MACHINE LEARNING WITH JUCE

RAPIDMIX NEW JUCE MODULE

CLASSIFICATION VS REGRESSION

- **Classification** : the output is an integer variable or a class label
- Examples : animal recognition, who is talking in a recording, chord recognition…
- **Regression** : the output is a float/double variable or continuous in the general sense
- Examples : parameters of plug-ins, black box modeling, source separation, text to speech algorithm…

TRAINING / VALIDATING / PROCESSING

- **Training** : giving to a ML algorithm a **training set**, a collection of testing **samples**, made of an input vector and an output vector
- **Validating** : feeding the ML algorithm with some new data, and evaluating the result
- **Processing** : using the ML algorithm with any data, and getting more or less expected results

FEATURES EXTRACTION

- Raw data might not be the most relevant input
- **Features extraction** : compressing the raw data, or calculating something relevant from it, to feed our ML algorithm during training and getting the expected results
- **Examples of audio features** : RMS, highest peak, spectral centroid, pitch, any pre-processed audio signal (envelope followers, bandpass filters)...
- Example : voice recognition / classification

EXAMPLES OF INPUTS / CONTROLLERS / OUTPUTS

- **Inputs** : mouse cursor position, pixel values, audio sample amplitude, MIDI/OSC/MPE messages, random float variables

- **Outputs** : class label, mapping, transfer function, parameter of any processor, MIDI/OSC/MPE messages

- **Controllers** : mouse, keyboard, MIDI/OSC/MPE keyboard, joystick, Leap Motion, Blocks, Seaboard, smartphone + bluetooth, audio samples, webcam, computer microphone, LEDs, accelerometer, touch screen, Wiimote, Kinect…

RAPIDLIB API IN THE JUCE MODULE RAPIDMIX

- **Classification and Regression JUCE classes**
- initialize / train / process main functions
- load / save the state of the algorithm after training
- DataSample structure (different in Classification and Regression cases)
- Examples

HACKHATHON TIPS AND TRICKS

- UI as simple as possible (GenericPluginEditor or Projucer)
- Preparation is important (features, processing, acquisition)
- **Live Training** vs **Offline Training** applications
- Desktop application vs audio plug-in
- Embedding the audio processing vs controlling something outside
- RMS value threshold / action to trigger the training

FINAL WORDS...

- Machine Learning for **fast prototyping**
- Solving complex problems with **simple algorithms**
- Machine Learning for **innovative / unexpected / happy accident** solutions
- New RapidMix JUCE module : probably one of the **easiest to use** Machine Learning APIs
- Putting Machine Learning in **new hands** (music producers, plug-in developers, music artists)

HAPPY HACKHATHON !