Canadian Journal of Experimental Psychology / Revue canadienne de psychologie expérimentale
2012, Vol. 66, No. 2, 90–97

1196-1961/12/$12.00 DOI: 10.1037/a0026639

Item Order Matters in a Function Learning Task

Peter J. Kwantes
Defence Research & Development Canada;
The University of Queensland

Andrew Neal
The University of Queensland

Michael Kalish
University of Louisiana at Lafayette

In a function learning task, participants are taught the relationship between 2 variables, a predictor (e.g., the dosage of a drug) and a criterion (e.g., its effect on mood). Of particular interest in this article is the question of what information does a participant use to generate a response for test examples that fall outside the training region—so-called, extrapolation items. In this article, we test whether the presentation of training items has an impact on the pattern of responses for items requiring participants to extrapolate, and examine, whether the 2 dominant accounts of function learning (Population of Linear Experts [POLE]: Kalish, Lewandowsky, & Kruschke, 2004; and Extrapolation Association Model [EXAM]: DeLosh, Busemeyer, & McDaniel, 1997) can account for this effect. The results show that a manipulation of trial-to-trial changes in the relative magnitudes of the predictor and criterion does influence subsequent extrapolation, and neither POLE, nor EXAM, was able to account for this effect in their current forms. We demonstrate that a model that encodes information about the trial-to-trial changes in the predictor and criterion, and which subsequently uses this information to adjust the retrieved value of the criterion, can account for the effect.

Keywords: function learning, memory, model, order

Functions are mathematical relations among continuous variables for which magnitudes on one or more predictor variables, or *x*'s, are used to determine a magnitude on a criterion, or output variable often referred to as the *y*. In a typical experimental task examining how people learn functions, participants learn the relationship between two continuous variables by being presented with pairs of values that represent the predictor and criterion. For example, participants may be told that their task is to learn the relationship between the dosage of a fictitious drug and its effect on a patient's mood. During training, they are shown dosages paired with measures of mood. Over several exposures to the pairs, participants form a mental representation of the relationship between the two variables.

The initial learning phase is followed by a testing phase in which participants are given values of *x* and asked, without help or feedback, to estimate what *y* might be. Performance is assessed by presenting items that test the participant's ability to *interpolate* and *extrapolate*. That is, participants are asked to generate estimates for *y* when presented with new *x*-values that lie within the range of *x*-values encountered during training (so-called *interpolation items*) and values that lie outside the training range (so-called *extrapolation* items).

In the current article, we examine the mechanisms that enable participants to extrapolate for values of *x* that fall outside the training region. Computational accounts of function learning are currently dominated by two frameworks that we shall refer to as, *mixture-of-experts* and *exemplar-plus-rule* models. Both frameworks explain function learning in terms of a process whereby training examples are committed to memory as values of *x* associated with a value of *y*. What differentiates the accounts is how *x* and *y* are associated. In the section that follows, we will make the distinction clearer by describing each of two models that represent the two frameworks in more detail.

Current Models of Function Learning

EXAM: An Exemplar-Plus-Rule Model

DeLosh, Busemeyer, and McDaniel's (1997) Extrapolation Association Model (EXAM) is based on well-known models of categorisation (Kruschke, 1992; Nosofsky, 1986; Nosofsky & Palmeri, 1998) and as such, has a successful pedigree as a framework for understanding how people learn concepts—be they categories or functions. EXAM learns and stores the association between pairs of *x* and *y* values encountered during training. At test, when a new value of *x* is presented to the model the closest matching *x*-value and its associated *y* is retrieved from memory.

Peter J. Kwantes, Defence Research & Development Canada and School of Psychology, University of Queensland; Andrew Neal, School of Psychology, University of Queensland; Michael Kalish, Institute of Cognitive Science, University of Louisiana at Lafayette.

This research was supported by two Small Grants from the Australian Research Council to the authors (96/ARCS205G and 2000100389) and by Applied Research Project (16br) from Defence Research & Development Canada. We thank Mike Humphreys and Stephan Lewandowsky for useful suggestions throughout the course of the modeling work and earlier drafts of this article.

Correspondence concerning this article should be addressed to Peter Kwantes, Defence Research & Development Canada - Toronto, 1133 Sheppard Avenue West, P.O. Box, 2000, Toronto, Ontario, Canada, M3M 3B9. E-mail: peter.kwantes@drdc-rddc.gc.ca

EXAM follows retrieval with an adjustment on the estimate of y. The magnitude and direction of the adjustment on y is proportional to the discrepancy between the test value of x and the x value of the closest matching example in memory. If the test value, X_P, is most closely matched to the example, X_m, EXAM computes an estimate of y by adjusting the retrieved mean response Y_m by an amount equal to $(X_p - X_m)$ weighted by the slope of the line connecting the examples that neighbor X_m. So, when X_m is retrieved, the slope of the line connecting (X_{m+1}, Y_{m+1}) and (X_{m-1}, Y_{m-1}) is used to guide the adjustment of Y_m. In the case of extrapolation items where either X_{m+1} or X_{m-1} does not exist, X_m is used in its place. The method that EXAM uses to adjust y ensures that, like humans, estimates extend in a near linear fashion into the extrapolation region for values of x that fall outside the domain defined by the training set. The formula used by EXAM is expressed as follows:

$$Y_{final} = \sum_{m=1}^{M} P(X_m|X)E(Y|X_m) \quad (1)$$

where

$$E(Y|X_m) = Y_m + \left[\frac{Y_{(m-1)} - Y_{(m+1)}}{X_{(m-1)} - X_{(m+1)}}\right](X_P - X_m) \quad (2)$$

and $P(X_m|X)$ is the probability of choosing exemplar X_m as the match to stimulus X.

POLE: A Mixture-of-Experts Model

Unlike instance-plus-rule models which learn the association between x and y values for each training item, mixture-of-experts models like POLE (Kalish et al, 2004) learn the association between each x and the mental representation of a linear function (a so-called, *expert*) that yields the correct response for y. When estimating y, the test value of x, activates the closest matching x-value in memory and its associated expert.

When POLE learns a linear function, it is possible that only one expert is needed for a correct response across the training items. For other, nonlinear functions, estimates at different points on the function are generated from the output of different experts with slopes that yield the right response. An estimate for y is determined by applying the test value of x to the x-nodes in memory and reporting the output of the expert associated with the closest match.

To summarise, both EXAM and POLE are exemplar models of function learning that use two very different methods for generating responses to test items requiring extrapolation. EXAM uses the relationship between neighboring exemplars to determine the direction and extent of an adjustment that must be imposed on a retrieved value of y. POLE requires no such adjustments on y because the response is generated by an expert that has been retrieved from memory.

The Goal of the Current Article

The goal of the current article is to evaluate the hypothesis that participants encode relative changes in magnitude from trial to trial during training, and subsequently use the information as a basis for extrapolation. Neither EXAM nor POLE encodes relative changes in magnitude from trial to trial during training. Information regarding the slope of the function is either calculated at retrieval (EXAM), or derived from prior knowledge (POLE). The argument that participants are sensitive to trial to trial changes in magnitude is consistent with sequence effects reported in the categorisation/classification literature—a literature that represents the theoretical backdrop for current models of function learning. Several categorisation articles (e.g., Jones, Love & Maddox, 2006; Stewart, Brown, & Chater, 2002; Zotov, Jones & Mewhort, 2010) report data in which the response on a given trial affects performance on the next trial. For example, Zotov et al. (2010) demonstrated that when participants were asked to generate circles belonging to particular size category, the size they produced tended to be biased away from the sizes relevant to the circles in the category of the previous trial.

We report an experiment that tests whether trial to trial changes in magnitude affect extrapolation performance by manipulating the relative changes in x and y on consecutive trials during training. We achieved this by using a quadratic function, and varying the order of presentation of the training examples. In one group, the training examples were ordered so that the changes in y from trial to trial were relatively large in relation to the changes in x. The other group was given the same set of training items. However, their order was modified, so the change in y from trial to trial was relatively small in relation to the change in x. We hypothesised that the manipulation of slopes during training would influence extrapolation. Specifically, we predicted that the group that was exposed to steep trial to trial slopes during training should have a steeper extrapolation function than the group that was exposed to shallow slopes. To anticipate, the results show that the manipulation of slopes during training does influence extrapolation. We demonstrate that EXAM and POLE, in their current form, cannot account for this effect, and show that a model that encodes trial to trial changes in relative magnitude during training can.

Method

Participants

Thirty first-year students at the University of Queensland participated for course credit.

Materials

The experiment used 20 training and 45 test items taken from DeLosh et al. (1997). The items are listed in the Appendix. The training function was the quadratic function used by DeLosh et al., $y = 210 - (x - 50)^2/12$. The order of the 45 test items was randomized uniquely for each participant. Two classes of training materials were constructed: one with "steep" slopes associated with the transition from one item to the next during training, and another with "shallow" slopes. The two groups were constructed in the following way. First, we calculated the slopes of all possible item-to-item transitions for the training function. Then, we took the mean of the absolute value of the slopes, $M = 1.292$, as a criterion to decide whether a transition slope was high, or low.

The next step was to generate 30 separate stimulus files—one for each participant. Participants were to be presented with each of the 20 training examples 10 times. The 20 training examples would be presented in 10 contiguous 20-item blocks. For each file, we started by selecting the first training example at random. If the file under construction was meant for membership in the "steep" condition, the second training example was selected at random, and the slope relating it to the first item was checked. If the slope exceeded the criterion, it was retained, if not it was discarded and a new candidate item was selected. Stimuli for the shallow condition were selected similarly, except that each sampled item was maintained if the slope relating it to the previous item was less than the criterion. In either condition, after 10 failed attempts to select a suitable item, the check was deferred to the next sampled item. The end result of the process was a set of items for which the majority of transitions from one training item to the next had either steep slopes with a mean absolute value of 1.91 or shallow slopes with a mean absolute value of 0.65.

Procedure

Participants were tested individually. Half of the participants were tested in the "steep" condition, and half were tested in the "shallow" condition. The procedure used in the experiment was almost identical to the one used by DeLosh et al. (1997) and Kwantes and Neal (2006). There was a training phase and a test phase. On each trial during training, participants were shown three horizontal number lines on the computer monitor. Each number line had tick marks starting at zero on the left side with value labels every 10 units. The top bar contained values from 0 to 100. The other two contained values from 0 to 280. On each trial, a marker, shown as a narrow vertical bar, was placed at a location on the top bar (labelled, "substance") to point to the value of a dosage. The dosage, written as a number, was also displayed above the number line. Using the computer's mouse, the participant then moved a similar marker on the second number line (labelled 'Predicted level of mood') to a position indicating the estimate of mood associated with the dosage. When the participant released the marker, the estimate, written as a number, was displayed above the number line.

After the participant made an estimate, it was locked in by clicking on a button labelled, "submit your answer." Upon locking in, participants were given three forms of feedback regarding their performance. First, the correct level of mood was presented on the bottom number line, which was labelled "actual level of mood." The correct answer, written as a number, was also displayed above the number line. Participants were also shown a difference score between their response and the correct answer. When training was finished, participants were tested on 45 new dosages that ranged in value from 1 to 100 units. The procedure was identical to the one used during the training except that the bottom number line was removed and participants were not given any feedback regarding their performance and that the order in which they were presented was randomized uniquely for each participant.

Results and Discussion

Training Items

We used unsigned error as the dependent variable, calculated as the absolute value of the difference between the subjects' estimated rate of spread and the correct answer. The learning curves for both conditions are shown in Figure 1. As is clear in the figure, subjects' estimates for the criterion improved with practice, the rate of improvement decreased with practice, and importantly, by the end of training, performance in the two groups was nearly identical.

Test Items

Six responses out of the total 1,350 responses were lost due to hardware failures. These cells were left empty and did not contribute to subsequent analyses. Participants' responses had, on average, a strong positive correlation with the correct responses ($M = 0.7$, $SD = .26$). Inspection of individual participants' response curves revealed two outliers. These participants' correlations were more than three standard deviations below the mean with values of $r = –.16$ and $r = –.22$, respectively. These participants were also the only two whose predictions were negatively correlated with the correct function and were excluded on the basis that they either clearly did not learn the relationship between x and y, or they failed to follow instructions.

Of particular interest for this experiment is participants' performance for items in the areas outside the training region—the extrapolation items. Participants' data are plotted in the left panel of Figure 2. For each participant we ran regressions to calculate the slopes of the y-estimates in the upper and lower extrapolation regions. In the upper extrapolation region (where x was greater than 70), the figure shows a clear and reliable tendency for the slope of the estimates in the "steep" condition ($M = –2.15$, $SD = 2.2$) to be more extreme than for that of the "shallow" condition ($M = –0.59$, $SD = 1.7$), $t(26) = 2.414$, $p = .012$. The pattern is consistent with the argument that participants tracked the changes in y relative to x during training, and that this information influenced participants' estimates for extrapolation items. The pattern of estimates in the upper extrapolation region is also consistent with the notion that participants found the materials in the shallow condition more difficult to learn than the items in the steep condition. As is clear in Figure 1, however, participants reached the same level of mastery over the training items in both conditions. Therefore, an explanation based on how well the materials were learned is unlikely.

In the lower extrapolation region (where x was less than 30), the slopes for estimates in the shallow and steep conditions were homogenized. We found no reliable difference between the slopes for estimates in the shallow ($M = 2.81$, $SD = 1.8$) and steep ($M = 2.79$,

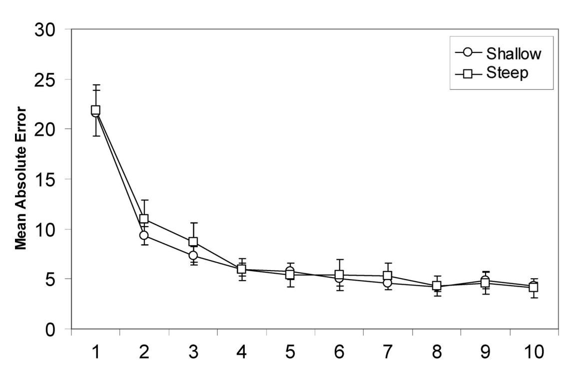

Figure 1. Learning curves with *SE* bars for the Shallow and Steep conditions.

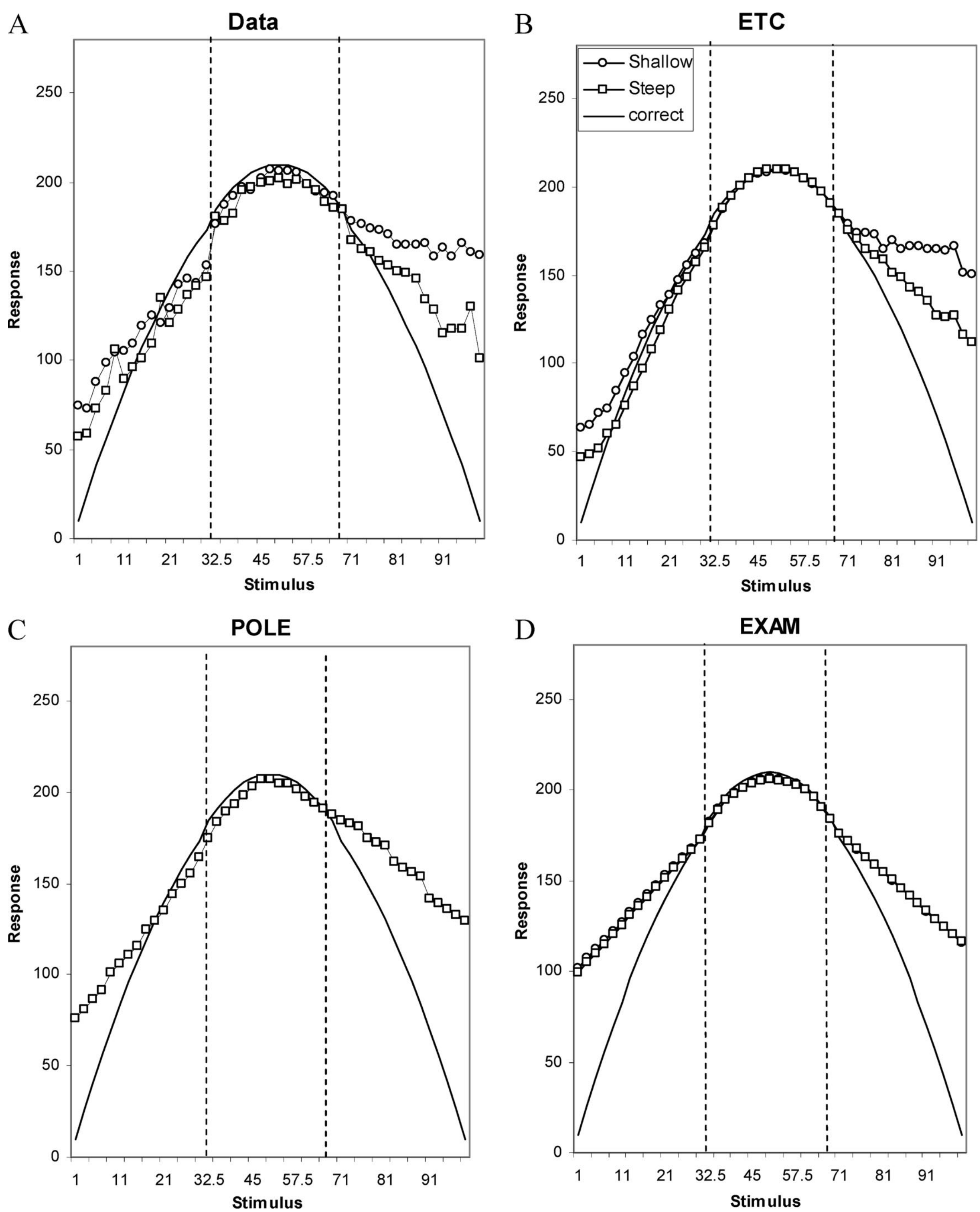

Figure 2. Participants' (A) and the models' (Panels B, C, D) estimates for the test items. The dashed vertical lines denote the range of x values that were used during training.

SD = 2.0) conditions. We suspect that the homogenization occurred because of participants' tendency to anchor responses for the criterion at zero for low test values of x.

In summary, the manipulation of slopes during training influenced extrapolation in the upper extrapolation region. This effect is consistent with the notion that participants encode trial to trial changes during training, and subsequently use this information as a basis for extrapolation.

Simulations

The steep and shallow conditions differed only in the order of presentation of trials. Current models of category learning are sensitive to trial order, and so may be able to account for the observed difference between the mean transfer responses of the two groups. As a test of this possibility, we fit both EXAM (DeLosh et al., 1997) and POLE (Kalish et al., 2004) to the data from our experiment. In order to fit EXAM and POLE to these data, we took some steps to enable the models to account for the observed anchoring at zero. For both models we rescaled both stimuli and possible responses to fall in the range (0,1). For EXAM, we provided a weak initial weight matrix that encodes the function $y = x$. For POLE, we provided biases over the set of experts to allow both $y = x$ and $y = 1-x$ to compete with local slope information during training. These measures are similar to those taken by Kalish et al. (2004) and are important for aligning model predictions with pre-experimental biases participants bring to the task.

EXAM and POLE received the same stimulus/feedback orders as the experimental participants. Because none of the parameters in either EXAM or POLE are designed to vary in response to the order of the training items, we reasoned that allowing the parameters to vary across groups to capture our results would be an unfair test. As a result, we constrained each model to use only one set of parameters.

Model fit was assessed only on the transfer trials. The correlations between the participant data and each model in each condition, as well as the root-mean-square error (RMSE) are shown in Table 1. The results were striking. While both EXAM and POLE are able to approximate the transfer results fairly closely, neither model was able to predict a substantial between-groups difference on the basis of trial order alone (see Panels C and D of Figure 2). For both models, the predicted mean responses are within 1% of each other through the entire range of test stimuli.

One way that the order of training items might affect function learning would be to assume that in addition to encoding the magnitudes of x and y during training, participants encode how y changes relative to x from one trial to the next as well. The information regarding change is then used to guide the adjustments on retrieved values of y to generate estimates for the criterion.

In what follows, we carry out a computational exercise in which we test a framework that encodes trial to trial changes in the variables of the function it is learning. We should note that what we will describe cannot be considered a proper model. The framework is highly simplified and simply meant to show that the basic idea of tracking trial to trial change could explain our data. We call the approach, *E*xtrapolation from *T*racked *C*hange (ETC). Despite using a different formalism from a model like EXAM, we had ETC perform the function-learning task in essentially the same way—training examples are retrieved from memory in response to a probe item, and adjustments are made on the retrieved value of y in response to the degree of disparity between the probe and best-matching item. The important *process* difference between our approach and the other models is the notion that, in addition to encoding the $x - y$ pairs, a memory trace for a pair includes information about how y changes from one trial to the next with respect to the change in x.

We conceptualised ETC as a multiple-trace memory architecture akin to Hintzman's (1984, 1986, 1988) exemplar model of memory, Minerva2. ETC represents each training trial as a memory trace. A memory trace contains fields; each of which contains a representation of a magnitude. The first contains a representation of the predictor x, the next contains the criterion value, y, and the last contains the change in y relative to x on the previously encoded trial (we denote the change with the symbol, Δ). The value of Δ contained within trace i is calculated by the following formula:

$$\Delta_i = \frac{Y_i - Y_{i-1}}{X_i - X_{i-1}} \tag{3}$$

Note that the calculation of Δ necessarily requires the existence of a previous trial. Consequently, there is no representation for Δ for the first trial (i.e., when $i = 1$). As well, there cannot be a representation for Δ when the values of x for trials i and $i - 1$ are the same. In either event, the field of the trace corresponding to Δ is set to 0.

Once memory is populated with training items, ETC is tested by letting a predictor resonate with, or activate, the contents of memory. The extent to which a memory trace is activated by the probe (X_P) is a function of the similarity of the two. We assume that proximal magnitudes share similar representations, and that the similarity between the representations decreases as the distance between them increases. DeLosh et al. (1997) and Kalish et al. (2004) deal with the similarity by implementing an activation gradient across a set of nodes that represent magnitudes. In our illustration here, we use the same formula suggested by DeLosh et al.'s EXAM model:

$$a_i(X_P) = e^{-\lambda(X_P - X_{T_i})^2} \tag{4}$$

Once activated, the system selects a single instance from memory as a match to the probe. The probability of selecting one memory trace is a function of the trace's activation. Specifically, the probability (p) of selecting trace i (denoted T_i) is equal to the activation of a trace divided by the sum of the activations of the j traces in memory (M). More formally,

$$p(T_i) = \frac{a_i(X_P)}{\sum_{j=1}^{M} a_j} \tag{5}$$

Like EXAM, if the probe values fall outside of the bounds set by the training items, the best match from memory will be one of the traces from an extreme end of the training range. Also similar to EXAM, we allow the retrieved value of y (denoted, Y_m) to be adjusted to a degree that reflects the disparity between X_P and the value of the predictor it retrieved from memory, X_m. Whereas EXAM's adjustment is weighted by the slope connecting the training examples that neighbor the best-matching example, we adjust Y_m on the basis of the slope associated with the example retrieved from memory, which we denote, Δ_m. That is, the adjustment is weighted by the slope connecting the retrieved item, and its

Table 1
Measures of Agreement Between the Models and Participants

	Slope	R(model, data)	RMSE
ETC	Shallow	0.99	8.25
	Steep	0.98	11.71
	Difference	0.84	—
POLE	Shallow	0.97	9.75
	Steep	0.98	14.60
	Difference	0.41	—
EXAM	Shallow	0.85	19.30
	Steep	0.96	19.34
	Difference	–0.33	—

relationship to the item that preceded it during training. We calculate the adjusted value of y (denoted as Y_{adj}) using an equation that is almost identical to Equation 2 above:

$$Y_{adj} = Y_m + \Delta_m(X_P - X_m) \quad (6)$$

Kwantes and Neal (2006) showed that estimates for test items in the lower extrapolation region tended to deviate from the true function, being deflected toward zero. They argued that this effect, which they termed "anchoring at zero," reflects the influence of participants' background knowledge or beliefs about functions. In many contexts, a value of 0 in the predictor is paired with a 0 in the criterion. For example, the speed of a sailboat will be 0 knots when the wind speed is 0 km/hr. Kwantes and Neal argued that the tendency reflects a form of *base-rate neglect* (Kahneman & Tversky, 1973), a phenomenon typically applied to findings in probability judgment tasks wherein subjects, to some varying extents, ignore prior probabilities when making decisions. Kwantes and Neal proposed that in a function-learning task, when x approaches zero, participants will be biased to ignore memory information about the relationship and favour an estimate based on the simplest possible relationship, that is, that y is also close to 0. We address this phenomenon by imposing a transformation on Y_{adj} to create a new estimate of y, called Y_{final}. We added a parameter β (which we set to 0.6) corresponding to the probability that the subject will draw on such background information about functions when generating a response. The degree to which a 0 response for y contributes to an estimate is a function of the similarity between the test value of x and an x of 0. More formally, we adjusted the model's estimate of y in the following manner:

$$Y_{final} = Y_{adj} \times [1 - \beta e^{-\lambda(X_P\tau)^2}] \quad (7)$$

As in the equation we use to calculate activations, the degree to which an item's response is anchored at zero will vary according to how strongly the item activates a ($x = 0$, $y = 0$) item in memory. The τ, set to 0.15, is a scaling parameter that determines the gradient by which the cue's similarity to 0 shapes the estimate.

To simulate the experiment above, we ran the same input files seen by our participants through the ETC framework. Panel B of Figure 2 shows the model's performance on the test items. Table 1 shows the correlation between the model's responses and the data as well as the RMSE separately for the steep and shallow conditions. Note that, like our participants, ETC exhibits a steeper slope for estimates in the steep condition than the shallow condition for the upper extrapolation region, but not the lower extrapolation region. In the lower extrapolation region, the slopes for the estimates in the two conditions become homogenized.

One aspect of the simulation data worth mentioning is that, upon inspection in Figure 2, although our approach captures the data better then its competitors, the correlation and RMSE associated with ETC in Table 1 does not appear to differ appreciably from those of the competing accounts. Each model produces estimates for y in the right range that follow the correct general inverted-U pattern, so neither measure provides a clear indication of the superiority of one approach over the other. To address the issue, we measured the difference between the two y estimates (one from each condition) for the same test value of x (which we refer to as *Difference* in Table 1) across the range of 45 x values tested in the experiment. We calculated the difference scores for participants and each simulation. Then, we correlated the participants' difference scores with those of each simulation. The values are shown in Table 1. As is clear in the table, the pattern of differences between estimates in the shallow and steep conditions across values of x is captured significantly better by ETC ($r = .84$) than either POLE ($r = .41$, $Z = 3.6$, $p = .0002$) or EXAM ($r = -.33$, $Z = 7.17$, $p \cong 0$).

General Discussion

In this article, we introduced another way in which an exemplar model of function learning could generate estimates for extrapolation items. Unlike competing accounts of function learning, learning a new functional relationship might involve encoding how the criterion changes from one training trial to the next as the predictor changes. Our participant data support the basic idea. When we control the magnitude of the slope relating each training example to the one preceding it during the learning phase, it influenced the slope of participants' extrapolation responses.

Our data present a challenge for models of function learning like EXAM and POLE. In their current form, neither EXAM nor POLE can account for these data, because they neither encode nor use trial to trial changes in magnitude. POLE stores the associations between x-values and experts that generate the correct response on y. To account for our data, one possibility might be to introduce a parameter that, during training, biases the system to associate x with an expert on trial i that has a similar slope to the expert associated with trial $i - 1$. Done this way, the shallow and steep extrapolation slopes exhibited by participants might occur because during training, the items were associated mainly with experts possessing shallow and steep slopes, respectively.

With respect to EXAM, one could argue that the framework we presented here represents an adjustment on that model. Like EXAM, training examples are stored in memory as x-y associates and like EXAM, its estimate for y is generated by retrieving an exemplar from memory and applying a rule. In fact, the equation we use to adjust y is almost identical to that used by EXAM. The only real conceptual difference between the two models is the source of the slope information that is used to drive the adjustment on y for new items.

Stewart, Brown, and Chater (2002) demonstrated that a simple model that categorizes one-dimensional stimuli strictly on the basis of its change in magnitude from one trial to the next can obtain a high degree of accuracy. The model they favoured to explain categorisation performance was one in which the subject judges category membership on the basis of a stimulus' magnitude and the relative change in the stimulus from the previous to the present one. Clearly, Stewart et al.'s account is similar to ours, and provides one more aspect in which models of function learning and categorisation can be seen as having a close kinship. Given their kinship in the modelling work then, it will be interesting to determine the extent of the similarity between the processes involved in function learning and category learning with respect to sequential effects.

Recently McDaniel, Dimperio, Griefo, and Busemeyer (2009) also reported sequential effects in learning functional relationships. They found that participants learned functions more quickly when the predictor values of the training items were sequentially increasing than when they were presented in random order. The finding is consistent with an account based on tracked changes in x and y. When items are presented in sequential order, a given item will always have the same associated slope because it is always preceded by the same item. When the order of the materials is random, a given item will have a range of different slopes associated with it, thereby increasing the variability associated with extrapolation judgments. We realise that our treatment of McDaniel's et al.'s findings is cursory; however, it and the data presented here represent a starting point for determining how order information contributes to the mental representations necessary for learning functional relationships.

It should also be noted that the experimental and modelling work reported here is limited to dealing with functions that are noncyclic in nature. Bott and Heit (2004) explored the mechanisms involved in learning nonmonotonic functions. In that article, they showed that previously published models of function learning do not deal well with cyclic functions. However, they also showed that adding a second component to deal with the cyclic component of a function allowed their exemplar-plus-rule model to yield y estimates that followed the sinusoidal pattern of the training function. Although we did not deal with cyclic functions in the work reported here, we suspect that our modelling work could be similarly extended to cyclic functions because they are derived from the same classic models.

Finally, it is worth discussing some implications that the modelling work here has for the role of memory in function learning. The modelling framework we used to clarify our thinking about how people learn functions is similar to a well-known model of episodic memory, Hintzman's (1984, 1986, 1988) Minerva2. A fair question that could be asked is whether adopting an episodic memory model entails the assumption that participants *recollect* y and its associated Δ when cued with x. We do not think it does. We postulate that functional relationships are learned implicitly in a manner much like that described in the classic work by Reber (1967). Although Minerva2 was proposed as a model of episodic memory, using the basic architecture to model our data does not commit us to the notion that memory retrieval invoked for learned functions is episodic in the traditional sense proposed by Tulving (1972). The same basic memory system has been used by others to explore retrieval from memory for so-called "implicit learning". For example, Jamieson and Mewhort (2009a, 2009b, 2010) provide several examples of how a model like Minerva2 can be used to explain the apparent implicit learning of an artificial grammar. The lesson we take from the modelling work is that the applicability of a memory model as an aid to understand memory processes is not bound by its originally intended use—it can help us to understand operations that are traditionally viewed as coming from different memory systems.

The simulation of our results also point to the role that the structure of information in memory plays on performance. Simon (1996) argued that much of complex behaviour reflects a simple mechanism's sensitivity to the regularities in a complex information environment. Simon's basic argument has since been applied to the study of reading (Kwantes & Mewhort, 1999) and models of lexical semantics (Jones & Mewhort, 2007; Kwantes, 2005). We make a similar point here for function learning. Our framework captures the data because of how the training materials were encoded, not because the steep and shallow conditions required different parameterizations. Sensitivity to the order of the training materials reflects sensitivity to the structure of the information that has been learned.

The data reported here provide some potential insight as to where participants get the information they need to adjust their responses to test items that require extrapolation. The function-learning literature has its theoretical basis in models of categorisation. It is therefore not surprising that some sequential effects one observes in a categorisation task have counterparts in function learning. We believe that the data and modelling demonstration we provided here, represent a starting point for determining the extent to which the learner's prior experience with the training material during learning helps shape the formation of conceptual knowledge.

Résumé

Dans une tâche d'apprentissage de fonction, les participants se font enseigner la relation entre deux variables (par ex., le dosage d'une drogue) et un critère (par ex., son effet sur l'humeur). Dans le présent article, la question à savoir quelle information est utilisée par le participant pour générer une réponse dans des exemples de tests sortant du cadre de l'entraînement – appelés items d'extrapolation – revêt un intérêt particulier. Dans cet article, nous testons si la présentation des items d'entraînement influence sur le profil de réponses pour les items où les participants doivent extrapoler et nous examinons si les deux interprétations dominantes de l'apprentissage de fonction (Population of Linear Experts [POLE]: Kalish, Lewandowsky, & Kruschke, 2004; and Extrapolation Association Model [EXAM]: DeLosh, Busemeyer, & McDaniel, 1997) peuvent expliquer cet effet. Les résultats montrent qu'une manipulation d'essai en essai des changements de magnitude relative du prédicteur et du critère n'influence pas l'extrapolation subséquente et ni POLE ou EXAM ne permet d'expliquer ces effets dans leur forme actuelle. Nous démontrons qu'un modèle qui encode l'information à propos des changements d'essai en essai du prédicteur et du critère, et qui utilise subséquemment cette information pour ajuster la valeur récupérée du critère, peut expliquer cet effet.

Mots-clés : apprentissage de fonction, mémoire, modèle, ordre

References

Bott, L., & Heit, E. (2004). Nonmonotonic extrapolation in function learning. *Journal of Experimental Psychology: Learning, Memory & Cognition, 30,* 38–50. doi:10.1037/0278-7393.30.1.38

DeLosh, E. L., Busemeyer, J. R., & McDaniel, M. A. (1997). Extrapolation: The sine qua non for abstraction in function learning. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 23,* 968–986. doi:10.1037/0278-7393.23.4.968

Hintzman, D. L. (1984). MINERVA 2: A simulation model of human memory. *Behavior Research Methods, Instruments and Computers, 16,* 96–101.

Hintzman, D. L. (1986). "Schema abstraction" in a multiple-trace memory model. *Psychological Review, 93,* 411–428. doi:10.1037/0033-295X.93.4.411

Hintzman, D. L. (1988). Judgments of frequency and recognition-memory in a multiple-trace memory model. *Psychological Review, 95,* 528–551. doi:10.1037/0033-295X.95.4.528

Jamieson, R. K., & Mewhort, D. J. K. (2009a). Applying an exemplar model to the artificial-grammar task: Inferring grammaticality from similarity. *Quarterly Journal of Experimental Psychology, 62,* 550–575. doi:10.1080/17470210802055749

Jamieson, R. K., & Mewhort, D. J. K. (2009b). Applying an exemplar model to the serial reaction time task: Anticipating from experience. *Quarterly Journal of Experimental Psychology, 62,* 1757–1783. doi: 10.1080/17470210802557637

Jamieson, R. K., & Mewhort, D. J. K. (2010). Applying an exemplar model to the artificial-grammar task: String-completion and performance for individual items. *Quarterly Journal of Experimental Psychology, 63,* 1014–1039. doi:10.1080/17470210903267417

Jones, M., Love, B. C., & Maddox, W. T. (2006). Recency effects as a window to generalization: Separating decisional and perceptual sequential effects in category learning. *Journal of Experimental Psychology Learning, Memory, and Cognition, 32,* 16–332. doi:10.1037/0278-7393.32.3.316

Jones, M. N., & Mewhort, D. J. K. (2007). Representing word meaning and order information in a composite holographic lexicon. *Psychological Review, 114,* 1–37. doi:10.1037/0033-295X.114.1.1

Kahneman, D., & Tversky, A. (1973). On the psychology of prediction. *Psychological Review, 80,* 237–251. doi:10.1037/h0034747

Kalish, M., Lewandowsky, S., & Kruschke, J. K. (2004). Population of linear experts: Knowledge partitioning and function Learning. *Psychological Review, 111,* 1072–1099. doi:10.1037/0033-295X.111.4.1072

Kruschke, J. K. (1992). ALCOVE: An exemplar-based connectionist model of category learning. *Psychological Review, 99,* 22–44. doi: 10.1037/0033-295X.99.1.22

Kwantes, P. J., & Mewhort, D. J. K. (1999). Modeling lexical decision and word naming as a retrieval process. *Canadian Journal of Experimental Psychology, 53,* 306–315. doi:10.1037/h0087318

Kwantes, P. J., & Neal, A. F. (2006). Why people underestimate *y* when extrapolating in linear functions. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 32,* 1019–1030. doi:10.1037/0278-7393.32.5.1019

Kwantes, P. J. (2005). Using context to build semantics. *Psychonomic Bulletin and Review, 12,* 703–710. doi:10.3758/BF03196761

McDaniel, M. A., Dimperio, E., Griego, J. A., & Busemeyer, J. R. (2009). Predicting transfer performance: A comparison of competing function learning models. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 35,* 173–195. doi:10.1037/a0013982

Nosofsky, R. M., & Palmeri, T. J. (1998). A rule-plus-exception model for classifying objects incontinuous-dimension spaces. *Psychonomic Bulletin & Review, 5,* 345–369. doi:10.3758/BF03208813

Nosofsky, R. M. (1986). Attention, similarity, and the identification-categorization relationship. *Journal of Experimental Psychology: General, 115,* 39–57. doi:10.1037/0096-3445.115.1.39

Reber, A. S. (1967). Implicit learning of artificial grammars. *Journal of Verbal Learning and Verbal Behavior, 6,* 855–863. doi:10.1016/S0022-5371(67)80149-X

Simon, H. A. (1996). *The sciences of the artificial.* Cambridge, MA: MIT Press.

Stewart, N., Brown, G. D. A., & Chater, N. (2002). Sequence effects in categorization of simple perceptual stimuli. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 28,* 3–11. doi:10.1037/0278-7393.28.1.3

Tulving, E. (1972). Episodic and semantic memory. In E. Tulving and W. Donaldson (Eds.), *Organization of Memory* (pp. 381–402). New York, NY: Academic Press.

Zotov, V., Jones, M. N., & Mewhort, D. J. K. (2011). Contrast and assimilation in categorization and exemplar production. *Attention, Perception, & Psychophysics, 73,* 621–639.

Appendix A

Training and Test Items Used in the Experiment and Simulation

Training	30, 32, 33, 35, 37, 39, 41, 44, 46, 49, 52, 54, 57, 59, 62, 64, 66, 67, 69, 70
Test	1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 32.5, 35, 37.5, 40, 42.5, 45, 47.5, 50, 52.5, 55, 57.5, 60, 62.5, 65, 67.5, 71, 73, 75, 77, 79, 81, 83, 85, 87, 89, 91, 93, 95, 97, 99

Received August 8, 2011
Accepted October 20, 2011 ■