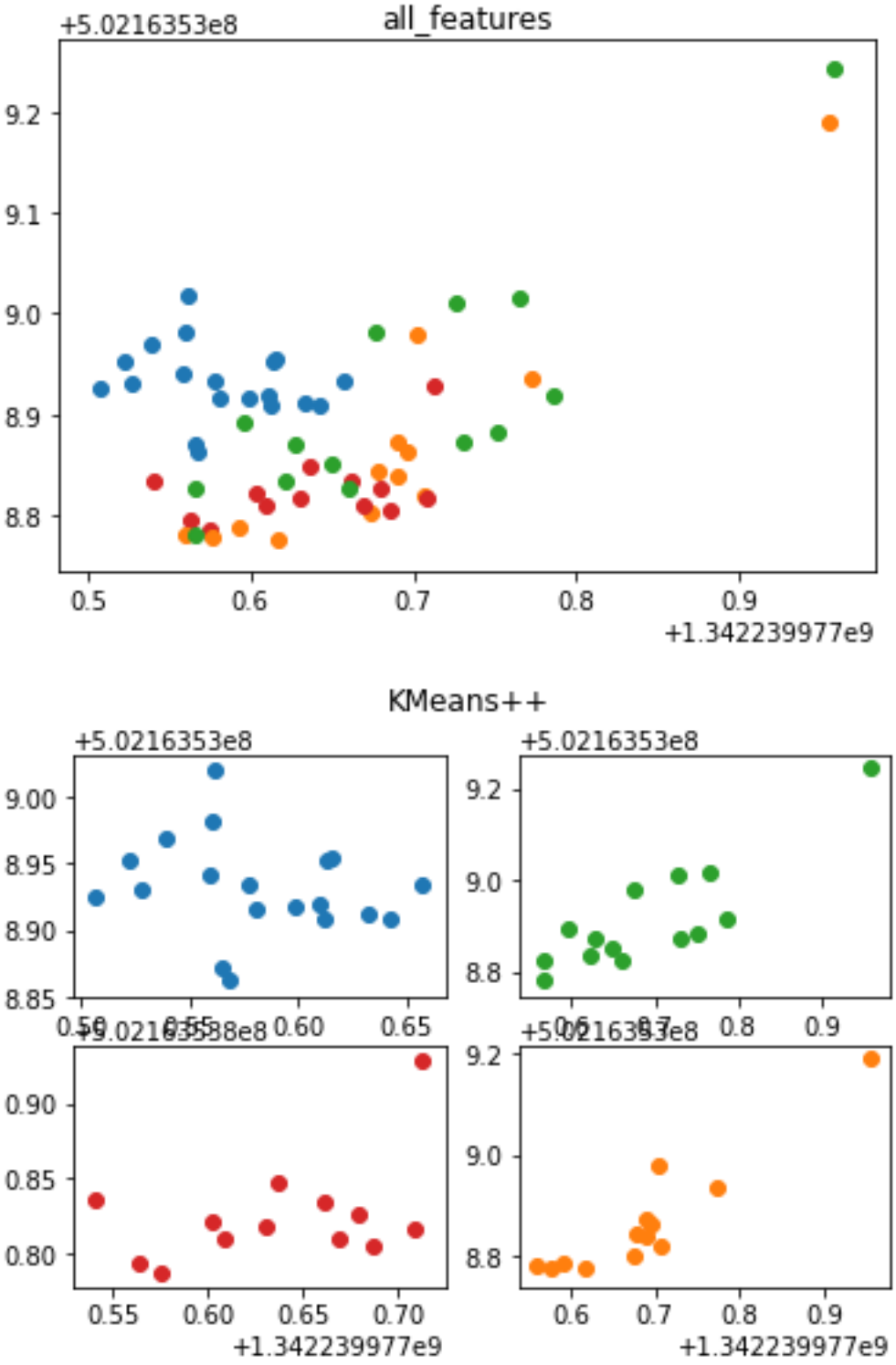


All signals - normalized

|

Signals plotted by class - normalized

- -

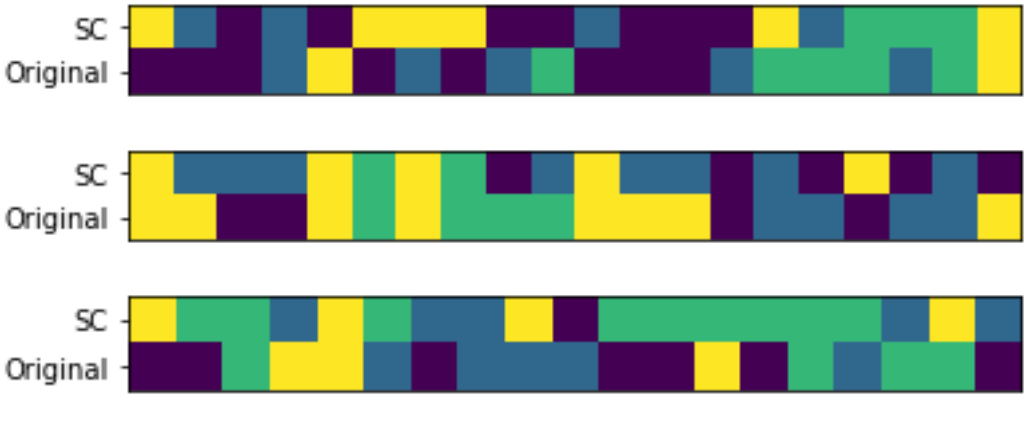

all_features
+5.0216353e8
9.2
9.1
9.0
8.9
8.8
0.5
0.6
0.7
0.8
0.9
+1.342239977e9
KMeans++
+5.0216353e8
9.00
8.95
8.90
8.85
0.50
0.55
0.60
0.65
+5.0216353e8
9.2
9.0
8.8
0.6
0.7
0.8
0.9
0.90
0.85
0.80
0.55
0.60
0.65
0.70
+1.342239977e9
9.2
9.0
8.8
0.6
0.7
0.8
0.9
+1.342239977e9

Pic.6

1.2.3 2.3 Spectral clustering

Try to project K-Means on Laplacian: spectral clustering:

Spectral clustering



Pic.7

Resulting accuracy is about 33%

1.2.4 2.4 Clusterisation Results and Discussion

Presented clusterisation methods do not provide satisfactory results. Means either they do not have needed feature sensivity or there are simply no sistematic elements in presented classes. The next logical step would be to apply geometrical feature analysis to the dataset.

1.3 3. Geometrical Feature Search

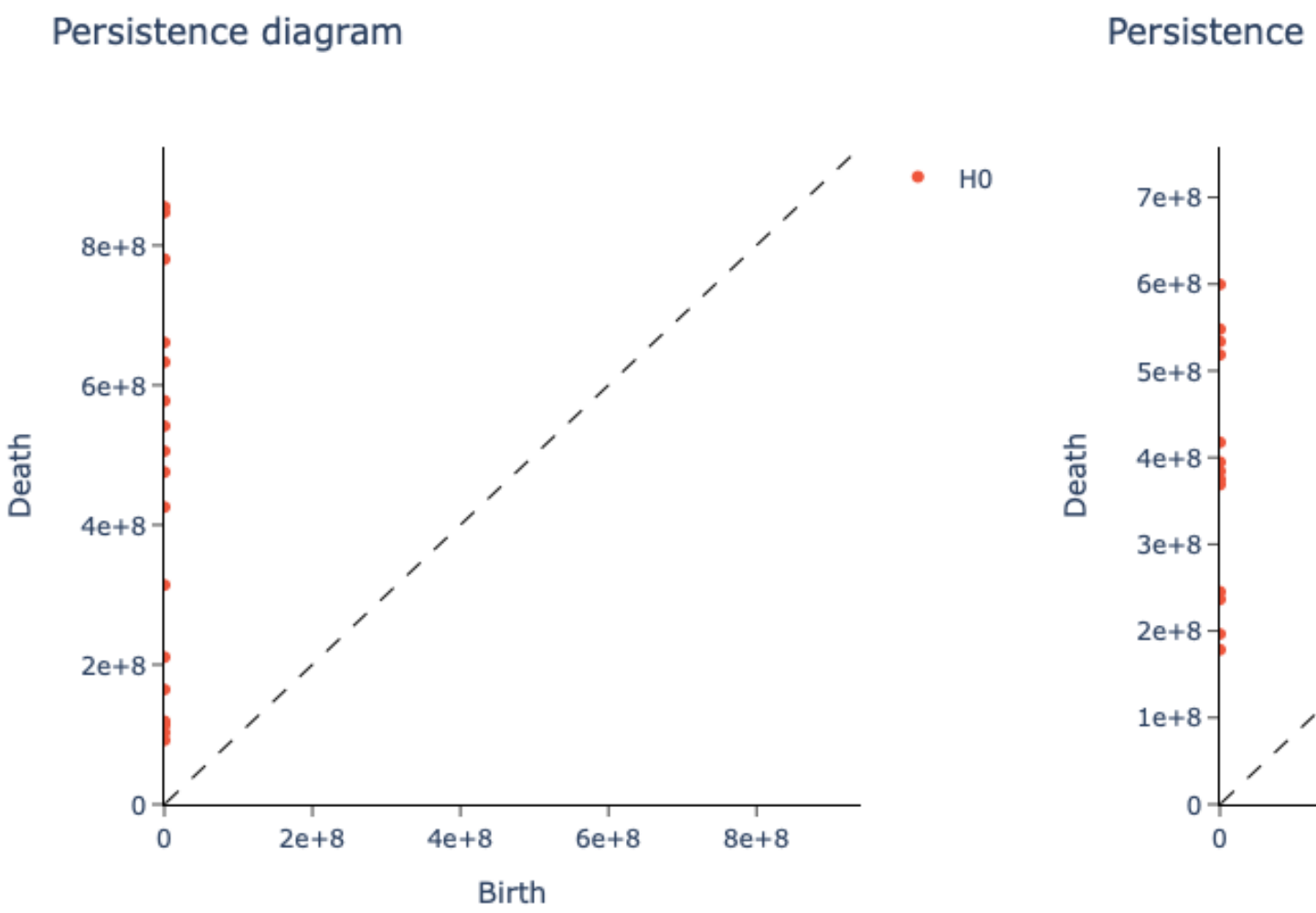
In this analysis the presented dataset will be analysed class-wise to find out if there are some significant feature which could be enough for more sensitive clusterisation or classification.

1.3.1 3.1 Persistence diagrams

Persistence homology is well known as a tool which allows to extract true features rather than artifacts and/or noise. Results of homological analysis is presented on the Pic.9

Pre

W2

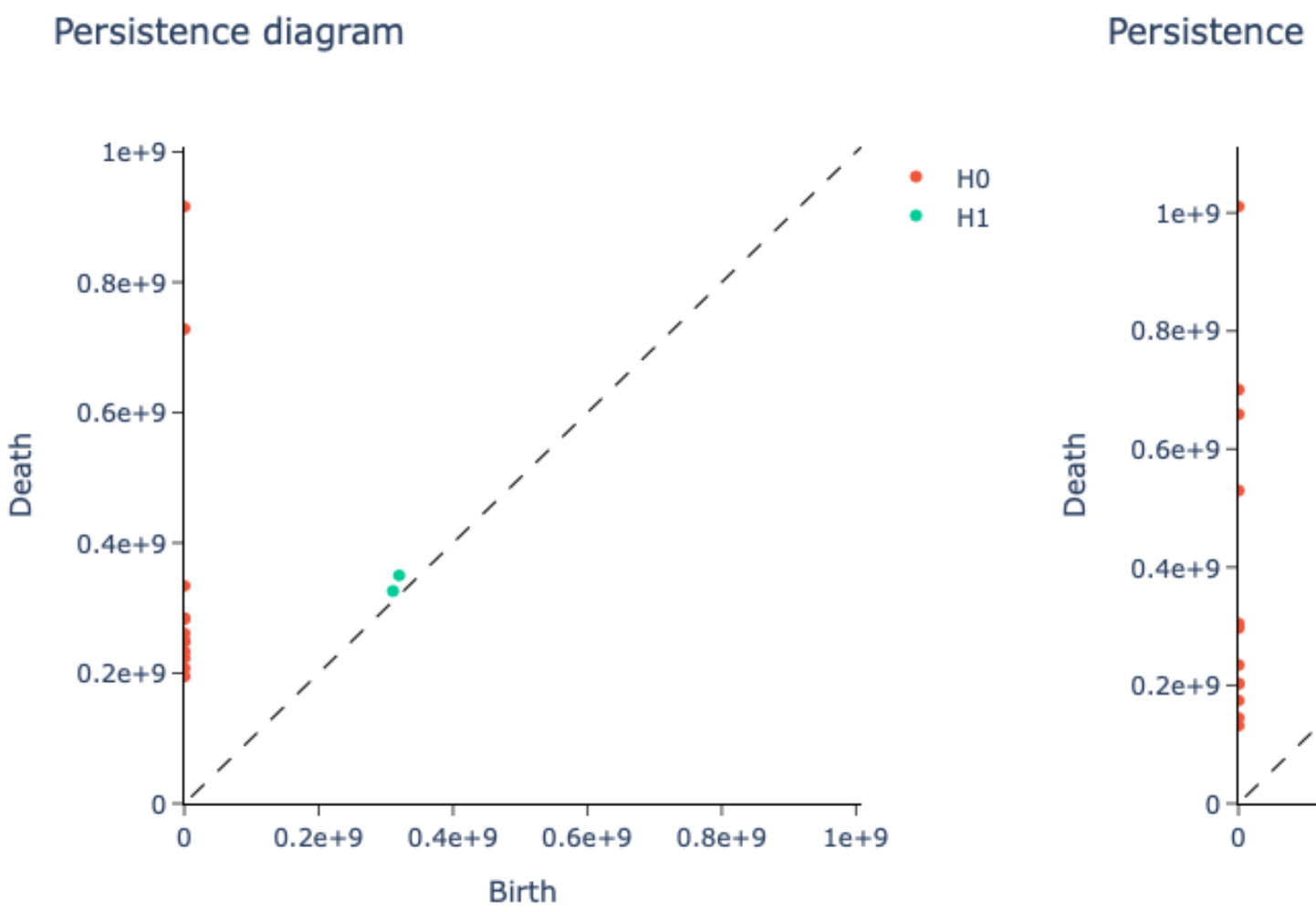

Persistence diagram
Persistence
H0
Death
Birth
0
2e+8
4e+8
6e+8
8e+8
1e+8
3e+8
5e+8
7e+8

|

W6

|

W8



Pic.8

Persistent homology shows clear difference between presented samples.

1.3.2 3.2 Persistence entropy

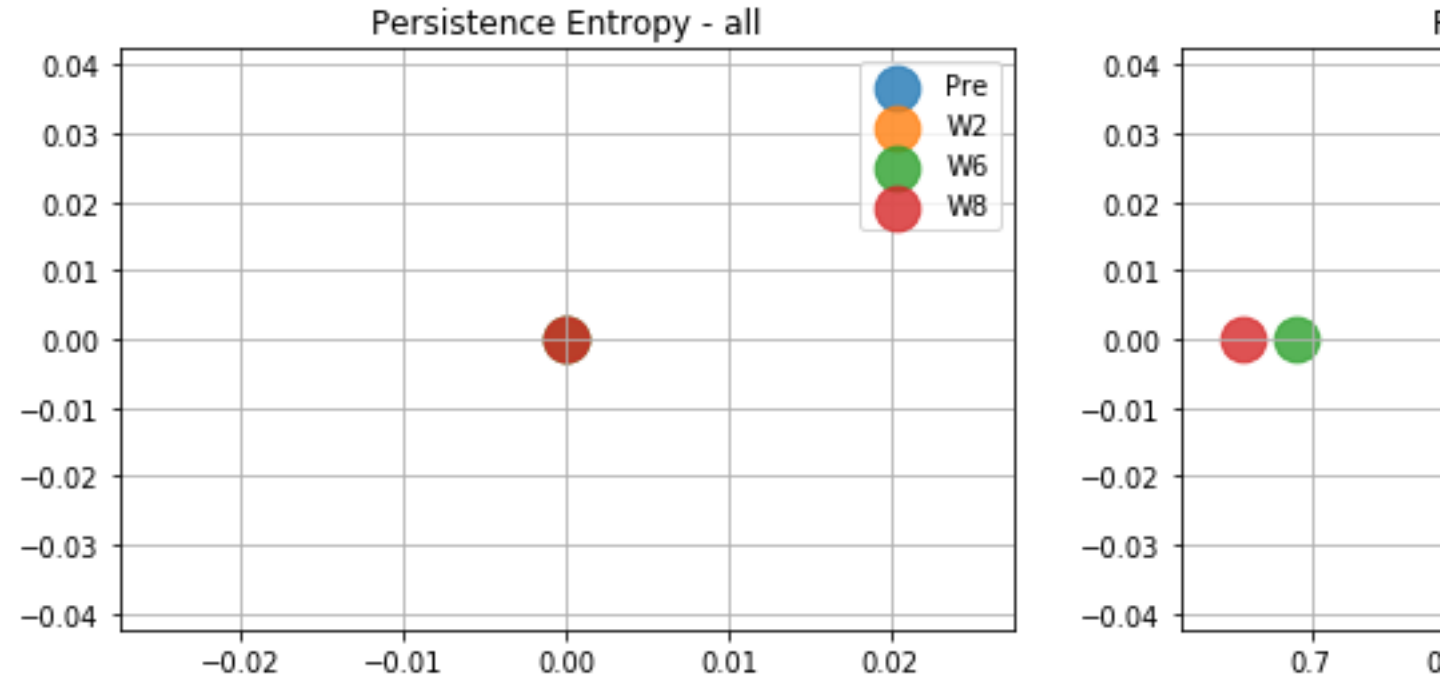
To be sure about results obtained with persistence persistence diagrams, will be also applied persistence entropy (Pic.10)

|

10% samples

|

30% of samples

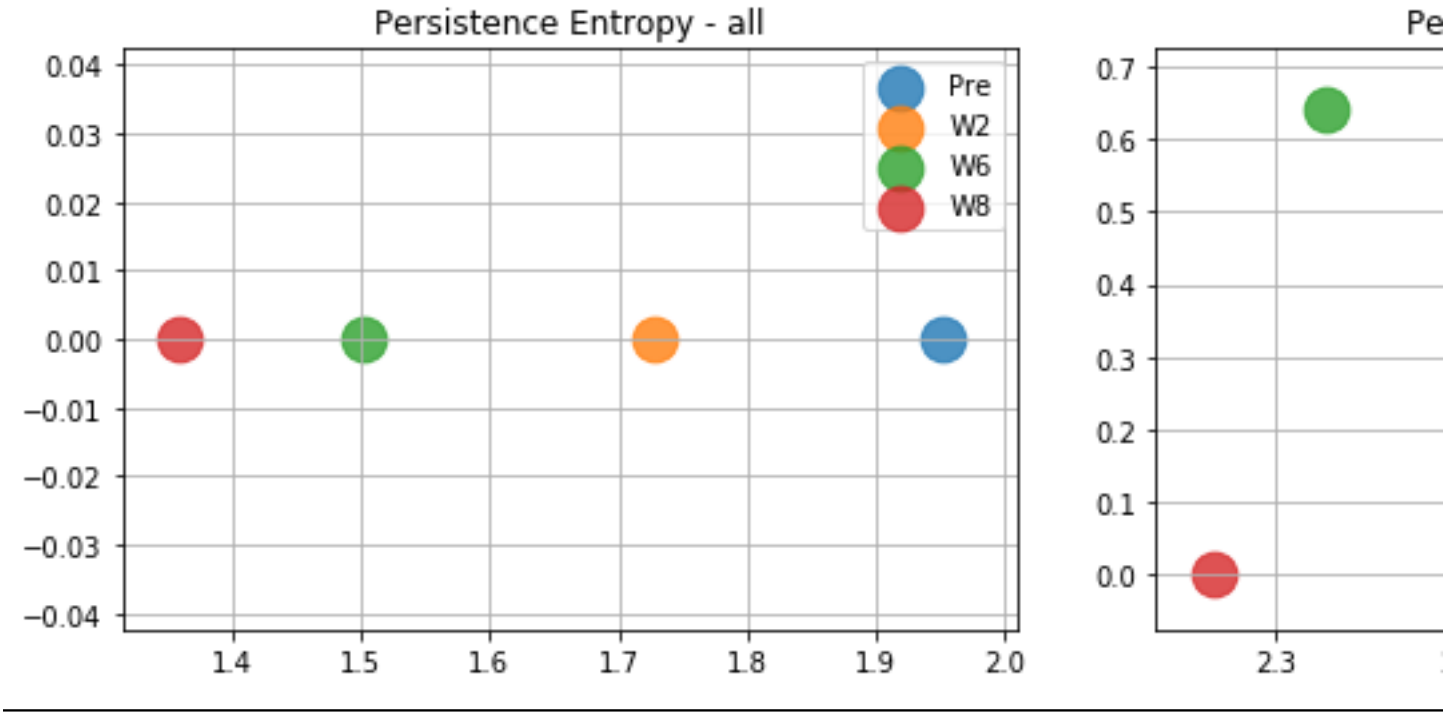


|

50% samples

|

all samples



Pic.9

This method also shows clear feature difference for all the four classes.

1.4 Worth to try:

- Persistence entropy with preprocessed (projected) signals
- Fourier transformation
- Taken's embedding for each class

1.5 4. Feature Extraction with Convolutional Neural Network

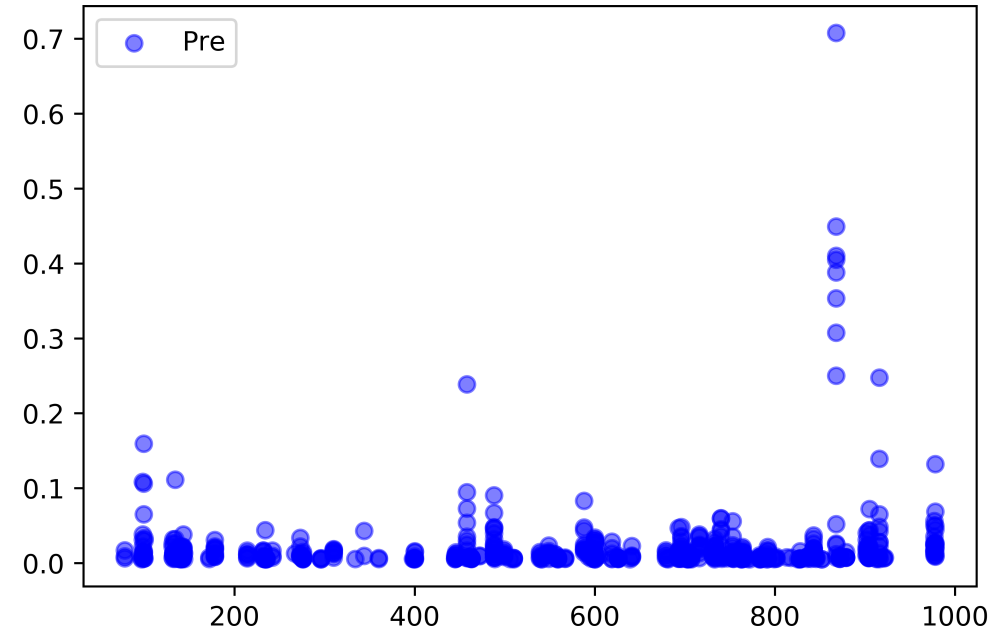
The next logical step would be prove if a deep learning method is able to catch the features of deformed signals in presented dataset (and experience says - yes, it can). But a check is needed. As feature extractor was used pretrained ResNet50, thresholded results are presented on the Pic.11:

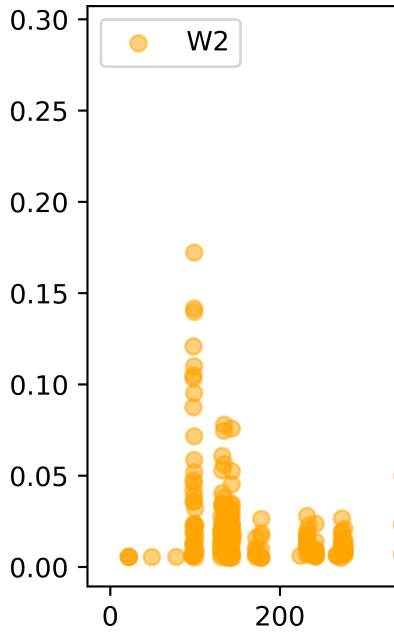
|

Pre

|

W2


Pre
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
200
400
600
800
1000

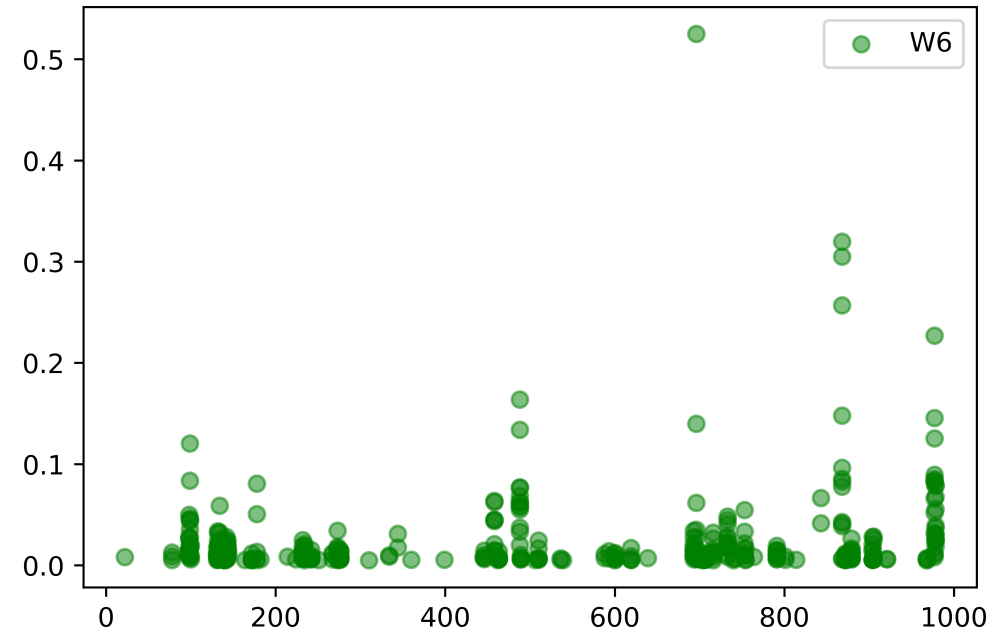

W2
0.30
0.25
0.20
0.15
0.10
0.05
0.00
0
200

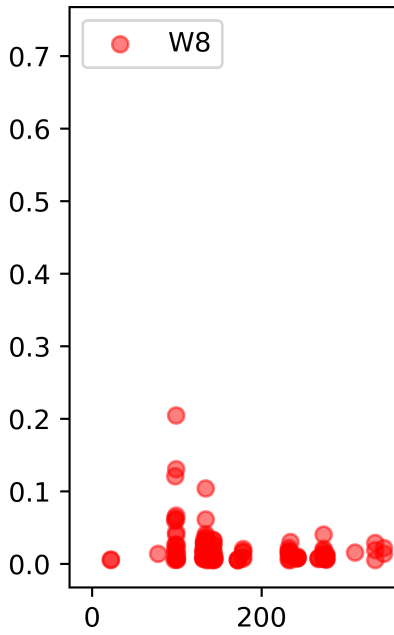
|

W6

|

W8


W6
0.5
0.4
0.3
0.2
0.1
0.0
0
200
400
600
800
1000


W8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
0
200

Pic.10

NN seems to be sensitive to the features of signal (even with limited input, which in case of the most presented signal - “Pre” is only 19). Anyway, a classification worth to try.

|

488

|

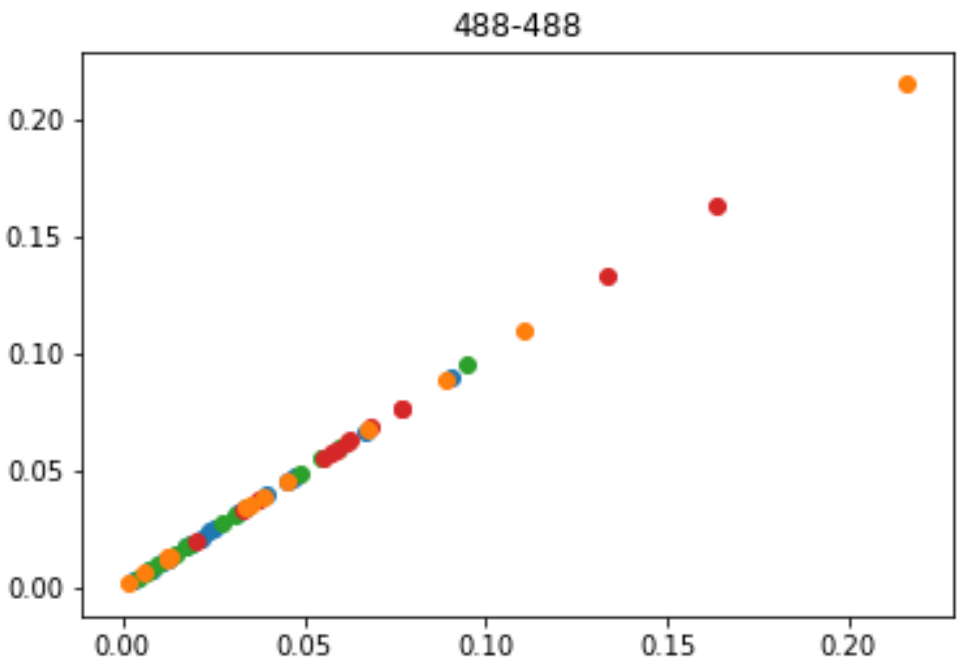
696

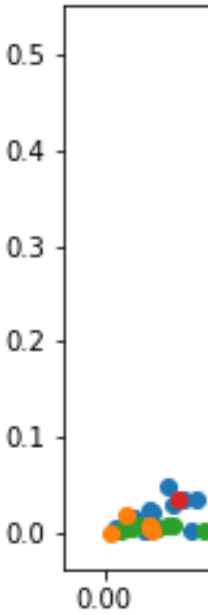
|

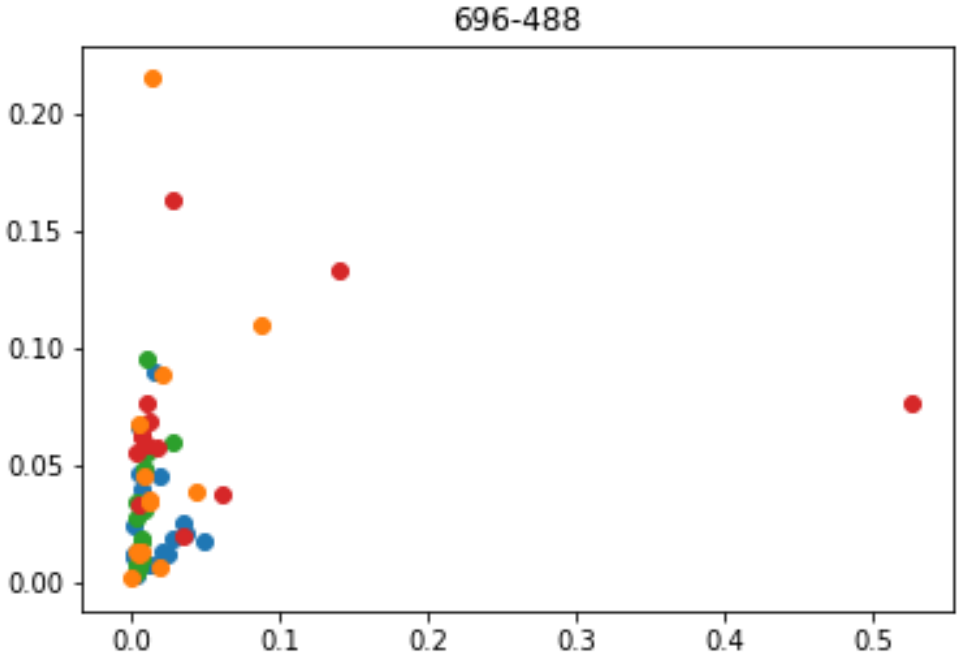
868

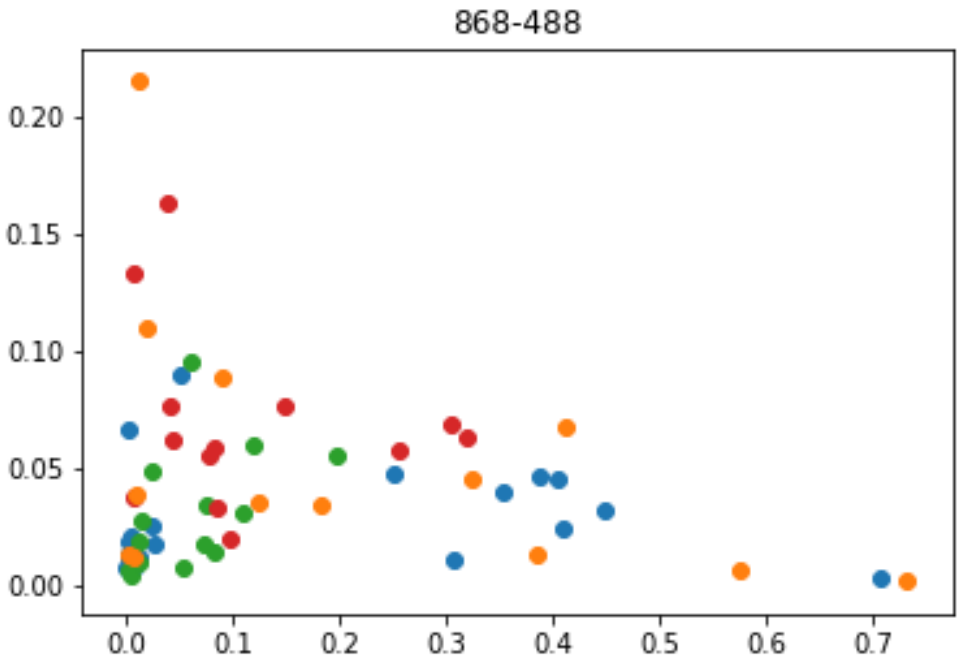
|

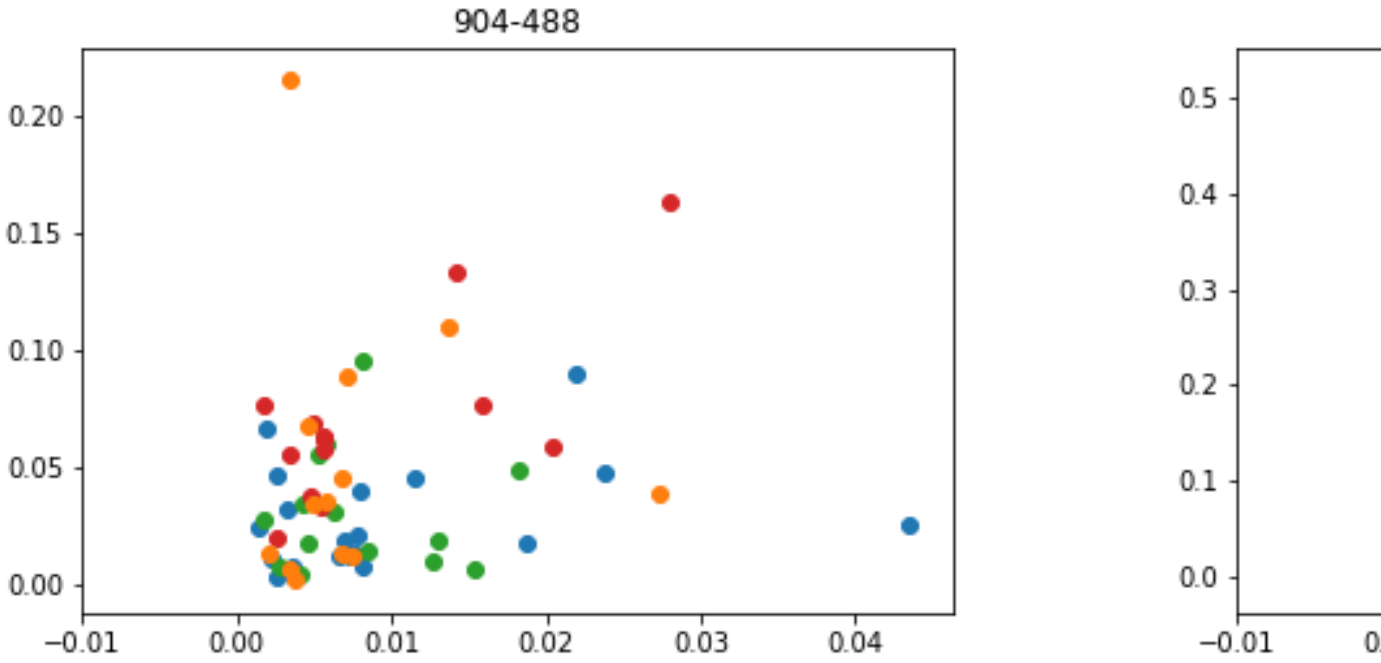
904






696-488
0.20
0.15
0.10
0.05
0.00
0.0
0.1
0.2
0.3
0.4
0.5


868-488
0.20
0.15
0.10
0.05
0.00
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7



Pic.11

Plots from Pic.12 shows coordinates from the most represented NN outputs. As expected, a NN extracted also the most represented features of signals. ### Worth to try: - Clustered anomaly detection for the multidimentional output space.

1.6 5. Signal Classification with Neural Net

1.6.1 5.1 Dataset Augmentation

The available dataset has limited number of samples (min 13, max 19) per class.

For data augmentation can be applied to the dataset with the following algorithm: 0. Find “Peaks-zones” 1. Define number of transformation N = [0,n]; 2. Apply the stretch transformation ST = [-20,20] N times to the “Peaks-zones” 3. Stretch “zones without signals” to compensate ST

[ ]: