State-Trace Analysis of Sequence Learning by Recurrent Networks

Fayme Yeates[1] (fy212@exeter.ac.uk) Andy Wills[1] (A.J.Wills@exeter.ac.uk)
Fergal Jones[2] (Fergal.Jones@canterbury.ac.uk) Ian McLaren[1] (I.P.L.McLaren@exeter.ac.uk)

[1]School of Psychology, College of Life and Environmental
Sciences, University of Exeter, UK.
[2]School of Psychology, Canterbury Christ Church University, UK.

Abstract

This study investigated the use of state-trace analysis (Bamber, 1979) when applied to computational models of human learning. We aimed to investigate the performance of simple recurrent networks (SRNs) on a sequence learning task. Elman's (1990) SRN and Cleeremans & McClelland's (1991) Augmented SRN are both benchmark models of human sequence learning. The differences between these models, comprising of an additional learning parameter and the use of response units activated by output units constituted our main manipulation. The results are presented as a state-trace analysis, which demonstrates that the addition of an additional type of weight component, and response units to a SRN produces multi-dimensional state-trace plots. However, varying the learning rate parameter of the SRN also produced two functions on a state-trace plot, suggesting that state-trace analysis may be sensitive to variation within a single process.

Keywords: Learning; state-trace analysis; SRN; sequence learning; Augmented SRN;

Introduction

State-trace analysis (Bamber, 1979) is a method that aims to establish whether one or more underlying processes are influencing behavior on a given task. The method has been applied to a variety of paradigms, including remember-know tasks (e.g. Dunn, 2008), face recognition (e.g. Loftus, Oberg, & Dillon 2004), categorization (e.g. Newell, Dunn & Kalish, 2010) and a variety of other areas (see Prince, Brown & Heathcote, 2011).

The procedure for a state-trace analysis is to plot the relationship between two dependent variables (*dimensions*) on two or more tasks (*states*). If these points follow a single, monotonic function, it can be hypothesized that the same latent variable underlies performance on the tasks. The influence of more than one latent variable on the tasks is implied when the state plots do not follow the same function, i.e. more than one monotonic function is visualized.

Computational models are created in the full knowledge of the processes involved in their construction. Thus, the primary use of state-trace analysis, to attempt to quantify latent psychological variables, does not seem to directly lend itself to computational modeling. But the fact that we should be able to make some predictions from the nature of the models about the types of processes involved in any simulation helps us interpret any state-trace analysis of the data produced by the simulation. This paper seeks to apply state-trace analysis to the simulation results produced by computational models on a sequence learning task both to evaluate the different types of model and as a means of evaluating the state-trace methodology itself.

The computational models chosen for this analysis are the simple recurrent network (SRN) introduced by Elman (1990) and the Augmented SRN (Cleeremans and McClelland, 1991). The basic SRN model is simple (see Figure 1), involving feed-forward input activation through a hidden layer. The activations of this hidden layer are copied back on each trial into a context layer, which is then fed back into the hidden layer as input on the next trial. This ensures that the representations of the previous trial are carried over, and gives the model the ability to learn sequential information.

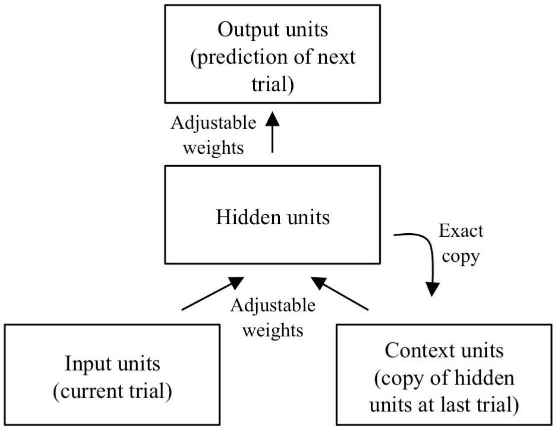

Figure 1: The architecture of the SRN (Elman, 1990).

Cleeremans and McClelland (1991) further developed the SRN in order to give a better account of the sequential effects demonstrated by human participants. This augmented simple recurrent network (AugSRN) differs from Elman's (1990) original architecture firstly in the inclusion of response units post-output. As a consequence, when making a response to the stimuli in an experiment, this response remains primed over future trials for a short time (Remington, 1969), because the response units are activated by the output units and feed activation back into the output units on the next trial.

The AugSRN also accounts for the priming of certain sequential pairings (Cleeremans & McClelland, 1991) by assuming that back propagation is implemented on not one set of connection weights, but two. One component is a set of slow weights, which produce small but permanent changes with minimal decay. These are complemented by fast weights, which have a higher learning rate but also a greater rate of decay: simulating transient, short-term learning.

Both models have successfully simulated a range of human datasets (Cleeremans, 1993), including modeling sequence learning in serial reaction time (SRT) tasks. Jones and McLaren (2009) found that an AugSRN could produce the detailed pattern of subsequence learning demonstrated by participants in their experiments. Using a two-choice SRT task participants received continuous strings of stimuli that followed an exclusive-or rule two-thirds of the time. If the previous two responses were the same (XX or YY), then the current trial would be one response (X), and if the previous two had not been the same (XY or YX) this would lead to the other response (Y). Participants under incidental conditions found it hard to learn about the subsequence XXX compared to the other types, a result which was successfully simulated by the AugSRN.

In a later experiment Yeates, Jones, Wills, McLaren and McLaren (submitted) used the same two-choice SRT task to investigate human sequence learning. Participants in this study were divided into two groups, both of which received sequences governed by a rule that they were not informed about. In one group, the current trial could be predicted two-thirds of the time to be different to the trial before last, i.e. 'first different to third' (Group 1: XXY, XYY, YYX, YXX) and in the other the current trial would be predicted to be the same as the trial before last, 'first same as third' (Group 2: XXX, XYX, YYY, YXY). Poorer performance was predicted in Group 2 under incidental conditions based on Jones and McLaren's (2009) earlier findings that participants were unable to learn about subsequence XXX (or YYY). The manipulation did indeed produce this difference between Groups. We found that variants of the SRN and AugSRN could simulate these data to differing extents depending on the parameterisation of the model (Yeates et al., submitted), which suggests that the differences between the SRN and AugSRN may be of interest in this context.

SRNs are considered to be single-process models (e.g. Frensch & Miner, 1994; Kinder & Shanks, 2003), where parameters can be altered to produce different effects, but these involve essentially one process. The standard view is that the two connection weight components in an AugSRN represent the same kind of process; of learning through back propagation, and that their differences are of amount and not kind. Varying the learning rate affects the efficiency of learning across training (Kinder & Shanks, 2001; McClelland & Rumelhart, 1986). However, one might hold the view that the two connection weight components are in fact different processes within the AugSRN, accounting for long- and short-term learning. Similarly, as response units were introduced to take account for short term priming of the previous response (Cleeremans & McClelland, 1991), we could argue that this additional component may also represent an additional, different process.

We therefore hypothesize that when comparing performance of the SRN and AugSRN we will see a clearly multi-dimensional state-trace plot, as the two models are different in kind. Further to this, we aim to examine the components of the AugSRN in more detail, with the aim to investigate whether state-trace analysis considers these additions to the original SRN separate processes within the model.

Given this analysis, our approach was to produce a state-trace analysis of these models' performance on a task based closely on the two-choice serial reaction time (SRT) experiments described in Jones & McLaren (2009) and Yeates, Jones, Wills, McLaren and McLaren (submitted). We aimed to compare the performance of these networks on this task, varying the free parameters of the models.

Modeling Sequence Learning

The SRT paradigm involves participants responding to stimuli on screen that follow some sequence (Nissen & Bullemer, 1987; Lewicki, Czyzewska, and Hoffman, 1987). Therefore, faster and more accurate responses are expected for those trials that are predicted by the sequences learnt in comparison to a control group, who would receive the same task but with a pseudorandom ordering (e.g. Anastasopoulou & Harvey, 1999, Jones & McLaren, 2009).

SRT Task Outline

The task experienced by each network follows closely that we have used with human participants, and lasted for two sessions, each with 20 blocks. Each block comprised 120 continuous trials of stimuli appearing on the right or left. The sequences making up each block were constructed differently for Group 1 and 2, and for the networks acting as control groups. For the experimental networks, all blocks in the first session and the first fifteen sequences in the second session were constructed from 40 triplets that followed the rule for each Group (Group 1: XXY, XYY, YYX, YXX; Group 2: XXX, XYX, YYY, YXY). Networks thus received ten of each subsequence type per block.

Two-thirds of experimental training trials followed the rule, as the third trial in a triplet was always consistent with the rule, as were half of first and second trials in a triplet by chance when subsequences were randomly concatenated. Test and control group training blocks were made up of pseudorandom sequences that included an equal amount of all subsequence types.

Model Construction

The parameters varied in the model for the purpose of the state-trace analysis are the number of hidden units and the learning rates, as well as the presence or absence of response units and presence of one or two connection weight components. Two units for both input and output were chosen to represent the stimuli (right or left circle fill) and predictions for the next trial (right or left), respectively. The activation of a single input unit was set to one, with the other set to zero to correspond to a left or right stimulus presentation. The units in each layer, from input and context to hidden and to output units, fed activation forward to every unit in the layer above (see Figure 1). The activation of the hidden and output units were determined by the logistic activation function (Rumelhart, Hinton, & Williams, 1986). Hidden unit activation was copied back to the context units on each trial with a lag of one cycle of the network. Each hidden unit also had a bias: a variable connection from a unit that had a constant activation of one. The hidden units mapped recurrently to the context units on a one-to-one basis. The feed-forward connections comprised of either one or two connection weights. These were modified by the back-propagation algorithm, which we ran without a momentum term (Rumelhart et al., 1986).

To simulate the experiment with humans reported in Yeates et al. (submitted), each model was run 128 times to match the number of participants taking part in the empirical study. Half of these simulations acted as controls (trained on pseudorandom sequences), with half receiving experimental sequences. 32 experimental networks followed Group 1 rules ('first different to third') and 32 followed Group 2 rules ('first same as third'). Initial connection weights were set for each network to random values between -0.5 and 0.5. Each simulation involved training for one session and fifteen blocks of a second session, followed by five blocks of test sequences. Therefore each simulation received 4200 training trials and 600 test trials.

The mean square error (MSE) was calculated as the difference between the location of the next trial, and the prediction of the model (see Jones & McLaren, 2009). This was taken as the measure of performance of the model on the task. As in previous simulations of these tasks, the MSE for trials consistent with the trained rule was taken from the MSE for inconsistent trials (Jones & McLaren, 2009; Yeates et al., submitted). This produces an estimate of learning about those trained sequences, and is also computed for control simulations. Half of the control simulations are assigned to the dummy variable Group 1, where 'first different to third' subsequences (XXY, XYY, YYX, YXX) are taken from the matching 'first same as third' subsequence (XXX, XYX, YYY, YXY). The remaining 32 simulations follow the Group 2 inconsistent-consistent calculation, with the MSE on 'first same as third' subsequences taken from the MSE on 'first different to third' subsequences. Comparing the differences between experimental and control groups on these scores allows learning to be assessed without any confound in terms of sequential effects (see Anastasopoulou & Harvey, 1999; Jones & McLaren, 2009). To summarize then, good learning will result in a larger difference of the form (Control network MSE for inconsistent trials - Control network MSE for consistent trials) - (Experimental network MSE for inconsistent trials - Experimental network MSE for consistent trials), as a lower MSE indicates better learning.

State-Trace Analysis 1: SRN and AugSRN

The task was simulated on an SRN and AugSRN with 20 hidden units. The SRN had a learning rate of 0.4, the AugSRN had a slow learning rate of 0.4 and a fast learning rate of 0.533. The AugSRN also possessed response units, unlike the SRN.

Results. Both models produced significant learning of both sequences, which was analyzed by means of an ANOVA with subsequences and blocks as within-subject factors, and experimental versus control as a between subject factor. The SRN exhibited a significant difference in consistent-inconsistent MSE scores between experimental and control simulations, $F(1,124) = 613.6$, $p < .001$. The AugSRN also demonstrated learning, $F(1,124) = 1113$, $p < .001$. As this difference of differences measures the learning in experimental networks compared to networks that experienced pseudorandom sequences (controlling for sequential effects), this difference is used to provide our index of learning and performance in what follows.

The SRN and AugSRN constituted the two states we wished to analyze, and we plotted performance of Group 1 against Group 2 on the axes as the dimensions. The plots follow the trace of training over collapsed blocks of five, with the seven points shown constituting the 35 training blocks. Figure 2 shows the state-trace plot of this data.

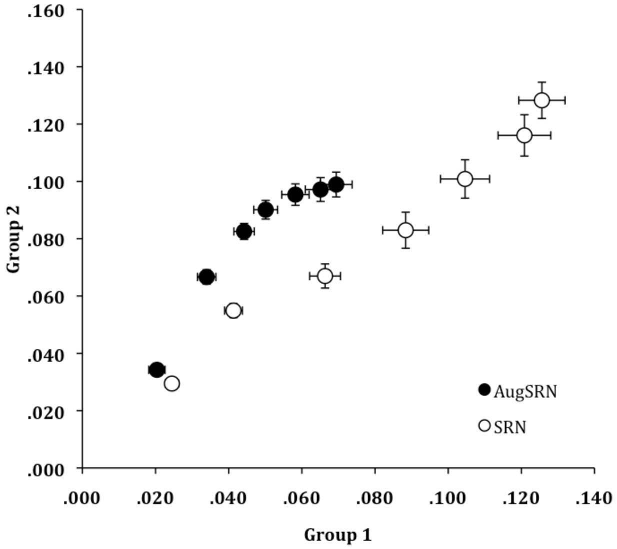

Figure 2. State-trace plot showing learning of AugSRN and SRN across training blocks of simulations.

Inspection of Figure 2 clearly suggests that there are two different, monotonic functions on the plot. We analyzed the data using hierarchical multiple regression, with the hypotheses that the model predicting Group 2 performance from Group 1 performance would be improved by the addition of Model Type as a variable, indicating a multi-dimensional model of the data. We simply coded this as a dichotomous nominal variable, with the AugSRN arbitrarily labeled as 1 and the SRN as 2. The addition of Model Type into the regression model significantly improved the R^2_{adj} value from 70.2% to 92.1%, ΔR^2 : $F(1,11) = 34.1$, $p < .001$, and overall, the model had a significant fit, $F(2,11) = 76.7$, $p < .001$; Group 2 = -.944xGroup 1 - .029xModel Type + .064. This corroborates the impression that the data on this plot require more than one function for a good fit, which suggests that there is more than one underlying process governing performance in these simulations. The conclusion is that the SRN and the AugSRN differ in kind (which is perhaps not surprising – though they are very similar types of model), but the important finding here is that the state-trace methodology is sensitive to this difference.

Discussion. This confirms our predictions about how we expected state-trace to represent learning by these different networks on this task, producing different functions and so confirming that they are genuinely different types of model. This could easily be attributed to either or both of the two differences between the SRN and AugSRN. We now investigate whether these models are themselves best characterized as single or multi-process models of learning.

State-Trace Analysis 2: Connection weight components

Here we ran simulations on four different models, two had response units and two had no response units. Within these dyads, we aimed to compare whether fast and slow weight components (the states in this state-trace analysis) were driving the multi-dimensional model seen in our first State-Trace Analysis. Therefore Model 1 had one connection weight component with response units, Model 2 had two connection weight components with response units (an AugSRN), Model 3 had one connection weight component with no response units (a standard SRN), and Model 4 had two connection weight components with no response units. Both had 20 hidden units and slow and fast weights of .4 and .533 respectively, as in the previous simulation.

Results. All four models learnt the sequences, analyzed as in Simulation 1. Model 1 showed a significant difference between experimental and control performance, $F(1,124) = 853.6, p < .001$. Models 2 and 3 showed learning, as seen in the results of State-Trace Analysis 1. Finally, Model 4 demonstrated the same learning, $F(1,124) = 1634, p < .001$.

When comparing models with one and two connection weight components we can see from Figures 3 and 4, which show the state-trace plots for models with and without response units respectively, that two monotonic functions appear.

When conducting a hierarchical linear regression as described in State-Trace Analysis 1, we this time coded Model as a predictor with the values of 1 and 2 for one and two components, respectively. Introducing Model into the regression alongside Group 1 in predicting Group 2 performance for models with response units (see Figure 3) produced a significant improvement in the R^2_{adj} value from 84.7% to 93.1%, ΔR^2 : $F(1,11) = 15.9$, $p = .002$, and overall, the model had a significant fit, $F(2,11) = 89.4$, $p < .001$; Group 2 = 1.231 Group 1 + .014 Model Type - .007. Similarly, when there are no response units (see Figure 4) adding Model as a predictor improves the regression model, with a significant improvement in the R^2_{adj} value from 56.0% to 89.6%, ΔR^2 : $F(1,11) = 40.0$, $p < .001$, and overall, the model had a significant fit, $F(2,11) = 56.9$, $p < .001$; Group 2 = 1.008xGroup 1 + .042xModel Type - .041.

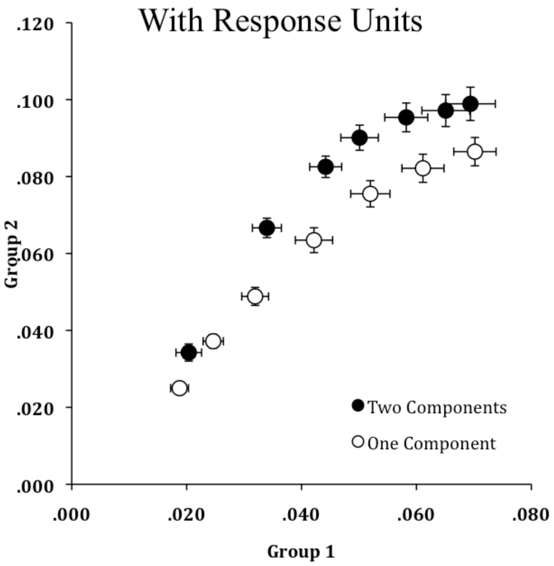

Figure 3. State-trace plot for Model 1 (one connection weight component) and Model 2 (two connection weight components) across training.

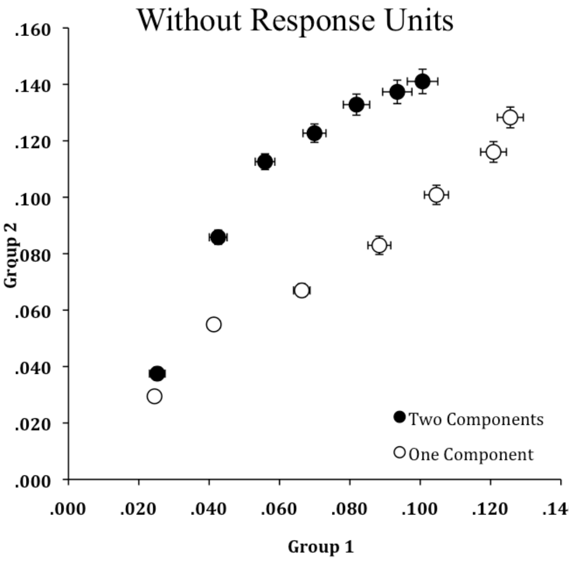

Figure 4. State-trace plot showing learning of Model 3 (one connection weight component) and Model 4 (two connection weight components), neither of which have response units, across training.

Discussion. Both in models with and without response units, multi-dimensional state-trace plots are produced when comparing those with one or two connection weight components. The state-trace analysis suggests that the two models are driven by different underlying processes, which in this case is due to the presence or absence of fast weights. Following the state-trace logic, this suggests that the two weight components within an AugSRN should be considered as distinct, different learning processes.

State-Trace Analysis 3: Response units

Does the addition of response units to the basic SRN, or an SRN with two connection weight components, produce separate functions on the state-trace plot? The same four models are presented below, comparing Models 1 (no response units) and 3 (with response units), which both have one component, and Models 2 (no response units) and 4 (with response units), which both have two connection weight components.

Results. We compare Models depending on whether they have response units or not, coded as 1 and 0, respectively. See Figures 5 and 6 for state-trace plots of one and two connection weight component models. We find that in models with one component, adding Model as a variable significantly improves the regression model, the R^2_{adj} value improves from 92.3% to 95.6%, ΔR^2 : $F(1,11) = 10.3$, $p = .008$, and overall, the model had a significant fit, $F(2,11) = 143.8$, $p < .001$; Group 2 = .942xGroup 1 + .013xModel Type + .006.

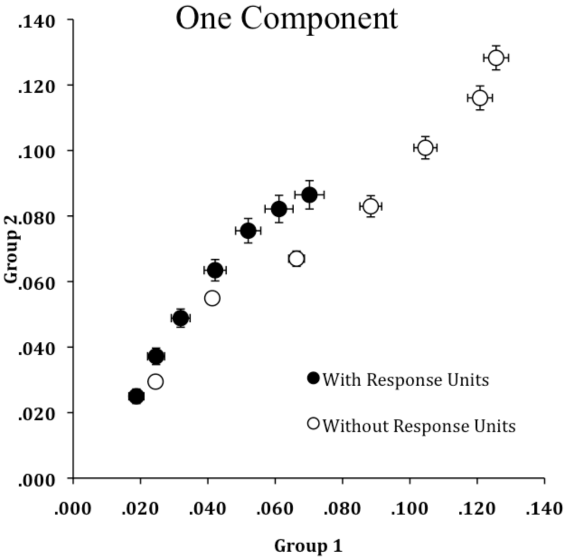

Figure 5. State-trace plot showing learning of Model 1 (with response units) and Model 3 (no response units), which both have only one connection weight component, across training blocks of simulations.

Comparing Models 2 and 4, with response units, the regression does not significantly improve when adding Model as a variable into the regression.

Discussion. Whilst the functions are not as distinct as in State-Trace Analysis 2, the comparison of models with and without response units still suggests a multi-dimensional structure. That the models with two connection weight components failed to reach significance is perhaps more a criticism of the linear regression method when analyzing these data. The fact that the separation of the two plots is less impressive (in size and reliability) when the presence (or not) of the response units is the manipulation than when the use of one vs. two sets of weights also suggests that for the type of model considered here, the main difference between the AugSRN and the standard SRN is the distinction between fast and slow weights.

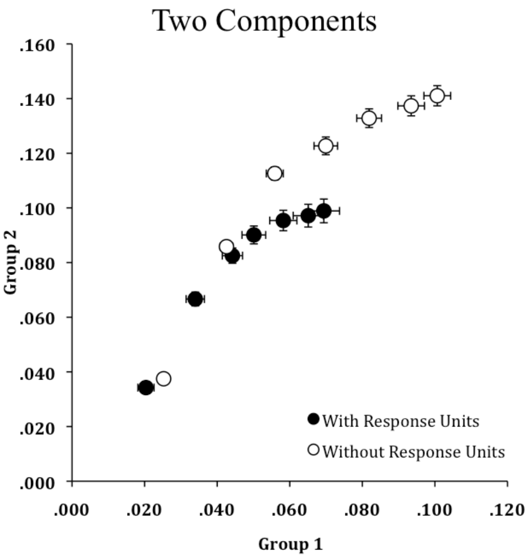

Figure 6. State-trace plot showing learning of Model 2 (with response units) and Model 4 (no response units), which both have two connection weight components, across training blocks of simulations.

State-Trace Analysis 4: Learning Rates

Finally, to ensure that the differences seen in State-Trace Analysis 2 between one and two component models were not simply a result of the total amount or rate of learning, we varied the learning rates of the one component model, keeping the hidden units set at 20. We set the learning rate to 0.933, equal to the sum of the fast and slow learning rates employed in the AugSRN simulations to plot alongside the earlier one process model with a learning rate of 0.4. We are aware that a one component SRN with a learning-rate equal to the sum of two component's learning-rates is not a direct equivalent. Nevertheless, our manipulation should allow us to discover if varying learning rate over this range produces different state-trace plots.

Results. An SRN with a Learning Rate of 0.933 learns the task, $F(1,124) = 556.8$, $p < .001$. The state-trace plot of these data, alongside the original Learning Rate of 0.4 can be seen in Figure 7, which clearly shows two functions. When adding the learning rate as a regressor into a model predicting Group 2 performance from Group 1 performance, the R^2_{adj} value improves from 75.8% to 92.4%, ΔR^2 : $F(1,11) = 27.2$, $p < .001$, and this model overall had a significant fit, $F(2,11) = 79.9$, $p < .001$; Group 2 = .895xGroup 1 + .048xLearn Rate - .010.

Discussion. The state-trace plot and the regression analysis clearly demonstrate two separate functions, which according to state-trace analysis suggests the presence of multiple processes. However, the two models differ only in the values assigned to their learning rates. State-trace analysis proposes that a multi-dimensional state-trace plot will result from the presence of multiple processes in a given dataset, which implies the influence of more than one than one latent variable within the SRN on performance. This suggests that either state-trace analysis is sensitive to differences within a single process, or alternatively the SRN must be considered a multi-process model of learning.

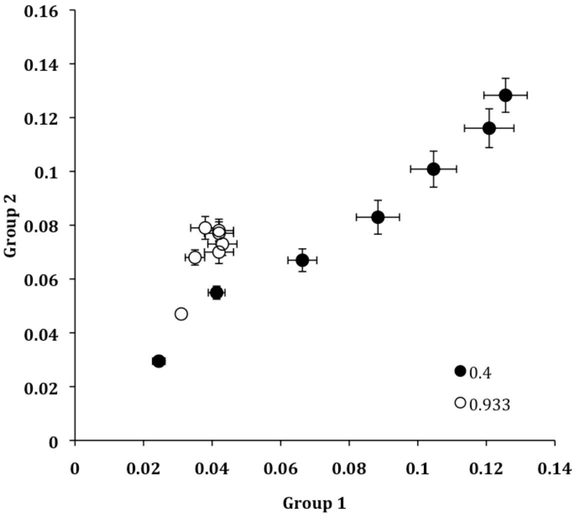

Figure 7 State-trace plot for an SRN with different learning rates (shown on graph).

General Discussion

The analyses of SRNs with one or two learning components, and those with or without response units give separate functions on state-trace plots, suggesting the presence of more than one latent psychological variable. However, simple variation of the rate at which the SRN learnt also produced a multi-dimensional state-trace plot, which raises questions for the interpretation of multi-dimensional state-trace plots. A parameter search, varying learning rates and number of hidden units, has been conducted and, within a reasonable range for the SRN on this SRT task, produces the same functions as the data presented above. We recognise that the regression method employed in analysing the data is limited to roughly linear functions, and suggest that other methods (e.g. Newell, Dunn & Kalish, 2010; Prince, Brown & Heathcote, 2011) are also explored. But our analyses are, if anything, insensitive to the differences visualised by the plots, so this does not compromise our conclusions

It seems, then, that not only multiple processes, but variations within a single process can produce multi-dimensional state-trace plots. The implications for state-trace analysis as a tool for the investigation of the number of latent variables underlying human behaviour needs to be considered, and further analysis of computational models with this technique is recommended.

Acknowledgments

The research reported in this paper was supported by a Postgraduate studentship and Exeter Graduate Fellowship awarded to Fayme Yeates, and an ESRC grant awarded to Ian McLaren and Fergal Jones.

REFERENCES

Anastasopoulou, T., & Harvey, N. (1999). Assessing sequential knowledge through performance measures: The influence of short-term sequential effects. *Quarterly Journal of Experimental Psychology, 52A,* 423-448.

Bamber, D. (1979). State-trace analysis: A method of testing simple theories of causation. *Journal of Mathematical Psychology, 19,* 171-181.

Cleeremans, A. (1993). *Mechanisms of implicit learning: Connectionist models of sequence processing.* Cambridge, MA: The MIT Press.

Cleeremans, A., & McClelland, J. L. (1991). Learning the structure of event sequences. *Journal of Experimental Psychology: General, 120,* 235-253.

Dunn, J.C. (2008). The dimensionality of the remember-know task: A state-trace analysis. *Psychological Review, 115(2),* 426-446.

Elman, J. L. (1990). Finding structure in time. *Cognitive Science, 14,* 179-211.

Jones, F. W., & McLaren, I. P. L. (2009). Human sequence learning under incidental and intentional conditions. *Journal of Experimental Psychology: Animal Behavior Processes, 35(4),* 538-553.

Kinder, A., & Shanks, D.R. (2001). Amnesia and the declarative/non-declarative distinction: A recurrent network model of classification, recognition, and repetition primining. *Journal of Cognitive Neuroscience, 13,* 648-669.

Lewicki, P., Czyzewska, M., & Hoffman, H. (1987). Unconscious acquisition of complex procedural knowledge. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 13,* 523-530.

Loftus, G.R., Oberg, M.A., & Dillon, A.M. (2004). Linear theory, dimensional theory, and the face-inversion effect. *Psychological Review, 111(4),* 835-863.

McClelland, J.L., & Rumelhart, D.E. (1986). Amnesia and distributed memory. In J. L. McClelland & D. E. Rumelhart (Eds.), *Parallel distributed processing. Explorations in the microstructure of cognition: Psychological and biological models* (Vol. 2, pp. 503-527). Cambridge, MA: MIT Press.

Newell, B.R., & Dunn, J.C. (2008). Dimensions in data: Testing psychological models using state-trace analysis. *Trends in Cognitive Science, 12(8),* 285-290.

Newell, B.R., Dunn, J.C., & Kalish, M. (2010). The dimensionality of perceptual category learning: A state-trace analysis. *Memory & Cognition, 38(5),* 563-581.

Nissen, M. J., & Bullemer, P. (1987). Attentional requirements of learning: Evidence from performance measures. *Cognitive Psychology, 19,* 1-32.

Prince, M., Brown, S., & Heathcote, A. (2011, October 31). The Design and Analysis of State-Trace Experiments. *Psychological Methods.* Advance online publication. doi: 10.1037/a0025809

Remington, R. J. (1969). Analysis of sequential effects in choice reaction times. *Journal of Experimental Psychology, 82,* 250-257.

Rumelhart, D.E., Hinton, G.E., & Williams, R.J. (1986). Learning internal representations by error propagation. In D. E. Rumelhart & J. L. McClelland (Eds.). *Parallel distributed processing* (Vol. 1, pp. 318-362). Cambridge, MA: Bradford Books.

Yeates, F., Jones, F. W., Wills, A. J., McLaren, R., & McLaren, I. P. L. (submitted). Modelling human sequence learning under incidental conditions.