**Twitterstream**

The feature extractor I created uses the twitter streaming to represent “the voice” of many people who express their opinions about some topic using this application. This feature extractor uses a Python code, along with the tweepy library to deal with the twitter API; a search term needs to be introduced in order to determine the topic of the tweets. Then, the number of tweets is used as the main variable. With the help of OSC protocol this information is send to WekInputHelper (using port 6448 as Input) in order to study the behavior of the streaming. Next, the information is send to Wekinator (using port 6449 as input) where a model was trained to respond to the number of tweets as follows: a Pure Data Patch which receives information from Wekinator (using output port 12000), reflects the behavior of the trained model representing with high frequency sounds tweets coming from a trendy search topic (Clinton, Trump, paz, plebiscito) and low frequency sounds when the number of tweets is not that meaningful for such a topic (ex: Shakira, Neymar, etc.).

Eventually, I would like to use this tool to make a more complete sonification project using the tweeter API and some random sound files using some Python libraries to do so. Also, I would like to study deeper the relationship between some twitter variables, like RT, followers, etc.

This system can be used to start learning about sonification using technology and social networks; also about API’s, Python, Twitter, Machine learning and what these new ways of communications can tell us about data and sound.

 How to compile it:

1. Use Python version 3.5.2 (https://www.python.org/downloads/)
2. Install tweepy Python library: (<https://github.com/tweepy/tweepy>),(<http://tweepy.readthedocs.io/en/v3.5.0/>).
3. Install Pure Data Extended version: (https://puredata.info/downloads)
4. Open files *twitterstream.inputproj* (WekInputHelper)*, twitterstream.wekproj* (Wekinator*)* (Inside the twitterstream folder) *and* *pd\_oscv1.pd* (Pure Data Patch).
5. Run the Python file *stream.py*, introduce a search term and the streaming information will start to come up.
6. The Pure Data Patch will start playing a high or low frequency sound to represent the number of tweets per topic.