# Using Raw Accuracy to Estimate Classifier Fitness in XCS

Pier Luca Lanzi

Artificial Intelligence and Robotics Laboratory
Dipartimento di Elettronica e Informazione
Politecnico di Milano
pierluca.lanzi@polimi.it

In XCS classifier fitness is based on the *relative accuracy* of the classifier prediction [3]. A classifier is more fit if its prediction of the expected payoff is *more accurate* than the prediction given by the other classifiers that are applied in the same situations. The use of *relative accuracy* has two major implications. First, because the evaluation of fitness is based on the relevance that classifiers have in some situations, classifiers that are the only ones applying in a certain situation have a high fitness, even if they are inaccurate. As a consequence, inaccurate classifiers might be able to reproduce so to cause reduced performance; as already noted by Wilson (personal communication reported in [1]). In addition, because the computation of classifier fitness is based both (i) on the classifier accuracy and (ii) on the classifier relevance in situations in which it applies, in XCS, classifier fitness does not provide information about the problem solution, but rather an indication of the classifier relevance in the encountered situations. Accordingly, it is not generally possible to tell whether a classifier with a high fitness is accurate or not, just looking at the fitness. To have this kind of information, we need the prediction error $\varepsilon$ which provides an indication of the *raw* classifier accuracy.

**Relative Accuracy.** In XCS, a classifier *cl* consists of a condition, an action, and four main parameters: (i) the prediction $cl.p$, which estimates the payoff that the system expects when *cl* is applied; (ii) the prediction error $cl.\varepsilon$, which estimates the error of the prediction $cl.p$; (iii) the fitness $cl.F$, which estimates the accuracy of the payoff prediction given by $cl.p$; and finally (iv) the numerosity $cl.num$, which indicates how many copies of classifiers with the same condition and the same action are present in the population. In XCS the update of classifier fitness consists of three steps. First, for all the classifiers *cl* in [A], the *raw accuracy* $cl.\kappa$ is computed: $cl.\kappa$ is 1 if $cl.\varepsilon \leq \varepsilon_0$; $\alpha(cl.\varepsilon/\varepsilon_0)^{-\nu}$ otherwise; $\alpha$ is usually 0.1; $\nu$ is usually 5. A classifier is considered to be *accurate* if its prediction error $cl.\varepsilon$ is smaller than the threshold $\varepsilon_0$; a classifier that is accurate has a *raw accuracy* $cl.\kappa$ equal to one. A classifier is considered to be *inaccurate* if its prediction error $\varepsilon$ is larger than $\varepsilon_0$; the *raw accuracy* $cl.\kappa$ of an inaccurate classifier *cl* is computed as a potential descending slope given by $\alpha(cl.\varepsilon/\varepsilon_0)^{-\nu}$. The classifier *raw accuracy* $cl.\kappa$ is used to calculate the *relative accuracy* $\kappa'$ as $(cl.\kappa \times cl.num)/\sum_{cl_i \in [A]}(cl_i.\kappa \times cl_i.num)$. Finally the *relative accuracy* $\kappa'$ is used to update the classifier fitness as: $cl.F \leftarrow cl.F + \beta(cl.\kappa' - cl.F)$.

E. Cantú-Paz et al. (Eds.): GECCO 2003, LNCS 2724, pp. 1922–1923, 2003.

**Raw Accuracy.** The idea behind *raw accuracy* is quite simple: instead of updating classifier fitness *cl.F* with the *relative accuracy* $cl.\kappa'$, we update *cl.F* with the *raw fitness* $cl.\kappa$. The fitness update becomes: $cl.F \leftarrow cl.F + \beta(cl.\kappa - cl.F)$. Classifier fitness now conveys different information since *raw accuracy* does not provide any knowledge about the classifier relative accuracy in the encountered situations. With *relative accuracy* this information is obtained through the use of classifier numerosity, and it is *intrinsically* exploited in Wilson's XCS when fitness is used, i.e.: (i) when the system prediction is computed; (ii) when offspring classifiers are selected from [A] with probability proportional to their fitness (see [3] for details). With *raw accuracy*, this information can be obtained by combining *raw accuracy* and *numerosity* either during the computation of the system prediction, either during the selection of offspring classifiers. This gives raise to four different XCS versions, which differ in the way *raw accuracy* and *numerosity* are combined. The first model, XCS`nn`, implements the most basic approach possible: classifier fitness is updated using the raw accuracy $\kappa$, but classifier numerosity is never used. XCS`nn` lacks of any information regarding the classifier relevance in the environmental niche which is instead available with *relative accuracy*. The second model, XCS`nga`, extends XCS`nn` by introducing numerosity for selecting offspring classifiers: fitness is updated using $\kappa$, classifier numerosity is used during offspring selection, i.e., the probability of selecting a classifier *cl* is proportional to $cl.F \times cl.num$. The third model, XCS`ne`, uses numerosity both for computing system prediction both for offspring selection. The fourth model, XCS`nu`, does not update fitness; *raw accuracy* $\kappa$ is used as the measure of fitness; this is equivalent to have $cl.F \leftarrow cl.\kappa$ so as to eliminate the parameter $F$; classifier fitness is thus computed directly from the prediction error.

**Discussion.** These different versions of XCS have been compared in [2] with Wilson's XCS [3]. Lanzi [2] shows that in simple single-step problems, XCS`nu` learns faster than all the other XCS versions producing also the most compact solutions (see [2] for details). In more complex problems, Wilson's XCS learns faster than XCS`nu` and XCS`ne` but it produces larger populations. Overall, the results reported in [2] suggest that *raw accuracy* might result in interesting performance, although Wilson's relative accuracy appears to provide the best trade-off.

## References


1. Pier Luca Lanzi. An Analysis of Generalization in the XCS Classifier System. *Evolutionary Computation*, 7(2):125–149, 1999.
2. Pier Luca Lanzi. A comparison of relative accuracy and raw accuracy in XCS. Technical Report 2003.14, Dipartimento di Elettronica e Informazione. Politecnico di Milano., March 2003.
3. Stewart W. Wilson. Classifier Fitness Based on Accuracy. *Evolutionary Computation*, 3(2):149–175, 1995. http://prediction-dynamics.com/.