Music Machine Learning

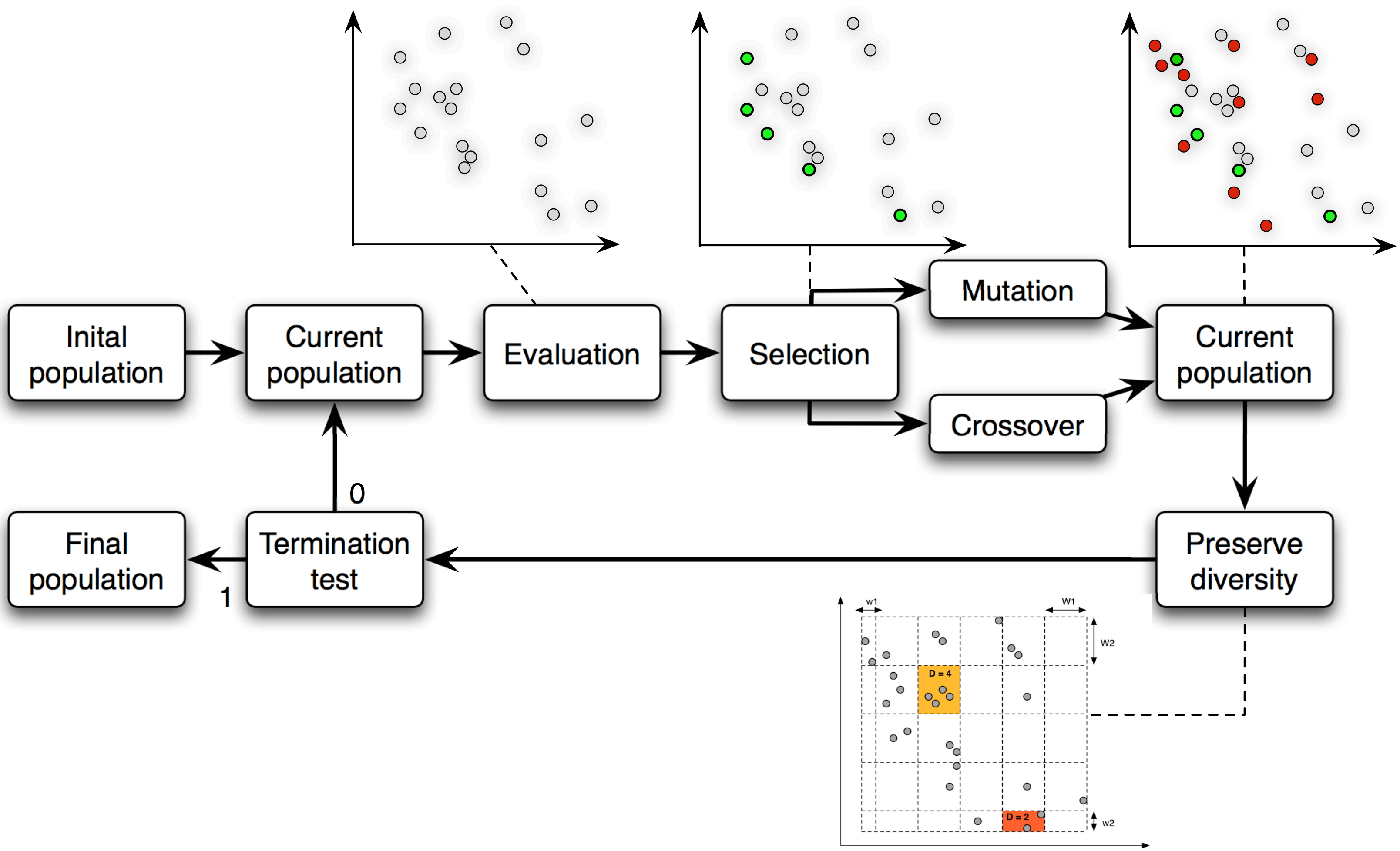
V – Boosting

Master ATIAM - Informatique

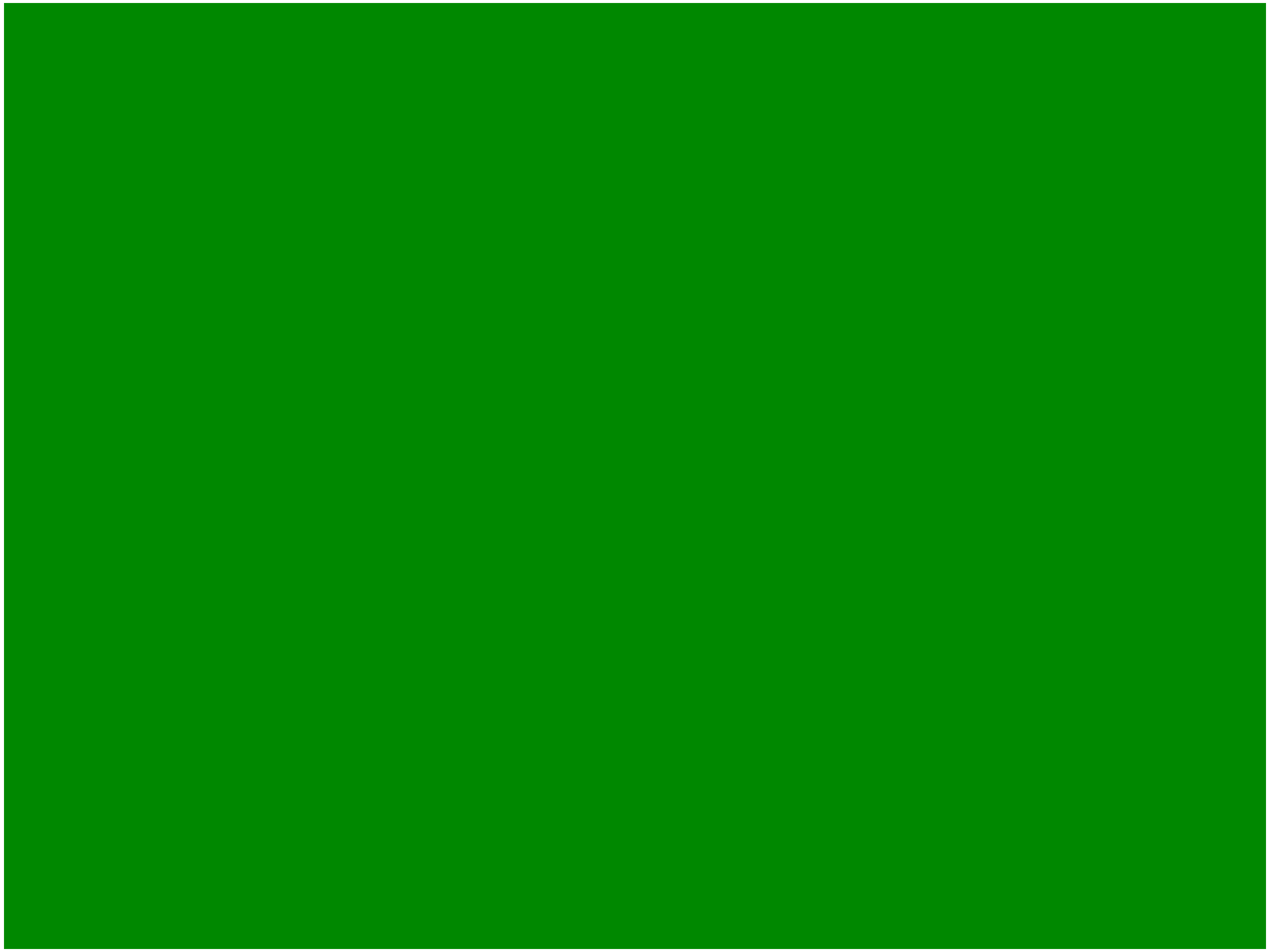
Philippe Esling (esling@ircam.fr)
Maître de conférences – UPMC
Equipe représentations musicales (IRCAM, Paris)

A few insights from evolution

Genetic algorithms



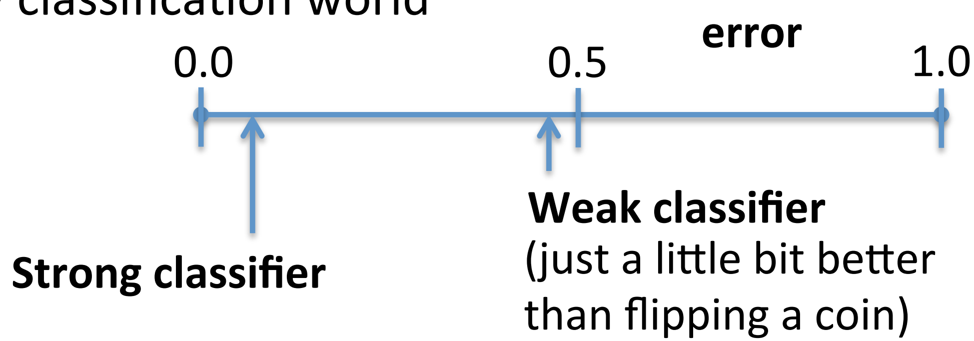
Genetic algorithms



Boosting

- So far we talked about *using a single method*
- Can you let multiple methods work on your behalf
- Is the crowd more intelligent than the individual participants?
- The whole thing is still in the framework of binary classification
- Am I holding a piece of chalk or a hand grenade ?
- So we live in a binary classification world

$$h = [-1, +1]$$



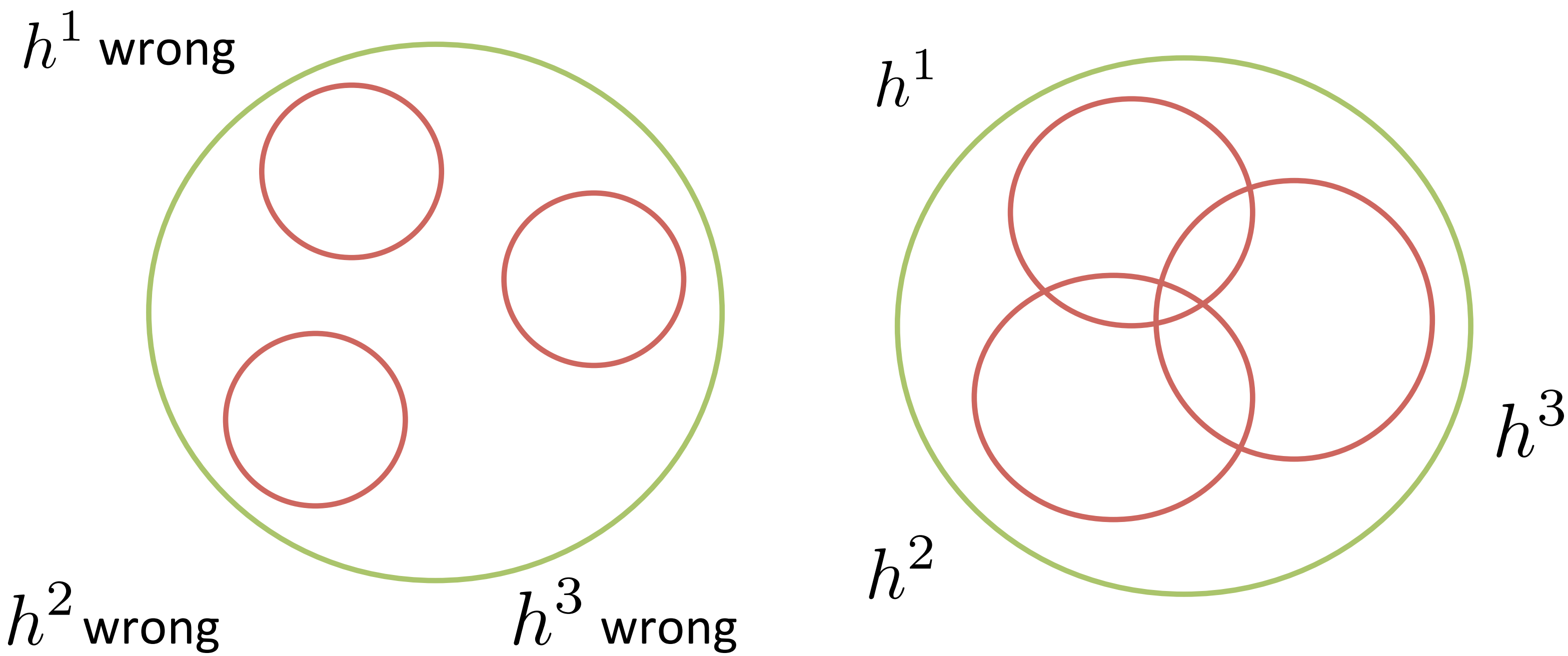
- The question is « can we build a strong classifier from several weak …
- … and let them vote to decide »
- On the theoretical surface, it seems extremely complicated
- Easy to implement and extremely efficient

Boosting

- Let's say we construct a classifier that works on a sample

$$H(x) = sign(h^1(x) + h^2(x) + h^3(x))$$

- We can see that if at least two agree, we will have a classification
- Hence, even if one is wrong, everything is alright as two agrees

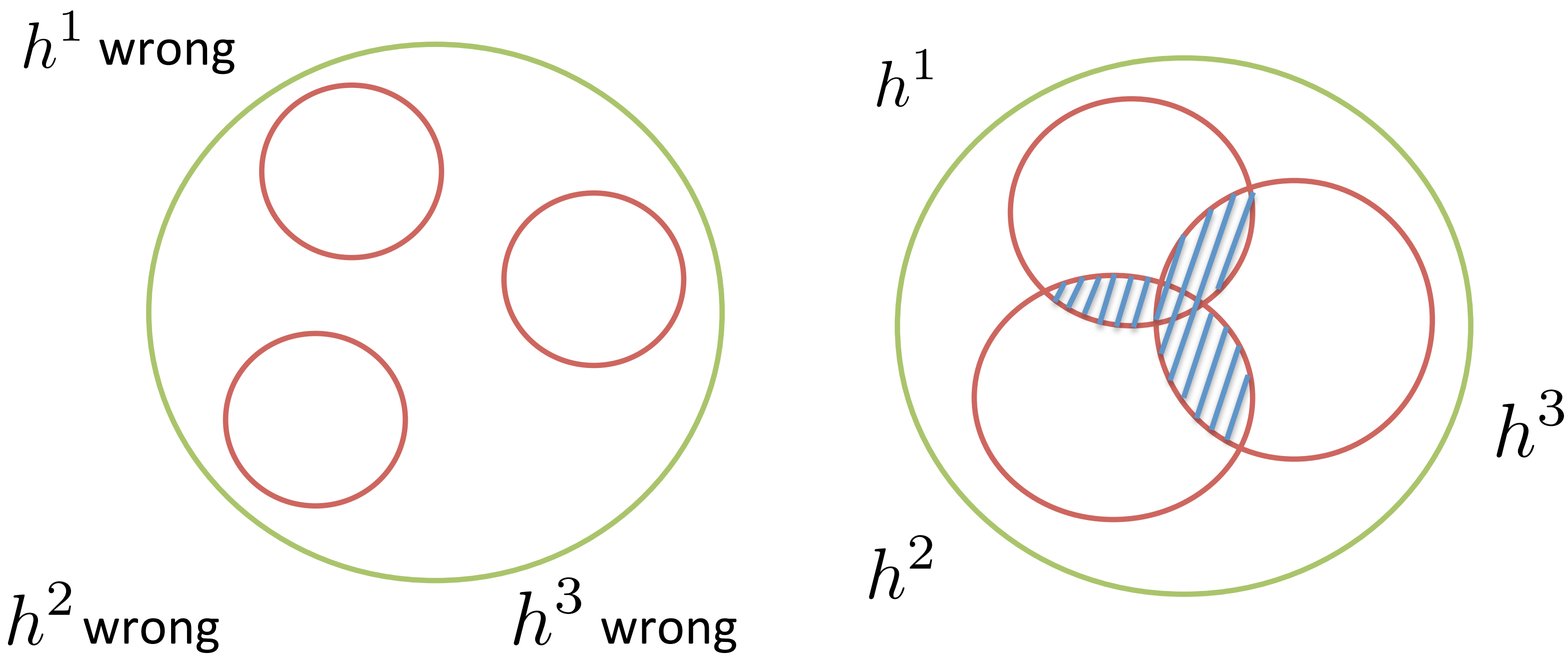


Boosting

- Let's say we construct a classifier that works on a sample

$$H(x) = sign(h^1(x) + h^2(x) + h^3(x))$$

- We can see that if at least two agree, we will have a classification
- Hence, even if one is wrong, everything is alright as two agrees



Boosting

1

Data $\longrightarrow h^1$

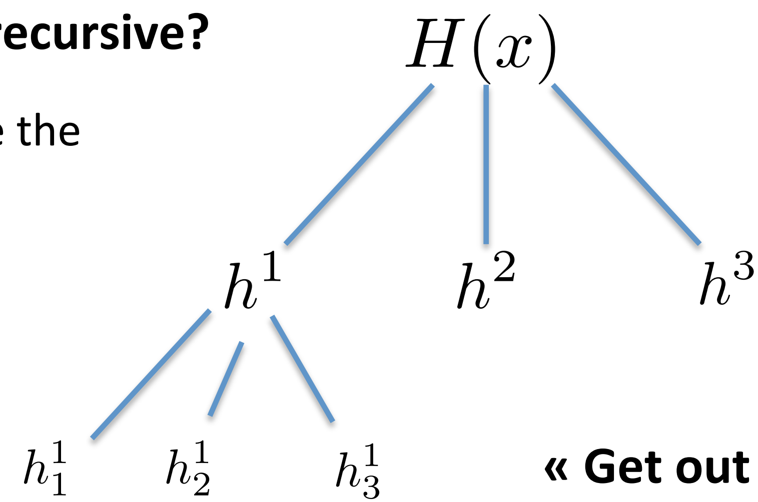
Data exageration of h^1 errors $\longrightarrow h^2$

Data exageration of h^2 errors $\longrightarrow h^3$

Can we make this recursive?

This might already be the result of a vote !

2

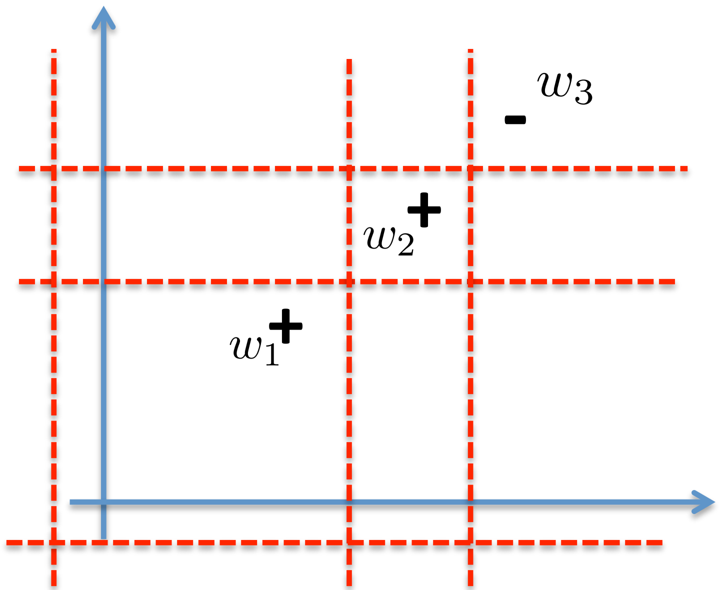


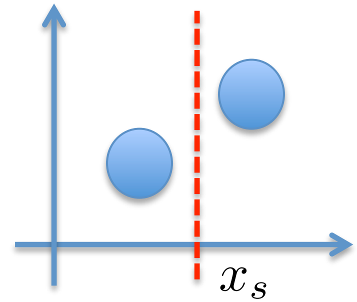
« Get out of the vote »

Boosting

3

Decision tree stumps





$$x > x_s$$

$$\epsilon = \sum_{wrong} \frac{1}{N} \qquad w_i^1 = \frac{1}{N}$$

$$\epsilon^n = \sum_{wrong} w_i^n \qquad \sum_i w_i^n = 1$$

We enforce a **distribution**

- So if we get the weights moving along classification
- We **exagerate** the impact of a specific point on the total error

Boosting

- We know that we should get lots of classifiers into the act
- But how many do we need to get an (almost) perfect classification

$$H(x) = sign(\underbrace{h^1(x)} + h^2(x) + h^3(x) + \ldots)$$

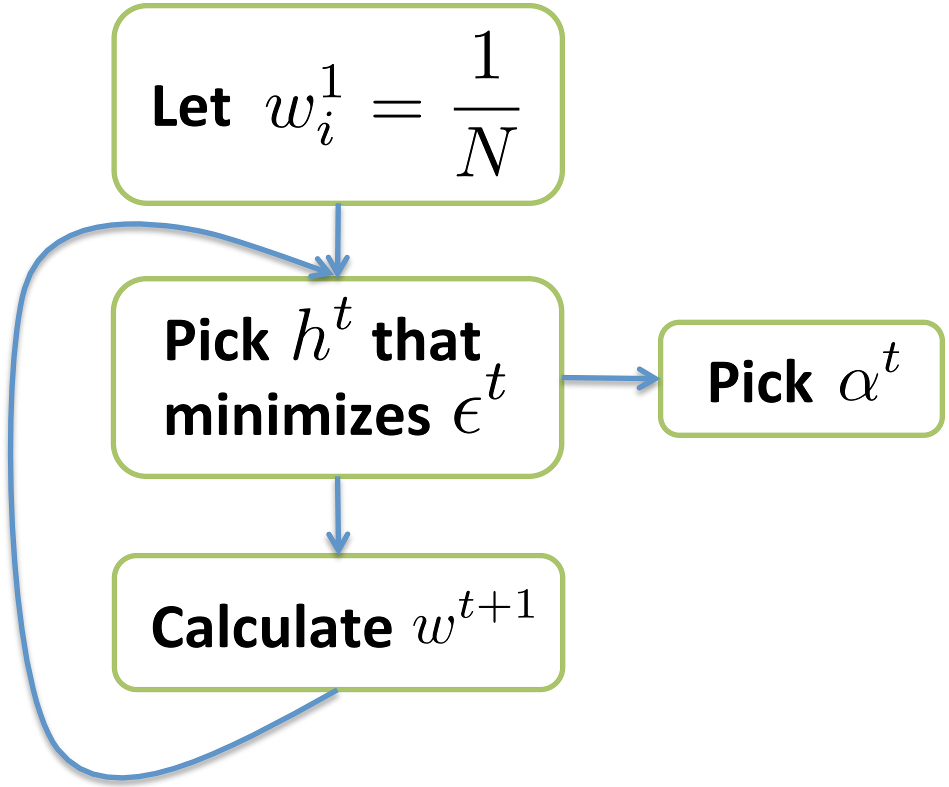
- As we have seen we construct this incrementally.
- But this almost looks like a scoring polynomial … So ?

4

$$H(x) = sign(\alpha^1 . h^1(x) + \alpha^2 . h^2(x) + \alpha^3 . h^3(x) + \ldots)$$

- The wisdom of a **weighted** crowd **of experts**
- The idea is to develop an automatic algorithm to compute this
- … But how do we get these things to work together?

Boosting



Until we produce a « perfect » classification of all samples

Boosting

Suppose $w_i^{t+1} = \frac{w_i^t}{z} e^{-\alpha^t h^t(x) y(x)}$

Normalizer
(ensures that w sums to 1)

Function +1/-1
(depending on the output)
Cf. SVM

After deriving, you found a bound on the minimum error if

$$\alpha^i = \frac{1}{2} ln(\frac{1-\epsilon^t}{\epsilon^t})$$

- So the error rate is bounded by an **exponentially decaying function**
- Guarantees to converge towards 0