Uses and Abuses and Misuses of Computational Models in Classical Conditioning

Eduardo Alonso (E.Alonso@city.ac.uk)
School of Informatics, City University London
London EC1 0HB, UK

Esther Mondragón (e.mondragon@ucl.ac.uk)
Division of Psychology and Language Sciences, University College London
London WC1H 0AP, UK

Abstract

Classical conditioning is at the heart of most learning phenomena. It is thus essential that we develop accurate models of conditioning; since, psychological models rely heavily in verbal accounts that are necessarily imprecise it has become apparent that the development of computational models is imperative. However, we need to separate the wheat from the chaff. In this paper we review the main uses of the term computational model in conditioning.

Keywords: Classical conditioning; computational models; psychological models; simulation.

Introduction

It is universally accepted that conditioning is at the basis of most learning phenomena and behavior: Indeed, models of associative learning have proved to be relevant to human and non-human learning both theoretically and in practice (Wasserman & Miller, 1997; Pearce & Bouton, 2001; Hall, 2002; Schachtman & Reilly, 2011). It is thus paramount that we develop accurate models of conditioning. In this enterprise, collaboration between computer scientists and psychologists has enjoyed considerable success (Schmajuk, 2010a; Alonso & Mondragón, 2011): Connectionist models have been used to better predict discrimination and categorization phenomena (Shanks, 1995). In addition it has been argued that classical conditioning rules can be naturally interpreted as an instance of more comprehensive computational neuroscience models (Dayan & Abbott, 2001; Schmajuk, 2010b).

This collaboration is sustained on various arguments: Expressing models in the form of algorithms provides us with formal ways of representing psychological insights and of calculating their predictions accurately and quickly; from computational models we also borrow a view on how information is processed, a computer analogy that has proved useful in understanding cognition; moreover, the underlying architectures of computational models, for instance the hidden units of an artificial neural network or the way feedback is computed in recurrent networks, resemble the mechanics of associative learning both at a conceptual and neural level; finally, machine learning models, such as temporal difference learning and Bayesian learning, can be understood as effective abstractions of the way associations are formed and computed.

In this paper we analyse critically the assumptions upon which such arguments are built. We identify two main trends in so-called computational psychology, more in particular in the use of computational models in the study of conditioning, namely, as simulators and as psychological models in themselves.

Computational Models as Simulators

Firstly, a computational model can be understood to be an implementation of a (pre-existing) psychological model. Simulations serve two main purposes: On the one hand, implementing a model requires precise definitions – be it in the form of a specific programming language or as a formal model, that in turn makes the original psychological model “accountable”. On the other hand, algorithms allow us to execute calculations rapidly and, most importantly, accurately. Automation is critical, particularly when the models are described in non-linear equations that can only be solved numerically as it is the case of recent psychological models of conditioning (Balkenius & Morén, 1998; Vogel et al., 2004; Mitchell & Le Pelley, 2010; Schmajuk & Alonso, 2012). Crucially, the outputs of a simulation feedback the psychological models – thus becoming an essential part of the cycle of theory formation and refinement.

It is worth noting though that the benefits derived from using implementations do not spring exclusively from the formal specification of the psychological models in equations and algorithms. Per se, such descriptions constitute a mathematical model, a necessary yet no sufficient condition for a formal model to be computational. The essence of a computational model lies in the fact that it is implemented. According to this view, in psychology, the same as in computational physics and in computational biology, a computational model is a model that has been simulated.

This view it is not without detractors: It has been argued that a model is computational if it is “implementable” – even if it was not originally described as a full-bodied computational model. We think that this is an abuse of the term computational since any psychological model of conditioning would fit this definition. To use a parallelism: This use of the term “computational” would make all models of physics since Galileo’s computational.

This brings up a subtler issue: We are using the term computational model in a “modern” sense. Indeed, a computational model is just a formal model of
computation and “computation” does not necessarily require its implementation in a computer. Mathematically, the notion of computation is a formalization of the concept of algorithm, a mechanical or automated procedure to prove theorems proposed by Alan Turing to attack Hilbert’s Entscheidungsproblem (Turing, 1937). Modern computers are mere physical instantiations of the abstract machines that would compute such procedures. But they don’t play a fundamental part in the definition of computation. Indeed, such definition was proposed well before the first digital general-purpose computers had even been designed. Contrary to our position is that a computational model needs to be implemented in a computer. Otherwise, a computational model does not add anything to what constitutes a mathematical model in its own right.

We would also like to comment on a second potential source of confusion about the term computational—that comes from cognitive science rather than from mathematics. The term “computational” has been linked to David Marr’s Tri-Level Hypothesis on vision where the “what” refers to the computational level, the “how” to the algorithmic level and the “where” to the implementational level (Marr, 1982). However insightful such analysis may be, clearly what Marr referred to as “computational” is “psychological”—when applied to cognition. Insisting on talking about psychological models as if they were computational based on such taxonomy is, in our opinion, a source of misunderstanding.

Computational Models as Psychological Models
The second use of the term “computational” is more controversial: A computational model can be considered a psychological model in itself. We argue that this position, a milestone in cognitive science and artificial intelligence, is a missuse of the term. Let’s illustrate our contention using a paradigmatic example: The use of Artificial Neural Networks (ANNs) in the study of conditioning has been advocated at several, inter-related levels that we are now analyzing.

Ontological Level
ANNs are considered material models of conditioning. The underlying reasoning is that (a) ANNs model by analogy natural neural networks and that (b) psychological processes, including conditioning, are ultimately embedded in natural neural networks; hence, indirectly, ANNs model conditioning.

However appealing this line of argumentation may be, it is widely acknowledged that ANNs do not resemble natural neural networks in any fundamental way (Enquist & Ghirlanda, 2005); besides, there is no strong evidence suggesting that electrical or chemical neural activity and associative learning are related (Morris, 1994)—or for that matter, that psychological processes can be localized in specific brain regions as recently exposed in (Vul et al., 2009), but already advanced in (Uttal, 2001).

Even if it did, a neural analysis would not necessarily be the right level to study associative learning phenomena. In the words of Burruhrs F. Skinner “The analysis of behaviour need not wait until brain science has done its part. The behavioural facts will not be changed (...). Brain science may discover other kinds of variables affecting behaviour, but it will turn to a behavioural analysis for the clearest account of their effects” (Skinner, 1989, pp. 18). Regardless of the antipathy that Skinner’s radical behaviorism provokes among neuroscientists such an statement does not contradict a version of reductionism that most of them would endorse, namely, Richard Dawkin’s hierarchical reductionism (Dawkins, 1986).

Formal Level
Relatively, that a version of Dirac’s rule can be taken as a model of both neural plasticity and long-term potentiation effects—the Hebbian rule (Hebb, 1949)—and association formation—for example, Rescorla and Wagner’s rule (Rescorla & Wagner, 1972)—cannot be considered as proof of any common underlying structure and should not be used as an argument to reduce psychological phenomena to their alleged neural substratum.

Likewise, that Rescorla and Wagner’s rule is essentially identical to the Widrow-Hoff rule (Widrow & Hoff, 1960) for training Adeline units and that, in turn, such a rule can be seen as a primitive form of the generalized delta rule for backpropagation only tells us that, computationally speaking, associative learning follows an error-correction algorithm1. What a computational model does not tell us, however, is which underlying psychological processes (attention, motivation, etc.) intervene in associative learning or how the physical characteristics of the units involved (e.g., the salience of the stimuli) affect such processes.

Clearly, sharing a common formal expression does not imply that the phenomena so expressed are of the same nature: For instance, power functions can be used to express the relationship between (1) the magnitude of a stimulus and its perceived intensity (Stevens’ law), (2) the metabolic rate of a species and their body mass (Kleiber’s law), and (3) the orbital period of a planet and its orbital semi-major axis (Kepler’s third law). Stressing this point, allow us to quote Richard Shull “The fact that an equation of a particular form describes a set of data will does not mean that the assumptions that gave rise to the equation are supported. The same equation can be derived from very different sets of assumptions” (Shull, 1991, pp. 246).

Put it another way, if the meaning of a mathematical/formal model is in the linguistic expression it takes (that is, if there is a unique isomorphism between phenomena and algorithms) then either (a) we cannot explain how a theory can be expressed in different sets of equations or (b) we will not be sure about the effect the addition or the removal of a simple parameter may have. Paraphrasing (Chakravartty, 2001), theories and models can be given linguistic formulations but theories and models should not be identified with such formulations.

1 Incidentally, backpropagation is merely a mathematical procedure to deriving partial derivatives—that was originally proposed to model nationalism and social communications not neural networks (Werbos, 1974).
Representational Level

ANNs are connectionist models according to which information is not stored explicitly in symbols and rules but rather in the weights (strengths) of the connections; learning would consist of changes in such weights. It is claimed, rightly, that these are precisely the assumptions associative learning models are based upon and hence, wrongly, that ANNs are an ideal candidate to model associative learning. This quite straightforward argument is, in fact, a fallacy: As connectionists (at least implementational connectionists) themselves concede the way we represent learning, either as continuous changes of weighted connections or as the result of discrete symbolic processing, is a matter of convenience and therefore irrelevant to the study of the structures involved.

Interestingly, this debate has centered in the difference between associative models and computational models of conditioning (Leslie, 2001): It is understood that associative models are historically and conceptually linked to connectionism (Medler, 1998) whereas computational (aka symbolic) approaches take their ideas from information processing (Gallistel & Gibbon, 2001). In this debate we agree with Peter R. Killeen in identifying both approaches as formal (Killeen, 2001): Turing machines and ANNs (as well as RMA machines, the Game of Life, and any programming language) are both computational models\(^2\); in particular, Turing Machines and ANNs are equivalent in their input/output behaviour, that is, they compute the same problems and accept the same languages (Chomsky, 1956)\(^3\).

Functional Level

ANNs typically approximate solutions by iteratively minimizing an error function. And this can be understood as a type of learning that resembles learning by “trial and error” of which associative learning is an example. However, it is worth emphasizing that ANNs merely implement numerical methods. They are statistical tools – with a misleading name, and certainly not the simplest, fastest or most efficient techniques (see, e.g., Mitchie et al., 1994). On the other hand, associative learning models such as Rescorla and Wagner’s express dynamic laws: Against public opinion, animals do not make predictions and iteratively update an associative value through error minimization towards an optimal one. The associative value at a given time is the right associative value –that exactly describes to which extend the CS has become associated to the US. Let’s put it another way: In standard conditions, if the animal “learned” a CS-US association after one single exposure then the animal would be wrong and its corresponding behaviour un-adaptive (unless, of course, we exposed it to a very salient US like in flavor-aversion learning). That the system described by Rescorla and Wagner’s rule is limited by an asymptote (the reinforcing value of the US) does not confer any special status to such value –rather it just defines a constraint of the system.

Structural Level

We are told that the layout of an ANN, the way units are connected between layers, can be seen as a cognitive architecture and, as such, as a psychological model. Let’s take a computational example to counter-argue this point: In computer science network communication is modeled according to the so-called Open Systems Interconnection model (OSI) (Zimmerman, 1980), moving from the physical layer that describes the electrical specifications of the devices the networks consist of up to the application layer that describes how the user interacts with a given piece of software. The question is: Why don’t we use the OSI model as a psychological model? At the end of the day, structurally, OSI would make as good a psychological model as an ANN. In fact, the OSI model implements a hierarchical and integrated architecture, that is, the type of cognitive architecture that a computational model should allegedly support (Sun, 2008). Thus that ANNs are networks implemented in architectures that take advantage of massive computational parallelism – not surprisingly, the new connectionism landmark paper introduced the Parallel Distributed Processing paradigm in cognition (Rumelhart & McClelland, 1986), does not confer them any psychological advantage: Any complex network would do (Newman et al., 2006).

Philosophical Issues

A final more general reason to explain the appeal of computational models in psychology rests on the idea that both computers and the brain are information processing systems, instantiations of a universal Turing machine or any other model of computation. But this alone does not justify the support the “computer metaphor” enjoys. After all, any phenomena can be expressed in terms of some sort of computation. If this is such a powerful metaphor is because it is deeply rooted in Western philosophy and the mechanization of (formal) reasoning, reformulated in the twentieth century in terms of computation. That computation has been effectively embedded in computers has reinforced the idea that so it is in the brain, that the study of the former will help understand the latter and, in a tour the force, that computers may be capable of displaying intelligence. Indeed, every scientific theory is shaped in the context of its age’s achievements and prejudices: Like Newton's laws of mechanics strengthened the view of the Universe as a deterministic machine that worked as the sophisticated clocks so popular at the time our conception of the mind as an information processing machine has certainly been influenced by the development of computing technology.

And precisely because of its generality the information processing model is not necessary or sufficient: Working physicists do not model electrons, atoms or galaxies as information processing entities –be it in the form of a cellular automaton as envisaged in (Zuse, 1969) or as a participatory universe (Wheeler, 1990); on the other hand,
Model Selection
The discussion on what a computational model of psychology constitutes affects how we select models and in turn may help us determine what a computational model “truly” is.

Selecting a model, psychological or not, described in natural language or mathematically, is a difficult task that relies in formal definitions and methods as well as on scientific practice and common sense (Kuhn, 1962; Feyerabend, 1975). Indeed, quantitative formulas have been developed to compare models based on the average size of the deviations from predicted values, the number of data points and the number of free parameters (Akaike, 1974; Schwarz, 1978). However, relying exclusively on such formalisms or applying blindly Occam’s razor is not advisable –evaluating a model requires good judgment based on careful consideration of many factors, both technical and logical (Baum, 1983). The very essence of a model refers to the choices scientists make –choices that reflect what they consider relevant beyond the mere quantitative.

Nonetheless, this analysis begs the question: When we assess computational models of psychology, what do we assess?

If computational models are simulators we would need to select amongst them according to their computational complexity, that is, according to the time and space they take to make computations –complexity that is related to but not reducible to the algorithms they implement. In addition, computing tools must be tested for reliability and dependability against failures –which, in turn, depends on various factors such as programming languages, operating systems, memory capacity, processing speed, as well as on software engineering and management requirements. Computational models as simulators add a new level of sophistication. But this sophistication comes at a price: A computer program is not as “aseptic” as a mathematical description. A computer program takes life in algorithms and data structures that must comply with software and hardware specifications.

On the other hand, if computational models are considered as a valid alternative to psychological models. Which criteria should we used to evaluate them and choose amongst them? There is no a clear answer to this question.

Conclusions
To sum it up, although the need to get influx from ‘outsiders’ is recognized within the psychology community (see Townsend, 2008) computational models should be taken with caution. Computational models may provide us with complementary idealized models of psychological phenomena and with powerful statistical tools to construct models of psychological data but they alone are not the appropriate instruments to answer psychological questions. This is an obvious, hardly original, conclusion –and yet more often than not we read flamboyant news about robots that learn, think and experience emotions or ANNs that can do anything psychological models do only better. On the other hand, given the increasing complexity of psychological models developing accurate and rapid simulators to test their predictions is, in our opinion, a must.

We would like to conclude with two warnings: An extreme case of the use of computational models as psychological models is what we call the “engineering” approach: We take psychological data and build a program that fits it. Since the data is psychological, it is argued, the program must constitute a psychological model. It should be obvious, however, that that psychological models, as any scientific model, are assessed against experimental data does not automatically make a theory that fits experimental data "psychological".

Another variant of this approach is to propose models of machine/computational learning as psychological models of learning. As an illustration, simple programs that, to some extent, learn to maximize a numerical signal by trial and error have been presented as a “theory of mind” (Sutton, 2003).

To summarize: The adjective "computational" in computational physics or computational biology refers to the use of computational tools, typically simulators and numerical processors but also data mining and data analysis techniques, to study data and phenomena as well as to assess the predictive power of theories and models. We suggest we “limit” the use of the term “computational” the same way when applied to psychology.

References


