Modelling of Novices' Control Skills With Machine Learning

Rafael Morales and Helen Pain*

School of Artificial Intelligence, University of Edinburgh, United Kingdom

Abstract. We report an empirical study on the application of machine learning to the modelling of novice controllers' skills in balancing a pole (inverted pendulum) on top of a cart. Results are presented on the predictive power of the models, and the extent to which they were tailored to each controller. The behaviour of the participants in the study and the behaviour of an interpreter executing their models are compared with respect to the amount of time they were able to keep the pole and cart under control, the degree of stability achieved, and the conditions of failure. We discuss the results of the study, the limitations of the methodology in relation to learner modelling, and we point out future directions of research.

1 Introduction

Previous research on supporting teaching and learning cognitive tasks has concentrated on high-level skills such as problem-solving in mathematics and physics, programming, and second language learning. Acquisition of real-time control skills of the sort required for playing a musical instrument, driving a vehicle or operating a tool have received much less attention. This paper attempts to make a contribution to the latter, more neglected area, with respect to learner modelling. Descriptions of strategies followed by apprentices of the simple task of balancing a pole attached to a cart are obtained by applying machine learning techniques to traces of the apprentices' behaviour. We consider whether these can be regarded as adequate representations of the evolving control skills of novices.

Machine learning techniques have been applied to pole balancing and other controlling tasks like flying a plane, operating a crane, and production scheduling (see Bratko et al., 1997, for an overview; Michie et al., 1990; Michie and Camacho, 1994; Urbančič and Bratko, 1994). The methodology, termed *behavioural cloning* (Michie et al., 1990), was originally motivated by the difficulties encountered in getting expert controllers to produce detailed explanations of their skills that can be embedded in programs. Learner modelling differs, however, in a number of respects from expert modelling, owing to the fact that the subject is not an expert, but a beginner whose behaviour manifests faulty and inconsistent performances.

The use of machine learning techniques for learner modelling has a long history (e.g. Gilmore and Self, 1988; Langley et al., 1984; Sleeman, 1982; Webb and Kuzmycz, 1996; see also Sison and Shimura, 1988). Machine learning offers the possibility of data-driven learner modelling, focused on the actual behaviour of the learner, without the prerequisite of detailed descriptions

* We thank Tom Conlon, Donald Michie, Kaśka Porayska-Pomsta, Michael Ramscar, Shari Trewin, Angel de Vicente, and three anonymous reviewers for comments on this paper. William Cohen deserves special thanks for allowing free use of RIPPER, and his prompt and kind response to all our questions. Rafael Morales is being supported by CONACYT and the Instituto de Investigaciones Eléctricas, Mexico, under scholarship 64999/111091.

of domain knowledge and its common variations (the latter usually referred to as 'misconceptions', 'bugs', or 'mal-rules'; cf. Sison and Shimura, 1988). The reduction of assumptions about domain knowledge gives grounds for expecting a decrease in the bias of the diagnosis, and hence greater flexibility to accommodate different (human) learning styles and different conceptions of domain knowledge (Jonassen and Grabowski, 1993). However, because machine learning does not necessarily relate to human learning, claims about the psychological status of models constructed with it have varied. Advocates of the approach have either attempted to embed their techniques in broader psychological theories (e.g. Langley et al., 1984), or they have assumed to model solely competence in the task, without claiming to describe plausible cognitive processes of human learners (e.g. Gilmore and Self, 1988; Webb and Kuzmycz, 1996).

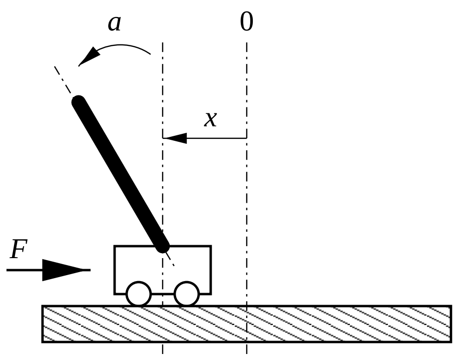
Our research differs from related work on behavioural cloning in that it focuses on apprentices, rather than experts. We are interested in making matches between subject and clone behaviour, whereas research on behavioural cloning has focused on maximising the clone's expertise. Our work differs from previous work on using machine learning for learner modelling in the time-constrained and highly dynamic nature of the domain, which demands a different approach to preparing the input data and evaluating the adequacy of the models. We focus on devising methods for diagnosing novice performers of real-time, control-like tasks, constructing representations of their strategies based on traces of their behaviour, and checking that the representations are faithful models of the novices' competence in the task. As to the psychological credibility of the models, we adopt a conservative approach: our intention is not to build accurate psychological models, but rather models that we could offer to learners as abstract representations of the strategies they follow; models that learners can identify themselves with and inspect as part of their learning process. This facet of the present work derives from our ongoing research on *participative learner modelling* (Morales et al., 1998).

To briefly summarize the rest of this paper, Section 2 describes an empirical study used to gather subject data. Section 3 we describes the behaviour of the participants in the task. The procedure of preparing the traces and inducing the models is presented in Section 4. The predictive power of the models is analysed in Section 5, their individualised nature is discussed in Section 6, and a comparison between the behaviour of participants and their respective models is made in Section 7. The general discussion of results and conclusions are given in Section 8.

2 The study

The basic task explored involved balancing a pole (inverted pendulum) attached to the top of a cart (wheeled vehicle) mounted on a straight track of finite length (Figure 1); the pole could fall over the cart only along the vertical plane passing through the track. The whole device could be controlled only by the application (or not) of a force of fixed magnitude, parallel to the track, but with a choice of left or right direction. A simulator based on existing code made available by Finton (1994) was coupled with a graphical user interface and then used instead of a physical device. In the empirical study, every *control run* started with a still pole tilted randomly ± 6 degrees on a still cart placed in the centre of the track, and ended whenever a *crash* occurred (i.e. any time the cart fell off the end of the track or the pole reached a horizontal position). User input was restricted to pressing arrow keys: $\uparrow$ to start a control run, $\leftarrow$ to push the cart to the left, and $\rightarrow$ to push the cart to the right. User keystrokes were collected for every 100 ms, the action corresponding to the last keystroke sent to the simulator, and the subsequent new state of the device displayed. The simulator was set up to calculate the state of the device in time increments of 20 ms. The combination of timings, of the interface and the simulator, resulted in a simulation five times slower than the real pole and cart device.

Six subjects took part in the study. They received a brief introduction to the task and the interface, and then were instructed to try keeping the pole in a non-horizontal position and the cart on the track. They were told to start a new control run after every crash. After five minutes of playing with the system, the participants were instructed to continue for another five minutes, and prompted to try harder in pursuing the task.



Parameter	Value
Cart mass	1 kg
Pole mass	0.1 kg
Pole length	1 m
Magnitude of force	10 N
Length of track	4.8 m

Figure 1. The pole and cart device. A position of the cart on the right (left) half of the track is taken to be positive (negative). An inclination of the pole to the right (left) of the vertical is considered positive (negative).

3 Behaviour of the participants

A straightforward measure of the performance of the participants is *control run length*, i.e. the amount of time they were able to avoid a crash. Because a participant could achieve a given control run length in several different ways, exhibiting different "control styles," we conceived an additional *index of stability* to give a more detailed account of the control process than the raw end result. The control strategy shown in Figure 2 defines a decreasing order of relevance of the state variables for controlling purposes, from the angular velocity of the pole to the position of the cart. Following it every state of the pole and cart was classified into one of five categories, in increasing order of stability: *falling* (0.0025), *tilted* (0.0474), *leaving* (0.5), *displaced* (0.9526), and *stable* (0.9975); the stability index per category was obtained by evaluating the sigmoid function $s(x) = 1/(1+e^{-3x})$ at $x = -2, -1, 0, 1, 2$. The stability of a control run was then calculated by summing up the stability of all its states, divided by the total of states in it. An overall stability index per participant was calculated as the cumulative effect of the whole set of states of the pole and cart generated by each participant. The last characteristic of the participants' behaviour we considered

falling:	**if** $\dot{a} > 0.5$ **push** right **else if** $\dot{a} < -0.5$ **push** left
tilted:	**else if** $a > 0.07$ **push** right **else if** $a < -0.07$ **push** left
leaving:	**else if** $\dot{x} > 0.4$ **push** right **else if** $\dot{x} < -0.4$ **push** left
displaced:	**else if** $x > 0.5$ **push** right **else if** $x < -0.5$ **push** left
stable:	**else** do nothing

Figure 2. Adaptation of a succesful control strategy from (Michie et al., 1990). Angles (a) are measured in radians (clockwise direction is positive), angular velocities ($\dot{a}$) in radians per second; cart positions (x) in metres, and cart velocities ($\dot{x}$) in metres per second. The adaptation consists in a change of the tresholds for the position of the cart, from zero to ± 0.5.

was the object they finally crashed, either the pole or the cart. Statistics per participant of these three aspects of their behaviour are presented in Table 1.

The behaviour exhibited by the group was far from the expert behaviour reported by Michie et al. (1990), whose expert was able to control the device for five minutes. In our study the longest control run lasted only two minutes, and even that was atypical of the performance of the participants. According to control run length, the participants seem to split into three categories: “short run” performers (participants S_3 and S_6), “medium run” performers (participants S_2 and S_4), and “long run” performers (participants S_1 and S_5). The lack of expertise and the variety in the group of participants can be regarded as a useful test of the robustness of the methodology and the adequacy of the models it produces.

Although it was possible to achieve long runs within a adventurous style (low stability), and to get short runs in a cautious manner, we expected a positive correlation between the index of stability and control run length. The results on overall stability, shown in Table 1, indicate that participants S_3 and S_6 had great difficulties at controlling the angular velocity of the pole; they achieved a very small number of stable and displaced states. S_1, S_2, and S_4 performed better, achieving higher stability scores, but below the outcome of S_5. In general, there was high variability in stability across control runs for all participants, although with a tendency to gain in stability over time. The performance of participant S_6 showed the least variation.

To prevent miscounting as a cart crash a loss of control of the pole from which it is impossible to recover even if there were more space in the track, a pole with an inclination of more than twelve degrees in either direction was regarded as crashed, as in (Bratko, 1995; Michie et al., 1990). As before, the behaviours of participants S_3 and S_6 were quite similar: both had difficulties in controlling the pole. Participants S_2 and S_4 again had similar behaviour, exhibiting less difficulties in controlling the pole than S_3 and S_6. Participant S_1 achieved relatively good control over the pole early in the study, but did not improve very much afterwards. On the other hand, participant S_5 had initial difficulties at controlling the pole, followed by a dramatic improvement.

4 Modelling procedure

All user actions on the simulated pole and cart were recorded in trace files in the general form *device status* $\rightarrow$ *user action*. The *device status* contained the values of the pole and cart positions and velocities, as displayed on the screen for the last 100 ms. The *user action* was either the

Table 1. Statistics per participant of control run length, stability index, and crashing conditions. Medians and geometric means are included because the distributions are skewed positively. Runs lasting less than 2.5 seconds were not taken into account in calculating the statistics of control run length and stability. There were five such control runs, and three of them are counted in the *Other* category in the section on crashing conditions.

Property	Statistics	S_1	S_2	S_3	S_4	S_5	S_6
Control run length	Median	23.1	16.7	9.4	18.6	19.8	9.4
(in seconds)	Geom. mean	22.6	17.1	9.4	18.5	20.3	9.7
	Mean	32.0	20.7	10.8	21.6	31.9	10.5
	Std dev.	29.1	13.3	5.9	12.9	29.2	5.0
Overall stability index		0.25	0.30	0.10	0.27	0.47	0.03
Crashing conditions	Pole	12	22	48	19	13	56
	Cart	7	8	0	9	5	2
	Other	0	0	1	0	4	0

action corresponding to the user's last keystroke in the last 100 ms, or a "no action" encoding the lack of a user keystroke in the same period. The procedure for extracting the models consisted of three steps: preparation of the traces for diagnosis, induction of a set of production rules, and informed refinement of it into a learner model.

Two problems had to be solved in the preparation stage. Due to possible delays between perception and action, we could not simply assume that the user action stored in a record corresponded to the pole and cart status in that same record: it could correspond to an earlier status of the device, displayed some hundreds of milliseconds before and hence stored in a previous record. A related problem was the treatment of no-actions, introduced by the system every time no keypress ocurred for the last 100 ms; the greater the reaction delay, the more *undesired no-actions* it caused. Observations during the study made clear that some no-actions were the correct interpretation of the participants' intentions, and hence it would be unwise for us simply to remove all no-actions from the traces.

The mean of the lag between the start of a control run and the issue of the participants' first action provides an estimated upper limit to reaction time in the task ($N = 202$, mean $= 706.4$ ms, std $= 318.0$ ms; median $= 647.5$ ms). It is likely for the task to become less dependent on raw reaction time after the first action has been issued[1]. We chose the value of 300 ms for reaction time, based on the mean lag of 343.2 ms between pairs of consecutive actions ($N = 10022$, std $=$ 442.3 ms; median $= 180.0$ ms), remarkably close to the estimated reaction time of 350 ms to pressing a key in response to a simple visual stimulus produced on the screen (Cotterill, 1989, cited by Michie et al., 1990). In practice, that meant aligning the device state and user actions with a shift of three, $\textit{device status}_k \rightarrow \textit{user action}_{k+3}$, and stripping the traces of all sequences of no-actions with less than three elements (cf. Michie et al., 1990).

[1] Decisions and actions, even if elicited in response to the present state of the device, are primed by previous states and actions; accumulated knowledge of the task allows some actions to be planned in advance; and some degree of parallelism of the cognitive processes of states perception, selection of responses, and execution of motor actions evolves.

We divided the sequence of control runs of each participant up in overlapping sections of roughly five minutes long (such that they did not split any control run). A five minute window was displaced over the sequence of control runs in steps of around thirty seconds, resulting in six sections for all participants apart from S_1, who got only four sections. Variations in the number of sections and their span came as a result of not splitting control runs and the variability in the length of control runs achieved by the participants. The groups of records resulting from the alignment, filtering, and sorting out described above were finally presented as input data to RIPPER, a domain-independent rule-learning system (Cohen, 1995).

Specific traits of the domain, such as symmetry, the range of the variables, and their interrelationship, could not be dealt with in the induction process itself. The limited amount of data, and the fact that both the starting and final states in every control run are necessarily asymmetrical, obscured the symmetry of the domain. In order to compensate for these limitations, we introduced symmetrical cases as input to the induction process: for every case *device status* $\rightarrow$ *user action*, a new case with a symmetrical device status and opposite action was included too. Post-processing of rule sets consisted of substituting a default rule for all the rules issuing no-actions.

5 Predictive power of the models

The procedure described in the previous section produced a number of rule sets per participant—four from S_1, and six from all other participants. For these rule sets to be properly called *individualised* models, they should exhibit two properties: first, they should match the subjects they were extracted from, and second, they should differ from each other[2].

For each five minute window (as described above) a model was induced. The predictive power of this model was tested on the following (roughly) thirty second window of control runs (again avoiding splitting individual runs). The final five minute window in each case was not verified. The results, presented in Table 2, were error rates between 8.4 and 44.2 percent ($N = 28$, mean $= 30.0$, std $= 8.7$). There is a significant difference between lower error rates for participant S_3, and higher error rates for S_2, S_3, S_4, and S_5 (One-way analysis of variance: $dfs = 4, 20, 24$; $F = 7.1407$, $p < 0.001$. Tukey-HSD test with significance level 0.05. Results on participant S_1 are excluded from the tests because of their different number of models).

Table 2. Predictive power of the models per participant.

Property	S_1	S_2	S_3	S_4	S_5	S_6
Mean no. of cases	366.7	154.0	167.0	213.8	235.2	207.2
Min. error (%)	27.0	26.8	21.7	28.5	23.1	8.4
Max. error (%)	36.4	39.9	42.3	44.2	41.7	21.6

Although these results are statistically highly significant when compared to raw random guessing (the binomial test of the combination of the least number of cases, $N = 118$, and the

[2] If the models were all very similar, they could still be individualised models, but we would not have evidence supporting that.

worst error rate gives $p < 0.001$), they do not argue for a good match by themselves. It could be argued instead that an undetermined amount of the error rates is due to errors in the alignment of states and actions (Section 4). Despite this caveat, it is worth mentioning that a mean of 72.4% of actions per participant were predicted by their models (min = 49.4, max = 93.5, std = 11.9), and that only a mean of 5.3% of actions per participant were predicted in the wrong direction (min = 0, max = 14.5, std = 4.8).

6 Differentiability of the models

Because participants in the study exhibited clear differences of behaviour in their attempts to control the pole and cart, we expected such differences to be apparent also in their models; i.e. models from the same participant should be similar among themselves and different to those from other participants. We opted for a simple dissimilarity measure between models: the level of disagreement in their predictions. A straightforward measure was defined in terms of the traces of the participants' behaviour as

$$d(M_a, M_b) = \frac{1}{\#C_a + \#C_b} \left(\sum_{c \in C_a} (M_b(c) - C_a(c))^2 + \sum_{c \in C_b} (M_a(c) - C_b(c))^2 \right), \qquad (1)$$

where M_a and M_b are models; C_a and C_b are case sets from which M_a and M_b were extracted, respectively; $C_x(c)$ is the action corresponding to case c in C_x; $M_x(c)$ is the action predicted by model M_x for case c; and actions are encoded as -1 for pushing-left, 1 for pushing-right, and 0 for no-action. The problem with this measure is that it depends on the accuracy of the alignment between states and actions, as recorded in the case sets. A second measure allows to compare directly the predictions given by models on the basis of a sample of the set of states generated by the participants during the study. It is defined as

$$d(M_a, M_b) = \frac{1}{1000} \sum_{s \in S} (M_a(s) - M_b(s))^2, \qquad (2)$$

where S is a sample of one thousand of such states.

Two cluster analyses were then applied to both dissimilarity matrices, using average group and Ward's method (Everitt, 1993). The number of groups were selected on the basis of visual inspection of the dendrograms produced by the clustering methods and plots generated by multidimensional scaling. The analyses suggested three and four groups using dissimilarity measure (1), and five groups using dissimilarity measure (2). Overall, they identify models corresponding to participants S_3 and S_6; Ward's methods also distinguished (some) models corresponding to S_1 and S_2; but none of the analyses distinguished between models of S_4 and S_5. The analyses agreed among them in 405 of 561 decisions (72%), and agreed with the known grouping in 360 of the decisions (64%)—there were $\binom{34}{2} = 561$ pairs of models that could be classified either in the same or different group. The best match with the known grouping was given by the Ward's method using dissimilarity measure (2): 480 of 561 decisions (86%). A binomial test shows that all results are highly significant in reproducing the correspondence between models and participants (random guess with $N = 561$ and $k = 360$ gives $p \ll 0.0001$).

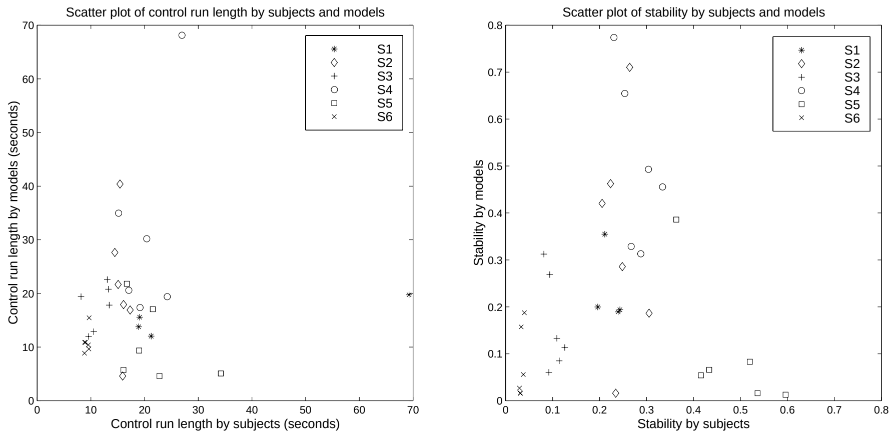


Figure 3. Comparison of control run length and stability achieved by subjects and models. For clarity, control run length for model $M_{4,6}$ is not shown (235.06 seconds).

7 Behaviour of the models

We built an interpreter for the models with a fixed reaction time of 300 ms, and ran every model fifty times, with the starting state of the pole and cart as before. The geometric mean of control run length and overall stability were measured both in the control runs produced by executing each model and in the control runs the model was derived from in the first place (Figure 3). The comparison between subjects' and models' performance gave three general results:

1. Models of participants S_3 and S_6 exhibited behaviour very similar to their respective subjects: short control runs, low stability, and difficulties in controlling the pole. Models of S_6 also showed high predictive power.
2. Models of participant S_4 usually outperformed participant S_4, with long control runs of high stability and good control. This relates to the *clean-up effect* observed by Michie et al. (1990), consisting of the behavioural clone outperforming the expert it was derived from.
3. Models of participant S_5 often performed considerably worse than participant S_5, and on several occasions performed worse than models of S_3 and S_6, despite the fact that S_5 was far better at controlling the pole and cart.

8 Discussion and conclusions

The assumption of a fixed amount of reaction time, both within a participant's control runs and among participants, made the alignment of states and actions in traces of the participants' behaviour easier, and we did not have any good arguments for doing otherwise. However, this neglects differences in sensorimotor abilities among participants, and the possibility of improvement with practice. It is possible that some participants overcame their sensorimotor limitations by taking into account additional information for short term planning, and issued actions more accurately.

Poorer control of the pole and cart produces a good sample of the whole space of possible states of the device, and corresponding control actions, from which accurate models can be induced. Behavioural cloning produces a small cleanup from low quality control and improves considerably medium quality control. In contrast, tight control of the pole and cart produces a biased sample of the whole space of control from which it is more difficult to induce a complete model of the control strategy followed by the subject. This, combined with the clean up attempted by behavioural cloning, resulted in brittle, low quality control strategies. Furthermore, participant S_5 had the most economical control strategy, in terms of the number of actions issued. That property translated into traces containing lots of *no-actions* that were overestimated by the machine learning algorithm[3], which in turn produced extremely economical models that were unable to keep the pole and cart in conditions such as those contained in the sample of the control space from which the models were induced.

Some issues concerning the application of machine learning to learner modelling, as performed in our study, need to be considered. Very little knowledge about the domain was taken into account for the production of the models. More information about the symmetry of the domain, the nature of the actions, and critical regions in the space of control during the process of induction, would produce better results. Also, the resulting models are "flat," embodying a purely reactive conception of the task. However, participants were presented with a goal: keep the pole and cart under control for as much time as possible; at least some of them appeared behaving in a goal-oriented way. In this respect, other machine learning approaches, like inductive logic programming, could be interesting alternatives to the more traditional machine learning techniques employed in our study (cf. Chiu et al., 1997). An approach to learner modelling simply based on extraction of the current model from the last minutes of the learner's behaviour appears to be too limiting, resulting in high variability of the models over time. A proper learner model maintenance module, comprising machine learning techniques as subcomponents, would be needed. Ten minutes of controlling behaviour per participant provides too little data to say anything conclusive about the learning of the participants and the evolution of their models.

In conclusion, we have presented an empirical study in applying machine learning to model novices in the task of controlling a pole on a cart. Our results indicate it succeeded in distinguishing between different subjects, hence producing clearly different models for them. A static test on data collected from the participants showed the models performed well at predicting the actions of the subjects in the short term, especially if the likely presence of noise due to our assumption of fixed reaction time is taken into account. Dynamic tests of the models consisted of executing them and comparing their performance with the original performance of the participants. A close correlation was noticed for three of the participants, while an evident discrepancy was discernible in the remaining cases. We advanced an explanation for the latter failure in terms of reaction time, the clean up of control behaviour and the high degree of stability achieved by some participants, and suggested some ways to overcome the deficiencies of our approach. Our results show machine learning can be a useful tool for diagnosis in learner modelling in domains involving control tasks, although it needs to be enhanced with more domain-specific knowledge, and embedded into a more comprehensive learner model maintenance system.

[3] We tried to ameliorate this effect by weighting false positives and false negatives in RIPPER. The effect was clearly appreciated in better performance of models from S_2.

References

Bratko, I., Urbančič, T., and Sammut, C. (1997). Behavioural cloning of control skill. In Michalski, R. S., Bratko, I., and Kubat, M., eds., *Machine Learning and Data Mining: Methods and Applications*. John Wiley & Sons. chapter 14, 335–351.

Bratko, I. (1995). Derivating qualitative control for dynamic systems. In Furukawa, K., Michie, D., and Muggleton, S., eds., *Machine Intelligence*, volume 14. Oxford: Clarendon Press. 367–386.

Chiu, B. C., Webb, G. I., and Kuzmycz, M. (1997). A comparison of first-order and zeroth-order induction for Input-Output Agent Modelling. In Jameson, A., Paris, C., and Tasso, C., eds., *User Modeling: Proceedings of the Sixth International Conference, UM97*, 347–358. Chia Laguna, Sardinia, Italy: Springer Wien New York.

Cohen, W. W. (1995). Fast effective rule induction. In Prieditis, A., and Russell, S., eds., *Machine Learning: Proceedings of the Twelfth International Conference*. Tahoe City, CA: Morgan Kaufmann.

Cotterill, R. (1989). *No Ghost in the Machine*. London: Heinemann.

Everitt, B. (1993). *Cluster Analysis*. London: Edward Arnold, 3 edition.

Finton, D. J. (1994). Controller-less driver for the cart-pole problem. Available on the World Wide Web at `http://www.cs.wisc.edu/~finton/poledriver.html`.

Gilmore, D., and Self, J. (1988). The application of machine learning to intelligent tutoring systems. In Self, J., ed., *Artificial Intelligence and Human Learning: Intelligent Computer-Aided Instruction*. London: Chapman and Hall Computing. chapter 1, 179–196.

Jonassen, D. H., and Grabowski, B. L. (1993). *Handbook of Individual Differences, Learning, and Instruction*. Lawrence Erlbaum Associates.

Langley, P., Ohlsson, S., and Sage, S. (1984). A machine learning approach to student modeling. Technical Report CMU-RI-TR-84-7, The Robotics Institute, Carnegie-Mellon University, Pittsburgh, Pennsylvania, USA.

Michie, D., and Camacho, R. (1994). Building symbolic representations of intuitive real-time skills from performance data. In Furukawa, K., Michie, D., and Muggleton, S., eds., *Machine Intelligence*, volume 13. Oxford: Clarendon Press. 385–418.

Michie, D., Bain, M., and Hayes-Michie, J. (1990). Cognitive models from subcognitive skills. In McGhee, J., Grimble, M. J., and Mowforth, P., eds., *Knowledge-Based Systems for Industrial Control*. London: Peter Peregrinus. chapter 5, 71–99.

Morales, R., Ramscar, M., and Pain, H. (1998). Cognitive effects of participative learner modelling. In Ayala, G., ed., *Proceedings of the Current Trends and Applications of Artificial Intelligence in Education Workshop*, 49–56. Mexico City, Mexico: ITESM.

Quinlan, R. J. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann.

Sison, R., and Shimura, M. (1988). Student modeling and machine learning. *International Journal of Artificial Intelligence in Education* 9:128–158.

Sleeman, D. H. (1982). Inferring (mal) rules from pupil's protocols. In *ECAI-82 : 1982 European Conference on Artificial Intelligence*, 160–164.

Urbančič, T., and Bratko, I. (1994). Reconstructing human skill with machine learning. In Cohn, A. G., ed., *ECAI 94: 11th European Conference on Artificial Intelligence*, 498–502. Amsterdam, The Netherlands: John Wiley & Sons.

Webb, G. I., and Kuzmycz, M. (1996). Feature based modelling: A methodology for producing coherent, consistent, dynamically changing models of agents' competencies. *User Modeling and User-Adapted Interaction* 5:117–150.