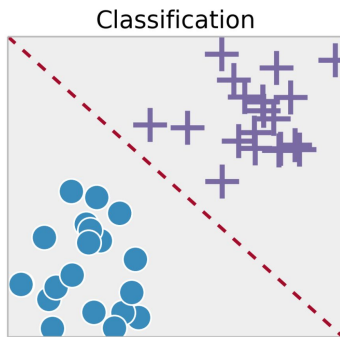


Wekinator

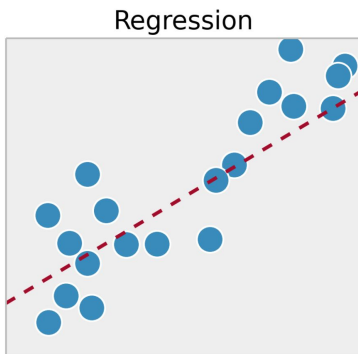
A tool for doing supervised machine learning tasks.

You can send numeric inputs to Wekinator from almost any type of sensor (webcam, microphone, leap motion, kinect, arduino) over OSC and have Wekinator crunch the numbers for either:

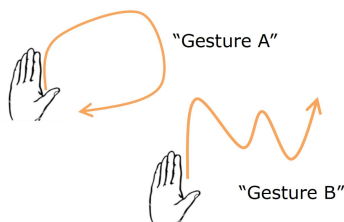
Classification: Classify inputs as belonging to class 1, 2, 3... etc.



Regression: Outputs continuous values (0.0 – 1.0). Good for graphics and music!



Dynamic Time Warping: Recognizes gestures over time.



Inputs and outputs:

ml4ixd_weki: Inputs and outputs in Processing. Can easily be changed.

mac_only: Mac only inputs + outputs. Cannot directly be changed unless you build them from scratch using openFrameworks.

Weki input helper

Small program that can help you filter noisy signals or control how inputs are sent to Wekinator.

Change what to send:

Send min, max, average, standard deviation of a feature over an analysis window

Send velocity or acceleration of a feature

Change when to send:

Frequency

Only send when a trigger condition is satisfied

ML5js / Tensorflow.js

Code examples in JavaScript / P5js that can (roughly) do the same tasks as Wekinator.

Advantages: Distributable on the web. The future?

Disadvantages: More coding. Less building blocks available. No GUI.

RunwayML

A new and powerful tool that can run more advanced models compared to Wekinator:

Im2text: Describes images/webcam feeds in sentences.

Yolo: Detect everyday objects (persons, cars, smartphones etc) and their positions/bounding boxes from images/webcam feed.

OpenPose: Detect human body positions from images/webcam feeds

EyeGaze: Find the position of where a person is gazing at a screen

You cannot train the models that Runway uses. Instead you should try to come up with interesting ways of using the outputs from Runway (which you can send to Processing or P5.js).

Still in beta – can be buggy.