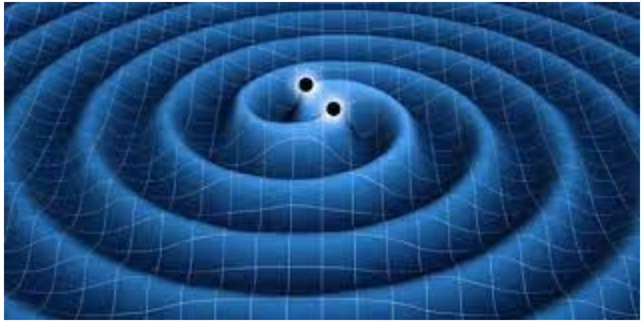


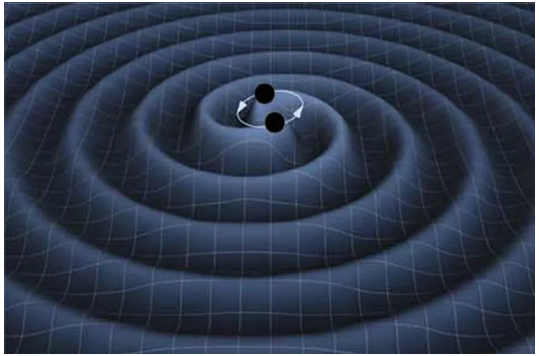
SAPIENZA
UNIVERSITÀ DI ROMA

Gravitational-Waves Glitch Detection : A Deep Learning Approach

- Senad Beadini
Master's degree in Computer Science

Gravitational Waves :

- Ripples of space-time :
    - Ex : A huge mass that rotates, Binary Black Hole



- According Einstein's theory, gravity = curvature of space-time

Can we detect Gravitational Waves ?

It is hard.. but yes!

LIGO :

- **L**aser **I**nterferometer **G**ravitational-Wave **O**bservatory

- It's able to detect the passage of gravitational waves.

It requires high-precision tools, sensitive to distances less than the length of the atomic nuclei !!!!

- In 2015, LIGO/VIRGO detect the first GW from a binary black hole merger.

Reference :

- Abbott, Benjamin P., et al. "Observation of gravitational waves from a binary black hole merger." *Physical review letters* 116.6 (2016): 061102.

Problems and challenges :

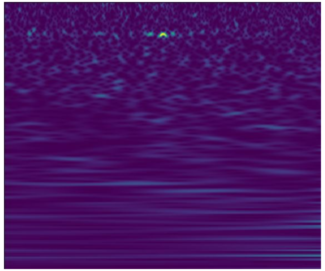
- High sensitivity ---> +likelihood of errors and noise

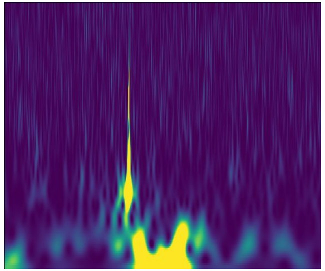
GW detectors are vulnerable to particular artifacts called glitches :

- caused by e.g. small ground motions, ringing of the test-mass suspension system at resonant frequencies, or fluctuations in the laser

These artifacts produce : false-positive results in GW search, reduce the significance of candidate gravitational-wave signals, corrupt data etc.

2 type of Glitches :





It’s essential to develop automatic processes which are able to detect and recognize glitches.

Pipeline : Gravity Spy (Zevin, Michael, et al. (2017))

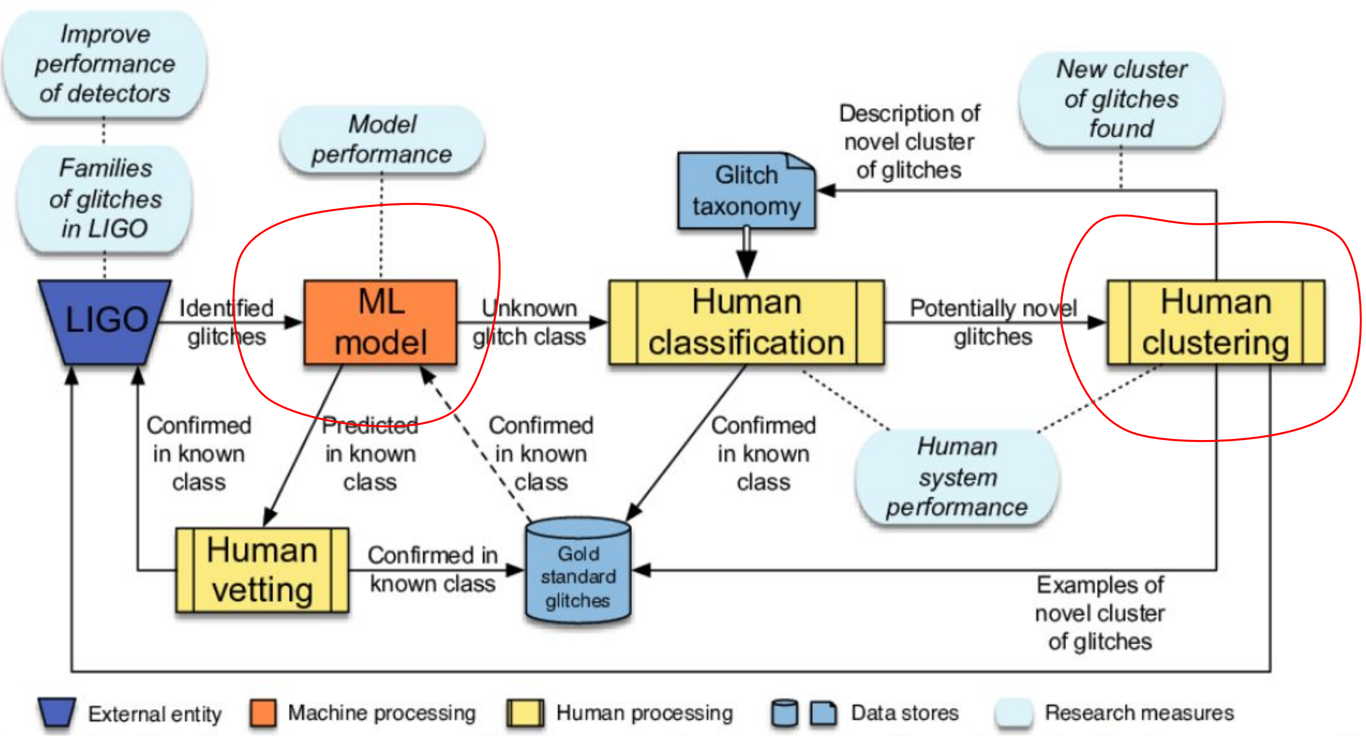


Figure 2: Gravity Spy system architecture, and overall data flow through the interconnected, interdisciplinary components of the project.

The aims of the project :

- We want (high accuracy) predictive models capable of recognizing glitches, to reduce human intervention as much as possible.

- We want to be able to recognize new types of glitches never seen before to help humans cluster new glitches never seen so far.

Let’s go further !

Why Big Data ?

- Huge amount of data, in 'observing' mode, LIGO generates terabytes of data every day ! [1]
- High velocity of generated data.
- Data can be in different format. Data usually are signals that need to be interpreted and processed.
- The data of different GW detectors usually are used to study better the results (e.g. LIGO, VIRGO and KAGRA)
- A GW detector costs Billions $. Data has been gathered by LIGO (and others) has an incalculable value from the point of view of science.

5 Vs of Big Data hold.

[1] https://www.ligo.caltech.edu/page/ligo-technology

Dataset :

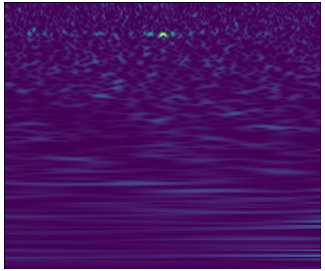
- kaggle dataset : https://www.kaggle.com/tentotheminus9/gravity-spy-gravitational-waves

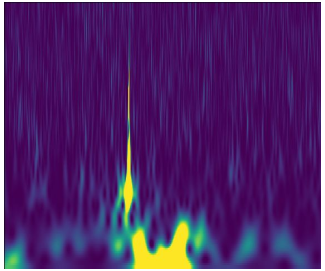
- It contains :
  - 22348 samples in the Train set
  - 4800 samples in the Validation set
  - 4720 samples in the Test set

roughly 4 GB

- There are 22 classes, 20 of them are different kind of glitches.
  - Two of them are :
    - “No_Glitch” : It contains pure signals without any glitch.
    - “None_of_the_above” : It contains unknown glitch never seen so far.

2 type of Glitches in Train set:





Related work :

[1]   Zevin, Michael, et al. "Gravity Spy: integrating advanced LIGO detector characterization, machine learning, and citizen science." *Classical and quantum gravity* 34.6 (2017): 064003

- Great result using CNN, however they use 20 classes and low accuracy for unseen glitches

[2] George, Daniel, Hongyu Shen, and E. A. Huerta. "Deep Transfer Learning: A new deep learning glitch classification method for advanced LIGO." *arXiv preprint arXiv:1706.07446* (2017).

- Transfer learning using S-o-A models in CV (ResNet, Inception etc) + proposed a clustering apprch.

[3]   Other recent approach using generative models.

In this project : I reproduce the results of [2]

Transfer Learning :

A deep learning technique which use a trained model to solve another similar task. Basically we exploit the knowledge of a model that has already learnt in a task for another similar problem.

However the training needs to be re-computed, we have less learnable parameters though.

Which models did I use ?

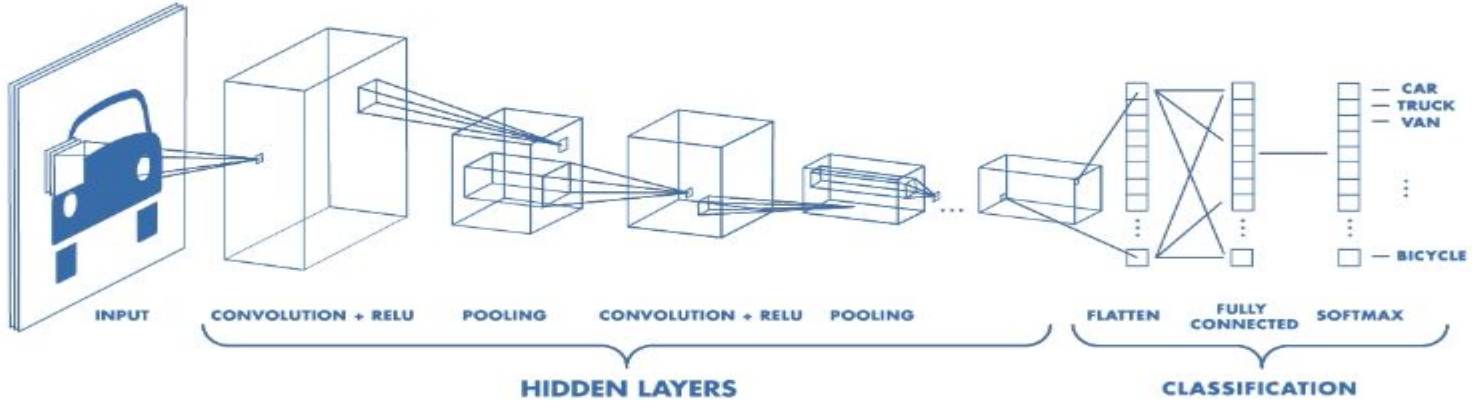
1) EfficientNetB0 : `20,438,553 parameters`
2) EfficientNetB1 : `38,584,982 parameters`
3) MobileNetV2 `18,646,966 parameters`

None of them used in [2]

For each model, training : #Epochs = 50, SGD, lr = 0.0001, momentum = 0.9

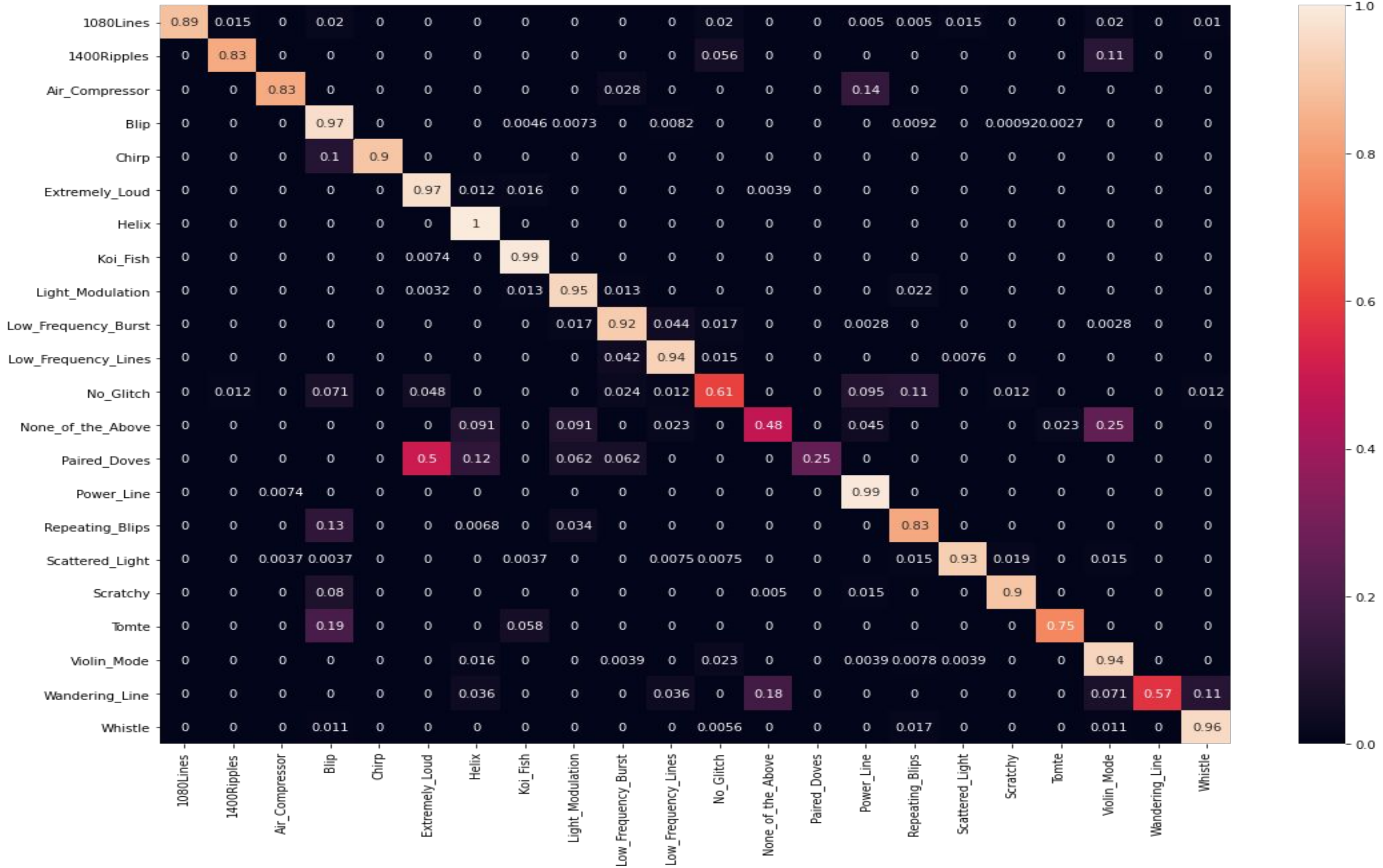
EfficientNetB0 :

- It’s a CNN and use residual blocks.



A standard convolutional neural network

EfficientNetB0 confusion matrix test set : (the best model)



Comparison with [2]

- They got 98.8 % accuracy with ResNet50 whereas I got 93%.
- However there are a few differences :
    - My test set is almost 3 times bigger.
    - My # epochs are half of them.
    - Limited computational power, just Colab :(

[2] George, Daniel, Hongyu Shen, and E. A. Huerta. "Deep Transfer Learning: A new deep learning glitch classification method for advanced LIGO." *arXiv preprint arXiv:1706.07446* (2017).

EfficientNetB0 metrics per single class :

- 14 classes have both Recall and Precision >= 0.89.

These 2 classes are crucial !!

This pair have the lowest Recall and Precision values.

How to deal those labels?

	precision	recall	f1-score	support
1080Lines	1.00	0.89	0.94	200
1400Ripples	0.88	0.83	0.86	36
Air_Compressor	0.91	0.83	0.87	36
Blip	0.94	0.97	0.96	1092
Chirp	1.00	0.90	0.95	40
Extremely_Loud	0.94	0.97	0.95	256
Helix	0.92	1.00	0.96	168
Koi_Fish	0.96	0.99	0.98	408
Light_Modulation	0.93	0.95	0.94	312
Low_Frequency_Burst	0.94	0.92	0.93	360
Low_Frequency_Lines	0.89	0.94	0.91	264
No_Glitch	0.67	0.61	0.64	84
None_of_the_Above	0.75	0.48	0.58	44
Paired_Doves	1.00	0.25	0.40	16
Power_Line	0.93	0.99	0.96	272
Repeating_Blips	0.77	0.83	0.80	148
Scattered_Light	0.98	0.93	0.95	268
Scratchy	0.96	0.90	0.93	200
Tomte	0.91	0.75	0.82	52
Violin_Mode	0.90	0.94	0.92	256
Wandering_Line	1.00	0.57	0.73	28
Whistle	0.97	0.96	0.96	180
accuracy			0.93	4720
macro avg	0.92	0.84	0.86	4720
weighted avg	0.93	0.93	0.93	4720

Clustering approach :

- Different ways. First of all we need an feature extractor since each data sample is a [479*579*3] d vector.

Exploiting the previous trained network as feature extractor.

Developing a new model in order to learn an embedding (PCA, AE, VAE)

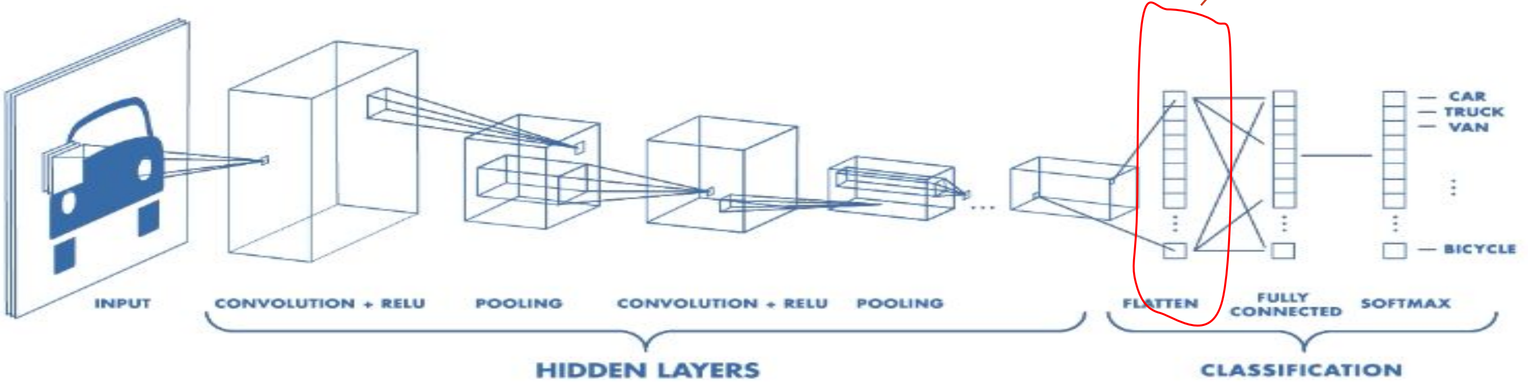
Following the [2], I did the first.

Clustering approach : DNN

- Build a feature extractor using the neural network.

- The idea : delete the last layers.

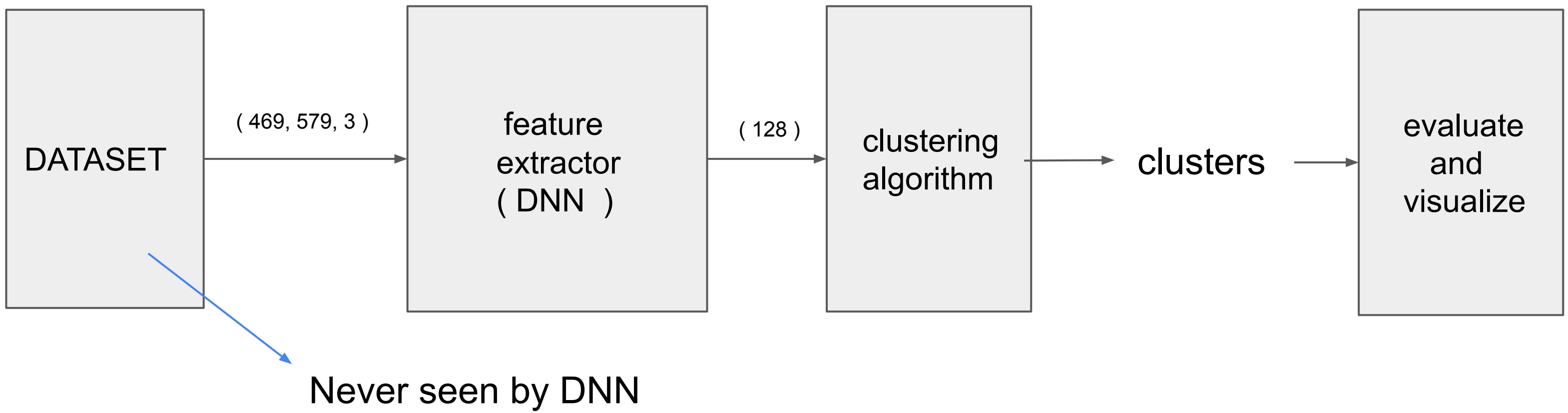
Using this layer as output



- In our network, each data sample is mapped into a 128 dimensional vector, there's a trade off (+dim +learnable param.)

Clustering approach :

- Once we have the feature extractor, I can use it to extract the latent features from the data.
- Then it's time to do clustering using an algorithm.
- Here I decided to exploit : 1) Kmeans 2) Gaussian Mixture Model



Clustering : evaluation

- Standard clustering metrics : Purity score and Silhouette score

Feature Extractor	Algorithm	Purity score	Silhouette score
DNN	Kmeans	0.6438	0.3174
DNN	Gaussian Mixture Model	0.6678	0.1942

Visualization :

- In this environment would be essential visualizing clusters and data (a human has to cluster data for a double check).

- How can we visualize clusters?

There exists some methods. A solution is t-SNE, proposed by [2] :

t-SNE : t-distributed stochastic neighbor embedding → **Non-linear dim. reduction method**

- Can we use PCA ? No → **Linear dim. reduction method**

[2] George, Daniel, Hongyu Shen, and E. A. Huerta. "Deep Transfer Learning: A new deep learning glitch classification method for advanced LIGO." *arXiv preprint arXiv:1706.07446* (2017).

t-SNE :

- It was introduced by Van der Maaten and Hinton in [1].

- Stochastic algorithm, strong (and difficult) math theory.

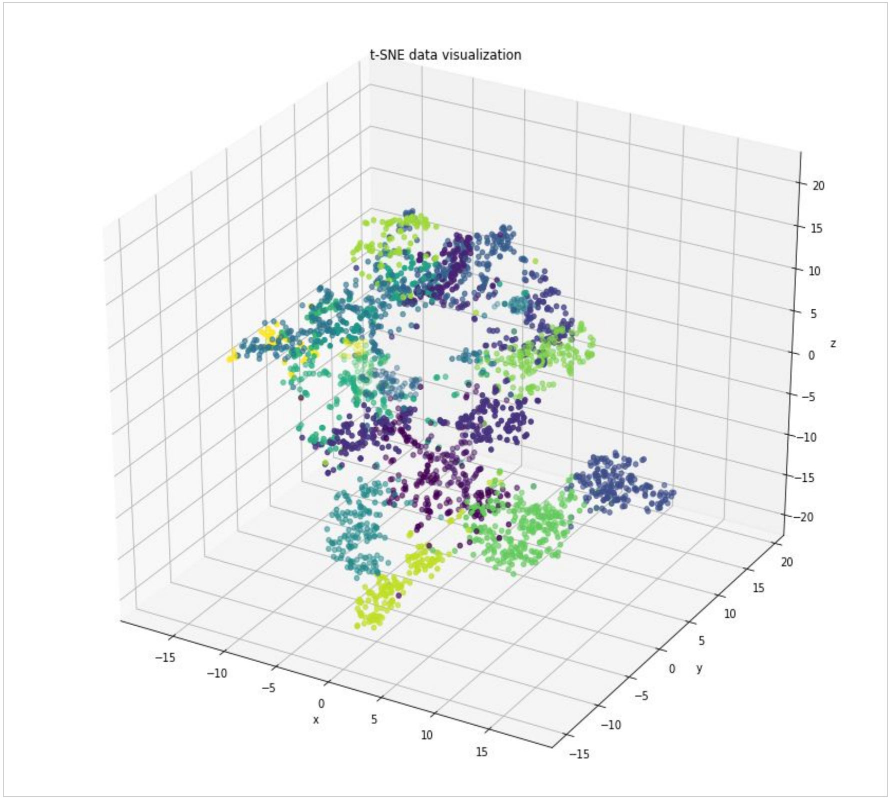
Main idea : it minimizes the **Kullback–Leibler** divergence between two distribution, the first one in the high dimensional space and the second in the reduced space.

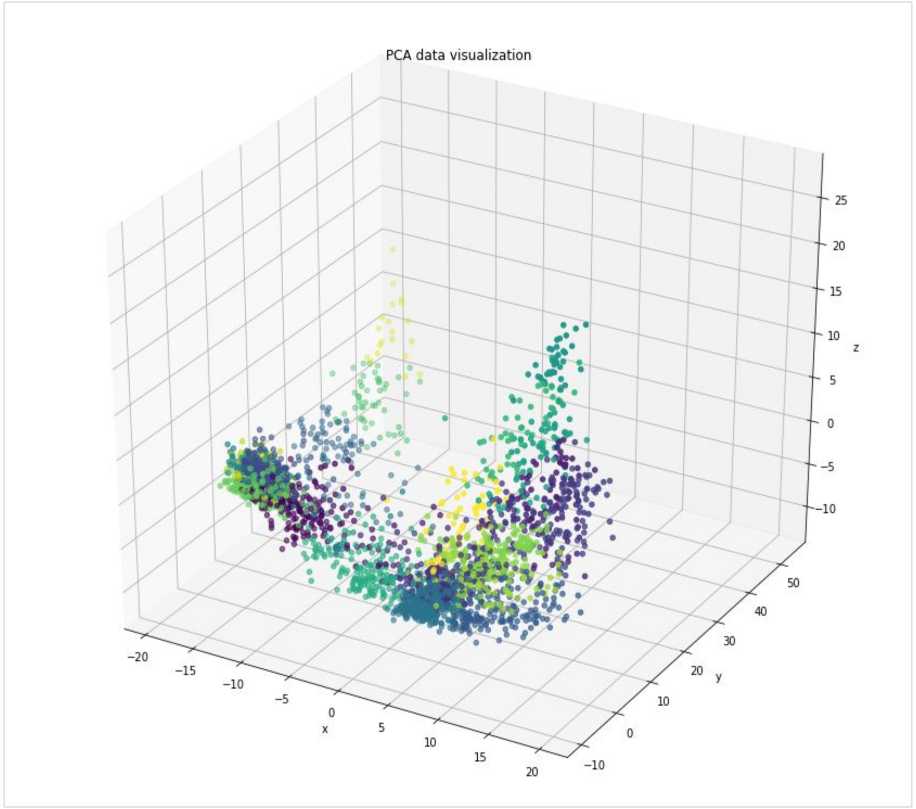
$$\boldsymbol{KL}\ (P \,||\, Q) = \sum P_{ij} \log(P_{ij} / Q_{ij})$$

Reference :

[1] Van der Maaten, Laurens, and Geoffrey Hinton. "Visualizing data using t-SNE." *Journal of machine learning research* 9.11 (2008).

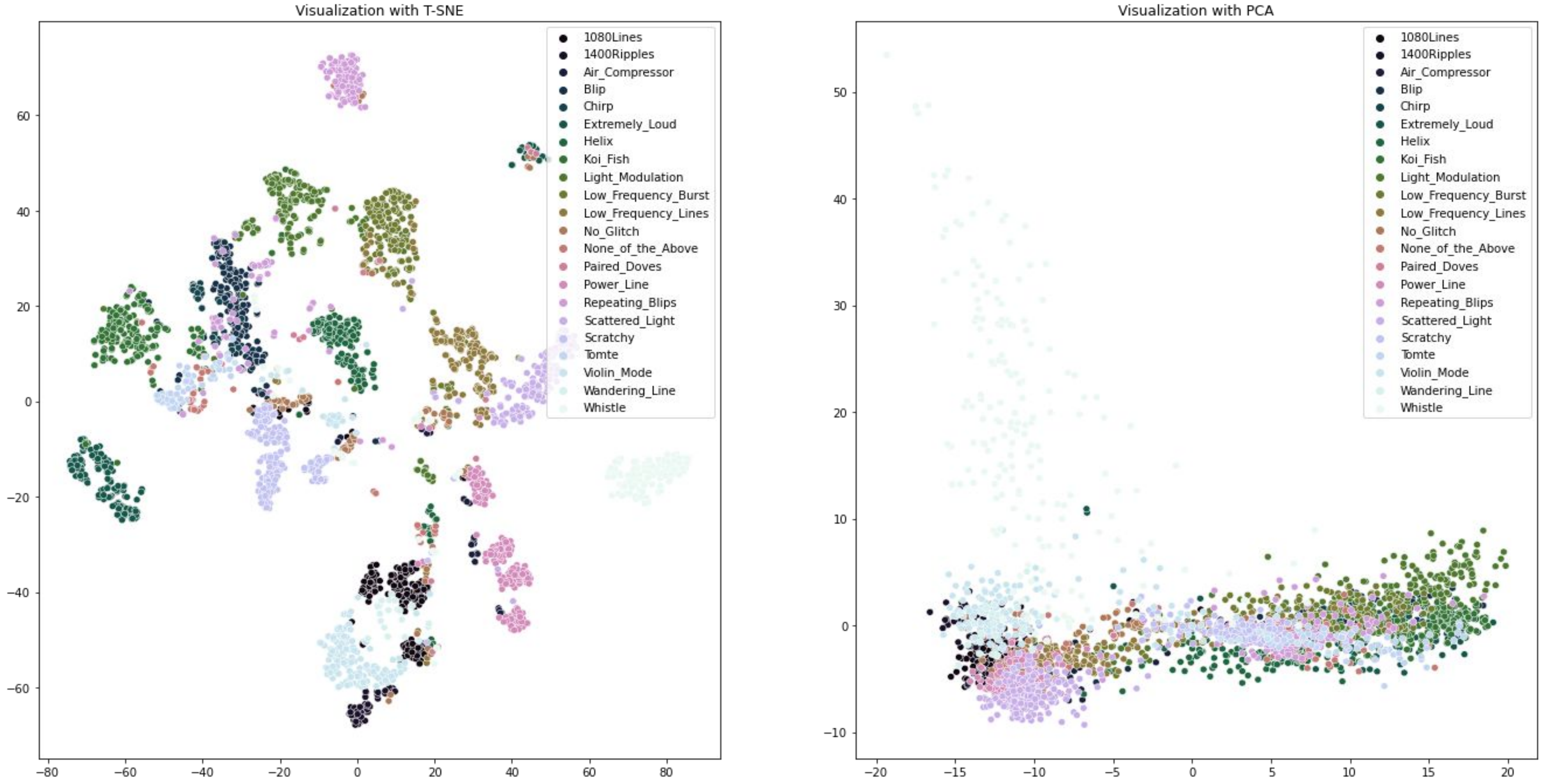
Clustering : visualization with t-SNE and PCA.



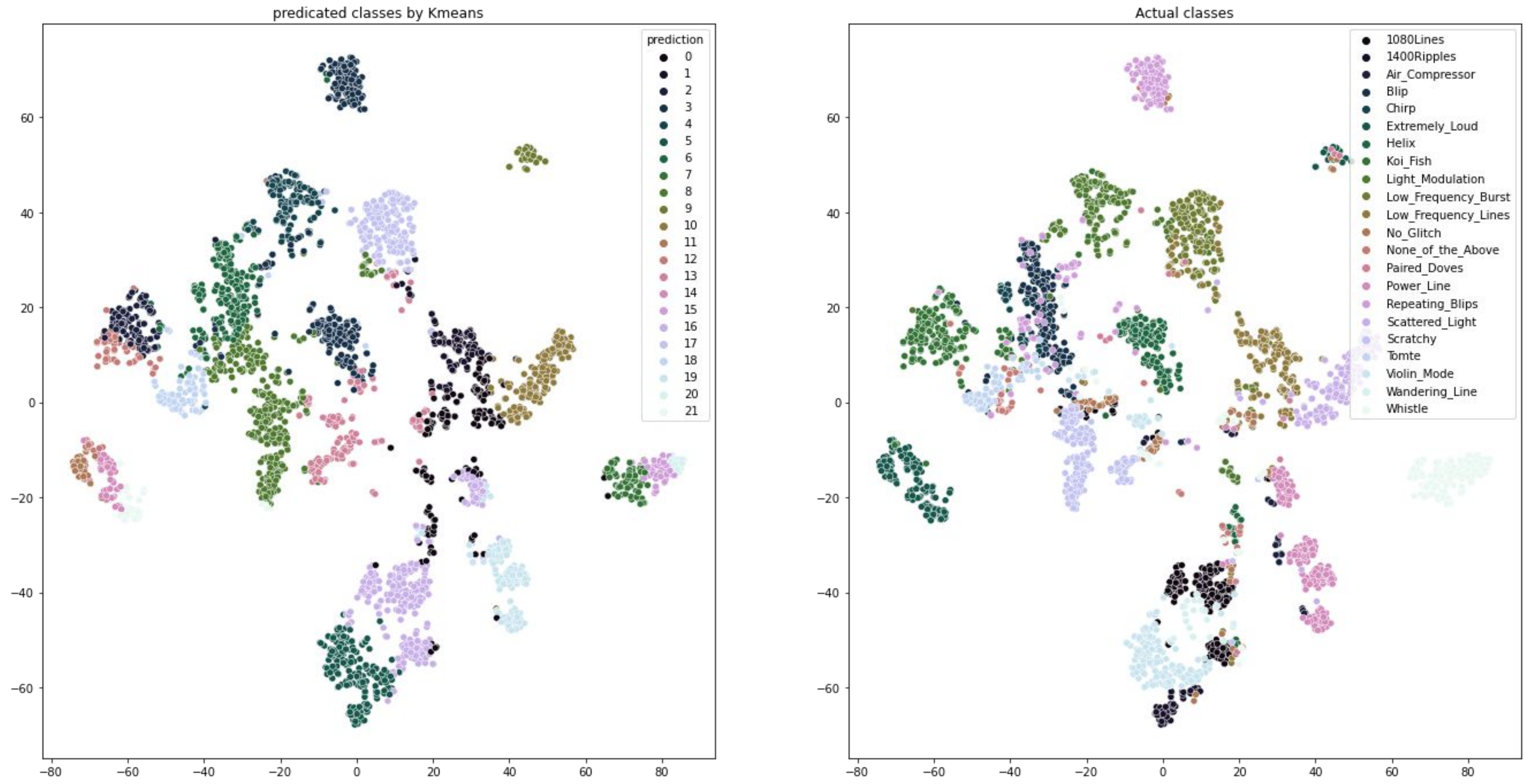


t-SNE seems good, PCA no...

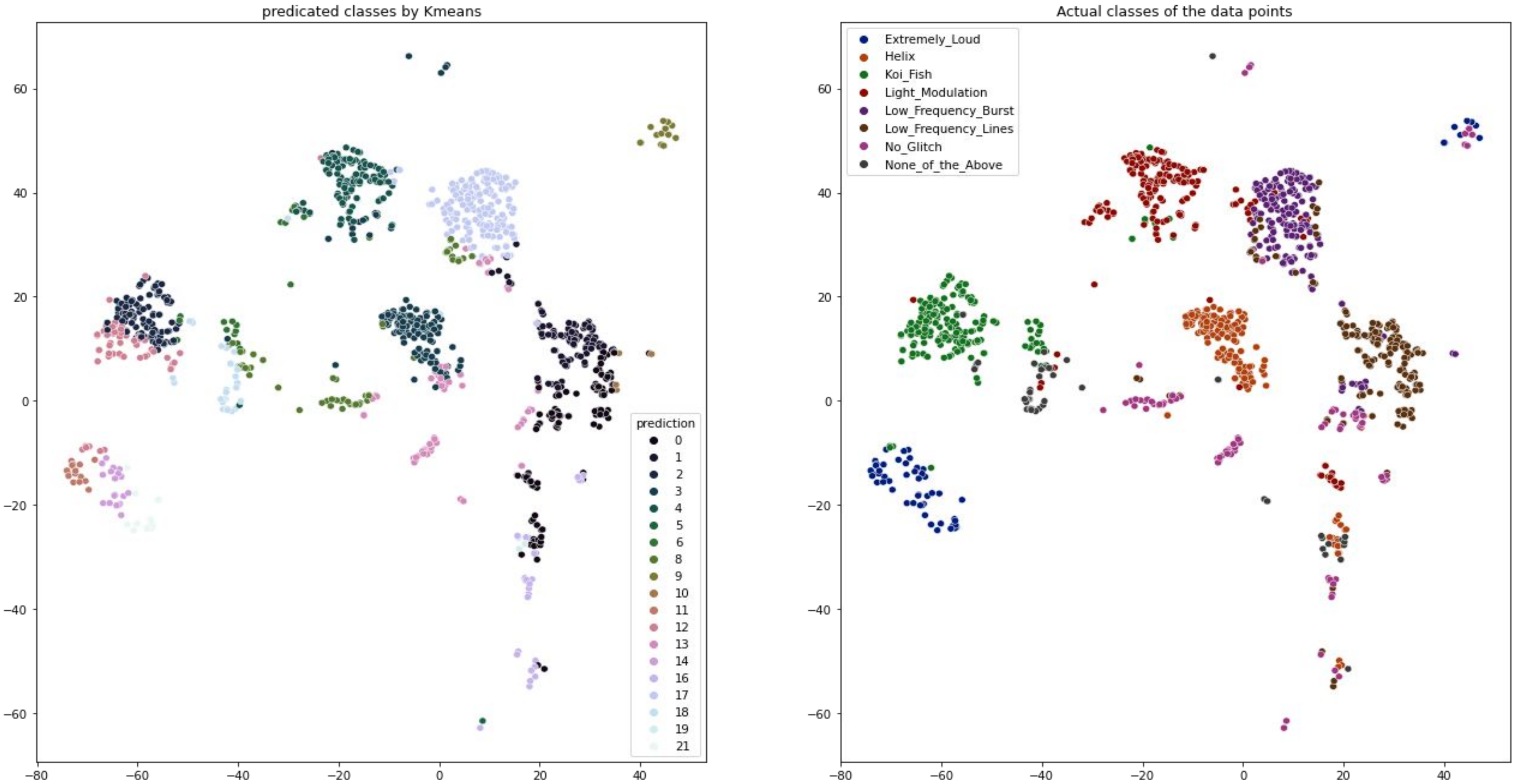
Comparison visualization in 2D



Visualization with t-SNE

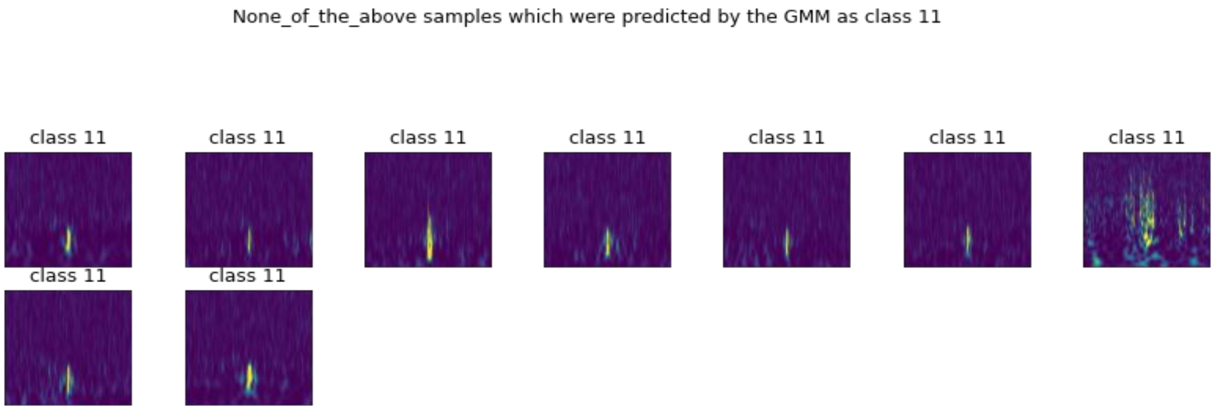


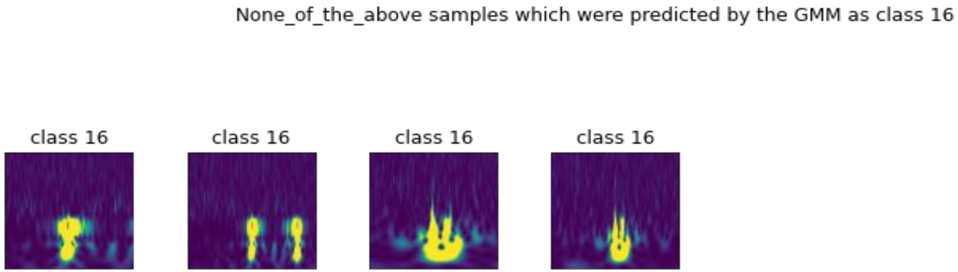
Visualization with t-SNE, a subsample of data.



Clustering : practice evaluation

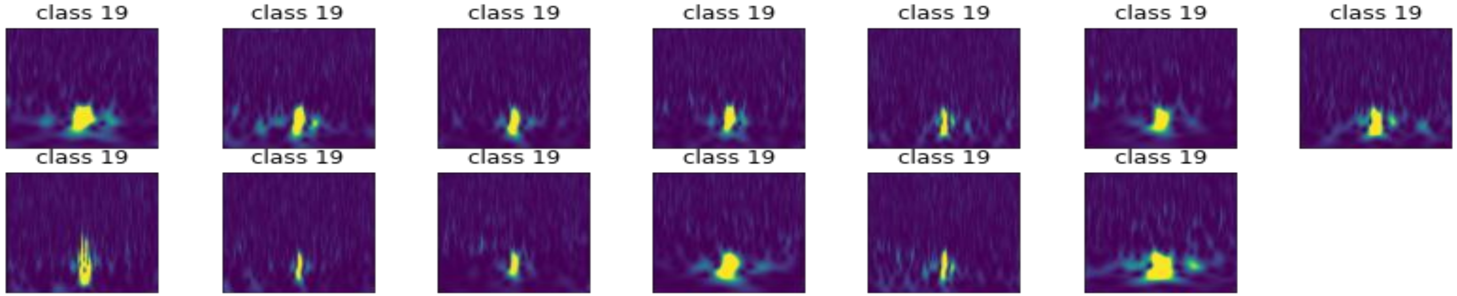
- If we have some unknown glitches ... how to deal?



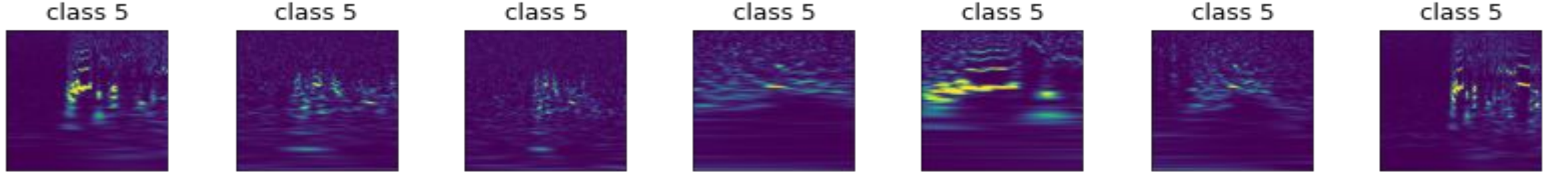


Note : here we are not interested in the labels but rather in the clustering itself.

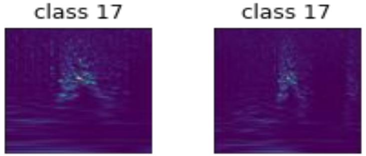
None_of_the_above samples which were predicted by the GMM as class 19



None_of_the_above samples which were predicted by the GMM as class 5

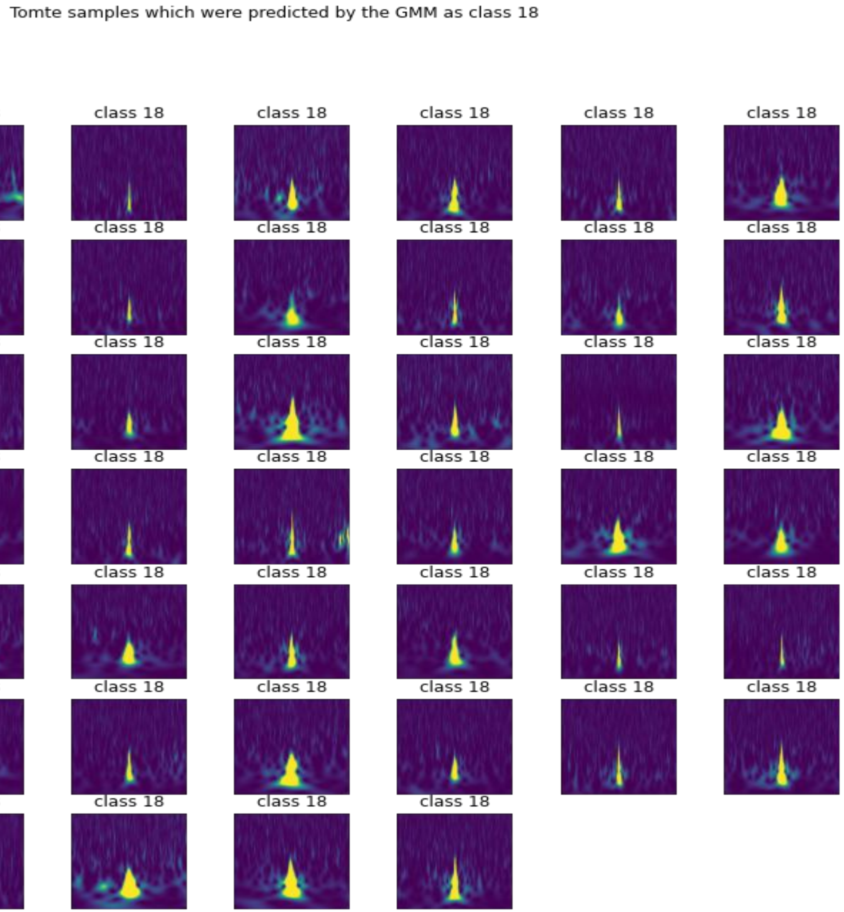


None_of_the_above samples which were predicted by the GMM as class 17



None_of_the_above samples which were predicted by the GMM as class 0

The previous are unseen data.. if we have a single known class ?



Only 3 cluster by GMM

Clustering is actually effective by using DNN as a feature extractor.

Conclusion :

In this project :

- I tried to reproduce the result that has been obtained by [2].

- I showed how EfficientNetB0 reach a well accuracy on Gravity Spy test set even though with a small computational power. The performance is comparable with the results obtained in [2].

- I've shown how transfer learning can actually be used with clustering to recognize new glitches and support human beings in the classification. Again confirmed [2].

- I expose how visualization can be used by humans in the clustering step in the gravity spy pipeline using t-SNE algorithm.

[2] George, Daniel, Hongyu Shen, and E. A. Huerta. "Deep Transfer Learning: A new deep learning glitch classification method for advanced LIGO." *arXiv preprint arXiv:1706.07446* (2017).

Feature work :

- CNN has been overtaken by Transformers, how would these models perform on this task? I have not found any paper about Transformer applied on this application.

- There’re exist other method to reduce the dimensionality of the data like Deep AutoEncoder. There are very few works in literature about it.

- Robustness of those models

Thank you for your attention