Confusion Matrices for Improving Performance of Feature Pattern Classifier Systems

Ignas Kukenys
Victoria University of Wellington
ignas@cs.otago.ac.nz

Will N. Browne
Victoria University of Wellington
will.browne@vuw.ac.nz

Mengjie Zhang
Victoria University of Wellington
mengjie.zhang@ecs.vuw.ac.nz

ABSTRACT

Learning Classifier Systems (LCS) have not been widely applied to image recognition tasks due to the very large search space of pixel data. Assimilating the image domain's Haar-like features into the XCS framework, the feature pattern classifier system (FPCS) has produced promising results in the numeral recognition task. However for large multi-class image classification problems the training rates can be unacceptably slow, whilst performance does not match supervised learning approaches. This is partially due to the fact that traditional LCS only retain limited information about the problem examples. Confusion Matrices show the classes that a learning technique has difficulty separating, but require supervised knowledge. This paper shows that the knowledge in a confusion matrix is beneficial in directing learning. Most importantly the work shows that confusion matrices can be beneficially adapted to non-supervisory learning.

Categories and Subject Descriptors

F.1.1 [**Models of Computation**]: Genetics-Based Machine Learning, Learning Classifier Systems

General Terms

Algorithms, Performance

Keywords

Learning Classifier Systems, XCS, Haar-like features

1. INTRODUCTION

Learning Classifier Systems (LCS) can be applied to model online learning scenarios. By combining reinforcement learning with evolutionary computation to enable construction of a population of rules they can successfully learn to operate within unknown environments. Online, robotic agents often rely upon visual input, due to the richness of the information available. The work presented in this paper uses LCS to successfully learn in the image domain by replacing previous encodings with sparse Haar-like features [2].

The underlying hypothesis is that traditional LCS approaches discard much of the information that they encounter during the learning process. This results in unacceptably long training times to reach convergence on large problems. The new method, feature pattern classifier system (FPCS), is motivated by considering machine learning approaches and the cognitive science background of LCS. We attempt to improve the learning rate by introducing 'internal' learning mechanisms, which retain information encountered during training. This is demonstrated on the well known MNIST hand-written digit dataset [4].

GECCO'11, July 12–16, 2011, Dublin, Ireland.
ACM 978-1-4503-0690-4/11/07.

2. FEATURE PATTERN CLASSIFIERS

Traditionally, a learning classifier system represents an agent enacting in an unknown environment via a set of sensors for input and a set of effectors for actions. XCS, a formulation of LCS that uses accuracy-based fitness to learn the problem by forming a complete mapping of states and actions to rewards [6], is used as the base system. For problems in the image domain common encodings, ternary or real-alphabet, are both computationally intractable and insufficiently flexible—it is not uncommon for two images to contain closely related states, but have no identical pixels (consider 256 shades of grey in greyscale images). One popular type of features that is used in state of the art image classification systems [5] is the Haar-like rectangular features.

The value of feature f at location $l = (x, y)$ and scale $u = (width, height)$, $f(s, l, u)$ can be computed with just a few lookup calls in the integral image II.

By thresholding the value of $f(s, l, u)$ between t_{low} and t_{high}, binary decision rules can be formed that detect the presence of desired level of contrast between neighbouring regions in the image. We thus propose the following conditions for use in the LCS decision rules:

$$c_i = c(f, l, u, t_{low}, t_{high}),$$
$$c_i(s) = \begin{cases} \text{true, if } t_{low} < f(s, l, u) < t_{high} \\ \text{false, otherwise} \end{cases}$$

Notice that the Haar-like features are weak, in the sense that a single feature is insufficient to describe a complex pattern. We therefore utilise a 'messy' encoding [3]: by allowing multiple feature conditions to be joined using a logical '*and*' operator, the resulting decision rule conditions are sufficiently complex to make learning feasible:

$$c(s) = c_1(s) \wedge \ldots \wedge c_m(s).$$

Confusion matrices are a useful tool in machine learning that enable analysis of the errors that the learning system is making. Table 1 shows part of the confusion matrix of

the proposed FPCS as recorded on an independent testing set after 4 000 000 generations (one generation is a single message instance). In order to improve the learning rate, the system can be guided towards the 'problematic' regions of the problem domain, for example to distinguish between examples in most often confused classes.

The first important observation is that the information present in the confusion matrix is not trivially available to the agent—for any given example/state, the agent does not know the correct action from the environment. In order to obtain such information, the agent stores (*state*, *action*, *reward*) triplets. It then deduces the correct action for a state when one of the actions has a higher reward than the others. We will refer to such functionality in the LCS as a *long term memory component*, a weak analogy to functional aspects of memory in cognitive systems. The long term memory has to operate under the following assumptions:

- Some of the states in the environment may be encountered multiple times during the exploration.
- If at least two (*action*, *reward*) pairs are present in the memory for a given state, and one of the rewards is higher, the corresponding action can be treated as *established truth* for that particular state.

If established truth (high reward for *action*) is available for any given state, then record (*action*, *established_truth*) in the confusion matrix component. If subsequent incorrect action (confusion), then focus training on mistaken actions.

3. EXPERIMENTAL RESULTS

The necessary adjustments to a standard XCS [1] include mutation probability $\mu = 0.4$ (significantly higher than typical XCS applications) with tournament selection fraction $\tau = 0.4$ and up to 10 'Messy' features. Each experiment was repeated 30 times and the result recorded on the separate testing set of 10 000 MNIST examples.

Table 1: Confusion matrix for the independent evaluation set (mean performance±standard deviation, %). Rows correspond to system classification, columns correspond to actual classes (E estimated, A actual).

E\A	0	1	2	...	9
0	**98±0**	0±0	1±0	...	1±0
1	0±0	**99±0**	0±0	...	1±0
2	0±0	0±0	**94±1**	...	0±0
...	...	...	...	...	...
9	0±0	0±0	1±0	...	**89±2**

Figure 1 confirms that the memory component was useful as at least some of the states were encountered multiple time so that the confusion matrix could record its best experiences. Performance improvements occur prior to the total number of instances in the data being experienced as it was coupled with the generalisation of the LCS. The 'true' (rather than deduced) CM was used in a divide and conquer approach of separate training sets to determine the benchmark, but unprincipled, performance.

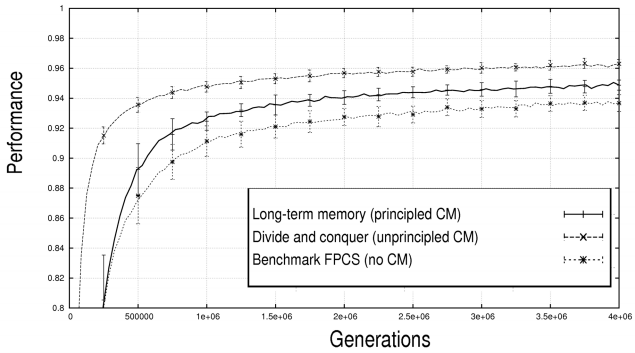

Figure 1: Performance (mean with standard deviation bars) of the different methods. The long-term memory FPCS approaches the 'ideal' divide and conquer system.

4. CONCLUSIONS

The concept of confusion matrix has been adapted to create a novel online mechanism for LCS. This is shown to have significant benefit for LCS performance over the same number of problem instances, without breaking the principles of LCS, i.e. online learning without prior knowledge. It is necessary to assume that there will be a degree of repetition in the environment.

5. REFERENCES

[1] M. V. Butz. *Rule-based evolutionary online learning systems: A principled approach to LCS analysis and design*. Springer Verlag, Berlin Heidelberg, 2006.

[2] I. Kukenys, W. N. Browne, and M. Zhang. Transparent, Online Image Pattern Classification Using a Learning Classifier System. In *European Conference on the Applications of Evolutionary Computation*, 27-29 April 2011.

[3] P. L. Lanzi and A. Perrucci. Extending the representation of classifier conditions part ii: From messy coding to s-expressions. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 345–352, 1999.

[4] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based Learning Applied to Document Recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[5] P. Viola and M. Jones. Rapid Object Detection Using a Boosted Cascade of Simple Features. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, 2001.

[6] S. Wilson. Classifier Fitness Based on Accuracy. *Evolutionary Computation*, 3(2):149–175, 1995.