There is No Free Lunch but the Starter is Cheap: Generalisation from First Principles

Chris Thornton

Cognitive and Computing Sciences

University of Sussex

Brighton

BN1 9QH

UK

Email: Chris.Thornton@cogs.susx.ac.uk

WWW: http://www.cogs.susx.ac.uk/users/cjt

Tel: (44)1273 678856

January 18, 1999

Abstract

According to Wolpert's no-free-lunch (NFL) theorems [Wolpert, 1996b, Wolpert, 1996a], generalisation in the absence of domain knowledge is necessarily a zero-sum enterprise. Good generalisation performance in one situation is always offset by bad performance in another. Wolpert notes that the theorems do *not* demonstrate that effective generalisation is a logical impossibility but merely that a learner's bias (or assumption set) is of key importance in determining its generalisation performance. However, in this paper it is argued that this may be an over-reading of the results. Situations can be identified in which a learner's assumptions are effectively *guaranteed* correct. The in-practice prevalence of these situations may account for the reliably good generalisation performance of methods such as C4.5 and Backpropagation.

Keywords: no-free-lunch, generalisation, learning complexity

1 Introduction

There has been lively controversy over Wolpert's no-free-lunch theorems [Wolpert, 1996b; Wolpert, 1996a; Wolpert, 1995b; Wolpert, 1992; Wolpert, 1995a; Wolpert and Macready, 1995] and Schaffer's closely related **conservation law** [Schaffer, 1994]. These results show that there is no guaranteed correct way of performing

generalisation. They thus affirm Hume's claim to the effect that the observation of 'the frequent conjunction of objects' does not permit the drawing of any particular inference concerning 'any object beyond those of which we have had experience' [Hume, 1740].

The underlying idea behind these results is easily stated. Let's say we have a particular learning method and we would like to know how well it will generalise on the problems from a specific domain. If we have no special knowledge about the domain then all problems in the domain have to be considered uniformly likely, i.e,. the problems in the domain have to be considered to follow a uniform distribution. In this context, the problems in the domain may be organised into 'opposites', such that the way the unseen (test) cases are classified in a particular problem is the reverse of the way they are classified in its opposite. A particular learning algorithm generalises cases in a specific way. Thus, if it performs slightly better than random guessing on a particular problem, it must perform slightly worse than random guessing on the problem's opposite. On a random selection of problems from the domain, a learning algorithm will therefore tend to produce above-chance performance on some problems and below-chance performance on other problems. Since the chances of it producing above-chance performance are *identical* to the chances of it producing below-chance performance, it will, on average, produce exactly the same performance as random guessing.

At first sight, the NFL result appears to demonstrate that effective (i.e., above-chance) generalisation is impossible in principle. But this is not the case. In the NFL scenario, we have the rather severe constraint that *nothing* is known about the domain. All problems have then to be considered equally likely and the process of applying a particular learner to some random selection of problems necessarily produces chance-level performance (on average). The explicit consequence of the NFL result is thus that in the situation where no domain assumptions can be made, chance-level performance is the inevitable result. But the subtext of the NFL work is that it is the assumptions a learner makes about its domain which are key.[1] As Michael Perrone has commented, 'What makes NFL important is that it emphasizes in a very striking way that it is the *assumptions* that we make about our learning domains that *make all the difference*.'[2]

However, interpreting the NFL theorems in this way raises a new concern. As Wolpert has noted, the biases of empirical generalisation methods are typically not made explicit and are only rarely justified in terms of the expected application domain. He notes that 'for many algorithms, no one has even tried to write down that set of [problems] for which their algorithm works well.' [Wolpert, 1996b]. However, it is clear that generalisation methods *are* capable of performing well in practice across a wide variety of situations [Thrun *et al.* 1991]. In

---

[1] Wolpert specifically mentions the requirement to prove that 'the non-uniformity in [the problem domain] is well-matched to your ... learning algorithm.' [Wolpert, 1996b, p. 19]

[2] From a posting to the 'connectionists' mail list.

latter operation depends entirely on the properties of the agent and should not, therefore, be considered a part of the generic complexity of the learning task. This should be estimated purely in terms of the identification operation.

Identifying the relevant indication involves identifying the connections that may exist between the learner's informational data and the actions in question. The complexity of this depends on the number of possible connections and this, in turns, depends

the data available to the learner agent take the form of combinations of values
of variables — a very common scenario — and that each particular combination
of values is treated as an n-dimensional datapoint. If the task is relational, we
know that particular actions are contingent on relati6n00.7(v)20t.120373Td[(on)-1600a9.6g3(task)-4000.

pbinations ond t31000.99.7(e3tions)]TJ81.1(t31000should7(
alues isn-dime30b7(v)20-11997(−continueP(v)il0)eH0(a6ak)-40out7(a-dim09ex-u]lly3(task)-40a

```
c   d   a   b   -->   f
a   b   d   b   -->   h
e   c   d   e   -->   h
c   b   a   e   -->   f
a   c   d   e   -->   f
b   c   a   e   -->   f
b   d   d   e   -->   h
e   d   a   c   -->   f
a   c   d   c   -->   h
c   d   a   c   -->   h
c   c   a   e   -->
```

Figure 3: Hybrid learning task.

in (the data for) a characteristically relational problem, e.g.

- the task may have genuine, non-relational aspects and thus exhibit a degree of meaningful clustering. The 'greater-than' task is a good example.

- The task may be represented to the learner in such a way as to create artificial non-relational aspects. An example of this situation is a parity task whose representation includes an extra input variable whose value always effectively records the parity status of the original inputs is an example.

In both of these situations, the exhibited clustering is useful for the purposes of learning, i.e., it can be used as the basis for generalisation. There are two further situations, however, in which the clustering is of no use whatsoever.

- The clusters may be an artifact of the way in which the learner's data have been selected or generated.

- The clusters may be the results of some sort of noise or data error.

In both of these cases, the clusters observed in the data are merely sampling artifacts and thus of no use whatsoever within the learning process.

To summarise, in a characteristically relational task we may see clustering effects arising from non-relational aspects of the task, characteristics of the task encoding, characteristics of the data selection process or noise/error. Effects due to the task encoding, data selection or noise may be termed **incidental**, on the grounds that their relationship with the underlying problem is not meaningful. Within this grouping, effects due to characteristics of the task representation may be termed **generalising** while effects due to the data selection or noise may be termed **non-generalising**. The various possibilities are tabulated in Figure 4.

7

	Cluster origin	
	Non-relational aspect	*Generalising*
Incidental	Problem encoding	*Generalising*
Incidental	Exemplar selection	
Incidental	Noise/error	

Figure 4: Origins of clusters in characteristically relational problems.

3.1 Typical scenarios

Despite the difficulties noted, it remains the case that non-relationality *does* produce clustering while relationality does tend to eliminate it. The existence or lack of clustering therefore can serve as a guide for *tentative* classification decisions. We have seen that almost all learning tasks show a certain degree of clustering. But the more clustering they exhibit the stronger the evidence in favour of a non-relational classification. The range of possibilities is illustrated in Figure 5. Each task here is displayed as a 2-dimensional graph and is therefore assumed to be defined in terms of two, numeric data variables and one action variable, whose value is either '1' or '0'. The problems represent typical scenarios from the perfectly non-relational to the perfectly relational. In the 'perfect

```
1 0 1 0 1 0
1 0 1 0 1 0 1
0 1 0 1 0 1 0
0     1 0 1 0
0 1 0 1 0 1 0
```

of the space.

shown in Figure 8-1 can be interpreted as defining a modulus-addition function

- unlearnable.

A task falls into one of the three learnable categories if the relational (non-

4.2 Geometric separability of frequently used datasets

We can demonstrate the non-relationality of typical learning problems empirically. We have already noted that the solving of a non-relationally-learnable

in a randomly selected testing set. Thus, on average, the 1-nearest-neighbour

relational effects (i.e., in terms of data similarity[7] or data clustering) will tend to perform well across the board. The almost universally good (i.e., above chance) performance of methods such as C4.5 and Backpropagation may be explained in these terms.

References

[1] Breiman, L., Friedman, J., Olshen, R. and Stone, C. (1984). *Classification and Regression Trees*. Wadsworth.

[2] Clark, A. and Thornton, C. (1997). Trading spaces: computation, representation and the limits of uninformed learning. *Behaviour and Brain Sciences*, *20* (pp. 57-90). Cambridge University Press.

[3] Diday, E. and Simon, J. (1980). Clustering analysis. In K. Fu (Ed.), *Digital Pattern Recognition*. Communications and Cybernetics, No. 10 (pp. 47-92). Berlin: Springer-Verlag.

[4] Dietterich, T. and Michalski, R. (1983). A comparative review of selected methods for learning from examples. In R. Michalski, J. Carbonell and T. Mitchell (Eds.), *Machine Learning: An Artificial Intelligence Approach*. Palo Alto: Tioga.

[5] Holte, R. (1993). Very simple classification rules perform well on most commonly used datasets. *Machine learning*, *3* (pp. 63-91).

[6] Hume, D. (1740). *A Treatise of Human Nature* (second edition). Oxford University Press.

[7] Kohonen, T. (1984). *Self-organization and Associative Memory*. Berlin: Springer-Verlag.

[8] Langley, P. (1977). Rediscovering physics with bacon-3. *Proceedings of the Fifth International Joint Conference on Artificial Intelligence: Vol I*.

[9] Langley, P. (1978). BACON.1: a general discovery system. *Proceedings of the Second National Conference of the Canadian Society for Computational Studies in Intelligence* (pp. 173-180). Toronto.

---

[7]Learning methods which may be classified as similarity-based include the CART algorithms [Breiman *et al.* 1984], the **competitive learning** regime of Rumelhart and Zipser [1986], the **Kohonen net** [Kohonen, 1984] and in fact any algorithmic method which is based on the method of **clustering** [Diday and Simon, 1980]. Methods which are clearly excluded from this class include the 'BACON' methods of Langley and co-workers [Langley, 1977; Langley, 1978; Langley *et al.* 1983; Langley *et al.* 1987] and related methods such as [Wolff, 1978; Wolff, 1980; Lenat, 1982; Wnek and Michalski, 1994]. These carry out explicit searches for relational effects and in many cases ignore similarity effects altogether.

[10] Langley, P., Bradshaw, G. and Simon, H. (1983). Rediscovering chemistry with the BACON system. In R. Michalski, J. Carbonell and T. Mitchell (Eds.), *Machine Learning: An Artificial Intelligence Approach* (pp. 307-329). Palo Alto: Tioga.

[11] Langley, P., Simon, H., Bradshaw, G. and Zytkow, J. (1987). *Scientific Discovery: Computational Explorations of the Creative Processes*. Cambridge, Mass.: MIT Press.

[12] Lenat, D. (1982). AM: discovery in mathematics as heuristic search. In R. Davis and D.B. Lenat (Eds.), *Knowledge-Based Systems in Artificial Intelligence* (pp. 1-225). New York: McGraw-Hill.

[13] Minsky, M. and Papert, S. (1988). *Perceptrons: An Introduction to Computational Geometry* (expanded edn). Cambridge, Mass.: MIT Press.

[14] Mitchell, T. (1997). *Machine Learning*. McGraw-Hill.

[15] Muggleton, S. (Ed.) (1992). *Inductive Logic Programming*. Academic Press.

[16] Rumelhart, D. and Zipser, D. (1986). Feature discovery by competitive learning. In D. Rumelhart, J. McClelland and the PDP Research Group (Eds.), *Parallel Distributed Processing: Explorations in the Microstructures of Cognition. Vol I* (pp. 151-193). Cambridge, Mass.: MIT Press.

[17] Schaffer, S. (1994). Making up discovery. In M.A. Boden (Ed.), *Dimensions of Creativity* (pp. 13-52). MIT Press.

[18] Thornton, C. (1997). Separability is a learner's best friend. In J.A. Bullinaria, D.W. Glasspool and G. Houghton (Eds.), *Proceedings of the Fourth Neural Computation and Psychology Workshop: Connectionist Representations* (pp. 40-47). London: Springer-Verlag.

[19] Thornton, C. and Clark, A. (Forthcoming). Reading the generalizer's mind. *Behaviour and Brain Sciences*, Cambridge University Press.

[20] Thrun, S., Bala, J., Bloedorn, E., Bratko, I., Cestnik, B., Cheng, J., De Jong, K., Dzeroski, S., Fisher, D., Fahlman, S., Hamann, R., Kaufman, K., Keller, S., Kononenko, I., Kreuziger, J., Michalski, R., Mitchell, T., Pa-
(Bala,)Tj26.-7999.98(H.,)-1099n7c7(R(R.,)Tj1J)-1000.37.26(v)1000.57(eell,)]TJ42.48010T(R.,)Tj1Rd

0000

[22] Wolff, J. (1978). Grammar discovery as data compression. *Proceedings of the AISB/GI conference on Artificial Intelligence* (pp. 375-379). Hamburg.

[23] Wolff, J. (1980). Data compression, generalisation and overgeneralisation in an evolving theory of language development. *Proceedings of the AISB-80 conference on Artificial Intelligence*. Amsterdam.

[24] Wolpert, D. (1992). On the connection between in-sample testing and generalization error. *Complex Systems,*