Locally Bayesian Learning with Applications to Retrospective Revaluation and Highlighting

John K. Kruschke

Contents

Revision of December 28, 2005.
Submitted for publication.

Locally Bayesian Learning with Applications to Retrospective Revaluation and Highlighting

John K. Kruschke
Indiana University

A scheme is described for locally Bayesian parameter updating in models structured as successions of component functions. The essential idea is to back-propagate the target data to interior modules, such that an interior component's target is the input to the next component that maximizes the probability of the next component's target. Each layer then does locally Bayesian learning. The resulting parameter updating is not globally Bayesian, but can better capture human behavior. Locally Bayesian learning can also learn faster than globally Bayesian learning in some situations. The approach is implemented for an associative learning model that first maps inputs to attentionally filtered inputs, and then maps attentionally filtered inputs to outputs. The model is applied to several phenomena exhibited in human learning that have heretofore been unaddressed by Bayesian learning models or by associative learning models. The Bayesian updating allows the associative model to exhibit retrospective revaluation effects such as backward blocking and unovershadowing. The back-propagation of target values to attention allows the model to show trial-order effects, including highlighting and differences in magnitude of forward and backward blocking.

Cognitive systems are often thought of as hierarchies of processes. Each process takes an input representation, transforms it, and generates another representation. That representation in turn is transformed by a subsequent process, until an ultimate representation corresponds with a response or outcome. For example, Marr (1982) expounded a representational framework that progressed from a representation of image intensity to a "primal sketch" to a "$2\frac{1}{2}$-D sketch" to a 3-D model representation. Palmer (1999) outlined four stages of visual processing, from image-based to surface-based to object-based to category-based. I am specifically interested in such architectures when applied to trial-by-trial, "online" learning. Each of the transformations within levels of the hierarchy is tuned by experience in the world.

Bayesian approaches to cognitive modeling have been especially attractive because they express optimal performance under specific assumptions. Bayesian approaches can be useful either to show that human behavior is nearly optimal, or to show specifically how human performance fails to be optimal. Bayesian approaches also stipulate how the model should adjust its distribution of parameter probabilities when data are supplied. Thus, Bayesian updating describes optimal learning. Bayesian learning has been applied to a range of phenomena from low-level perceptual learning (e.g., Eckstein, Abbey, Pham, & Shimozaki, 2004) to high-level causal induction and language acquisition (e.g., Regier & Gahl, 2004; Steyvers, Tenenbaum, Wagenmakers, & Blum, 2003).

If the Bayesian approach to learning is to be a general principle for modeling the mind, then it is logical to attempt Bayesian learning for the entire hierarchy of representations simultaneously. However, in a system as complex as the mind, replete with myriad parameters, it is unlikely that every episodic experience catalyzes a monolithic Bayesian updating of the complete joint parameter distribution simultaneously. Perhaps it is not being too mystical, however, to imagine that there is Bayesian updating within modules. Perhaps for small subspaces of parameters, there is Bayesian updating within each subspace. The problem is that most modules in the mental hierarchy are not in direct contact with the stimuli provided by the outside world, and so they do not know what data to use for updating their parameters.

There are three main points in this article, addressed in turn. First, I report a new general scheme for doing locally Bayesian updating in models structured as successions of component functions. The essential idea is to back-propagate the target data to interior modules, such that the interior targets are those that maximize the probability of the target in the subsequent layer. Second, I implement the approach for an associative learning model that first maps inputs to attentionally filtered inputs, and then maps attentionally filtered inputs to outputs. Third, I apply the model to several phenomena exhibited in human learning that have hereto-

Supported in part by Grant BCS-9910720 from the National Science Foundation. A portion of this research was accomplished while the author was on sabbatical during Fall 2004. For comments on earlier drafts, I gratefully acknowledge Jerome Busemeyer, Michael Lee, Richard Shiffrin, Joshua Tenenbaum, Eric-Jan Wagenmakers and two anonymous reviewers, along with David Blei, Jason Gold, Richard Golden, Robert Jacobs and Chen Yu. Some of this research was presented at the 38th Annual Meeting of the Society for Mathematical Psychology, 6 August 2005, University of Memphis, TN. Correspondence can be addressed to John K. Kruschke, Department of Psychological and Brain Sciences, Indiana University, 1101 E. 10th St., Bloomington IN 47405-7007, or via electronic mail to kruschke@indiana.edu. The author's world wide web page is at http://www.indiana.edu/~kruschke/

fore been unaddressed by Bayesian learning models or by associative learning models. The locally Bayesian model can learn faster than the globally Bayesian model in some applications. The Bayesian updating allows the associative model to exhibit retrospective revaluation effects such as backward blocking and unovershadowing. These effects are challenging for many non-Bayesian associative learning models. The back-propagation of target values to attentionally filtered cues allows the model to show trial-order effects, including highlighting and differences in the magnitudes of forward and backward blocking. These trial-order effects are challenging for many extant Bayesian learning models.

Bayesian modeling generally

The benefits of Bayesian approaches to model fitting and model comparison have been compellingly discussed and demonstrated (e.g., Lee, 2004; MacKay, 2003; Myung & Pitt, 1997). Here I provide a brief overview of Bayesian modeling, as background for discussing Bayesian models of learning. Suppose we have data that we are trying to model. Each datum represents the target response $t^{(i)}$ on the i^{th} trial when the cognizer is presented with stimulus $x^{(i)}$. We denote a model, also called a hypothesis, by M. The model is a mathematical formula that generates *probabilities* (or probability densities) of possible data values for each input $x^{(i)}$. Typically M will have parameters θ whose values determine the exact numerical behavior of the model. Thus, the model is a formula that generates $p(t|\theta,x,M)$.

An example of such a model is the well known simple linear regression model with Gaussian noise, which expresses the probability density of a value t as a function of the stimulus x and three parameters: the intercept β_0, the slope β_1, and the standard deviation σ. The formula for this model, M, is $p(t|\beta_0,\beta_1,\sigma,x,M) = (1/(\sigma\sqrt{2\pi}))\exp(-.5[t-(\beta_1 x+\beta_0)]^2/\sigma^2)$.

In a Bayesian approach, we think of many values of each parameter as being possible, with each value having a certain probability of being correct. Before we have any experimental data about a situation being modeled, we specify a *prior* probability distribution over the parameters, denoted $p(\theta|M)$, which quantifies our degree of belief in each value of θ.

One goal we might have is estimation of parameter values from data. In a Bayesian framework, this goal means that we want to shift our probabilities for each parameter value when given data. *Bayes' theorem* expresses how to do that:

$$p(\theta|t,x,M) = p(t|\theta,x,M)p(\theta|M)/p(t|x,M) \tag{1}$$

The distribution $p(\theta|t,x,M)$ is called the *posterior* of θ. Notice that Bayes' theorem (Equation 1) expresses the posterior distribution $p(\theta|t,x,M)$ in terms of the model's predicted probabilities $p(t|\theta,x,M)$ and the prior distribution $p(\theta|M)$. The denominator $p(t|x,M)$ will be discussed shortly.

A second goal we might have is generating the response predicted by a model. As mentioned before, in a Bayesian framework there is no single value for the parameters; instead, many values of the parameters are possible, each with a certain probability or degree of belief. So to generate the probability of a response value y (which could be the same value as the datum t), we integrate across all possible values of θ, weighted by the probability of θ:

$$p(y|x,M) = \int d\theta\, p(y|\theta,x,M)\, p(\theta|M). \tag{2}$$

where the probability $p(\theta|M)$ is whatever our current beliefs are, which might incorporate previously observed data. When $y=t$, Equation 2 expresses the denominator of Bayes' formula in Equation 1.

When we desire a unique value for the predicted output, rather than a probability distribution over possible values, and when t is a metric variable, then the predicted output is taken to be the expected value:

$$\bar{y} = \int dy\, y\, p(y|x,M) \tag{3}$$

when y is continuous or $\bar{y} = \sum y\, p(y|x,M)$ when y is discrete.

A third goal we might have is model comparison. We might have two (or more) different models, M_1 and M_2, each with its own set of parameters, θ_1 and θ_2. Or, we might have one model form with two different priors on the parameters, which can then be thought of as competing models. In either case, we start with some prior belief about the probability that each model is true. These prior probabilities of the models are denoted $p(M_i)$. Bayes' formula tells us how to modify those beliefs when we consider the data:

$$p(M_i|t,x) = p(t|x,M_i)\, p(M_i)/p(t|x) \tag{4}$$

where the denominator is given by

$$p(t|x) = \sum_i p(t|x,M_i)p(M_i) \tag{5}$$

In some applications, there is a continuum of models rather than a finite set, and so the summation in Equation 5 becomes an integral. Notice that the integral from Equation 2 shows up again in Equation 4, and thus the integral appears in all the three goals of Bayesian modeling.

Much of the effort in Bayesian modeling goes into evaluating the integral in Equation 2. For simple models, the integral can be evaluated analytically; that is, using clever mathematical derivation. In other cases, the integral can be approximated analytically, with simpler formulas substituted for the exact model. When neither of those approaches is feasible, numerical approximation is used. For very small parameter spaces, the parameter space can be sampled at regular intervals, like a comb or grid, with the terms of the integral computed at each interval and summed up. For even slightly larger parameter spaces, there are far too many grid points to evaluate in a feasible time, and therefore sophisticated Monte Carlo sampling schemes have been invented to sample the parameter space proportionally to probability density.

Everyday Bayesian reasoning

"How often have I said to you that when you have eliminated the impossible, whatever remains, however improbable, must be the truth?" So said Sherlock Holmes in Arthur Conan Doyle's novel, *The Sign of Four* (1890, Ch. 6). This reasoning is actually a consequence of Bayesian belief updating, as expressed in Equations 4 and 5. Let me re-state it this way: "How often have I said to you that when $p(t|x,M_i) = 0$ for $i \neq j$, then, no matter how small the prior $p(M_j) > 0$ is, the posterior $p(M_j|t,x)$ must equal one." Somehow it sounds better the way Holmes said it.

The Holmesian logic upgrades belief in a hypothesis when belief in another hypothesis is downgraded. The complement of that logic downgrades belief in a hypothesis when another (mutually exclusive) hypothesis is upgraded. For example, when an object d'art is found fallen from its shelf, the prior may indict the house cat, but when the visiting toddler is seen dancing next to the shelf, the cat is exonerated. This downgrading of a hypothesis is sometimes called "explaining away." It also follows from Bayesian belief updating: When $p(t|x,M_j)$ increases but $p(t|x,M_i)$ is unchanged for $i \neq j$, then $p(M_j|t,x)$ increases while $p(M_i|t,x)$ decreases for $i \neq j$.

We will later see results from associative learning experiments, referred to as unovershadowing and backward blocking, respectively, consistent with these forms of reasoning (although people are not quantitatively accurate with this sort of reasoning, especially when there are more than two hypotheses; see Van Wallendael & Hastie, 1990).

Bayesian modeling as cognizing

When applied to data analysis, Bayesian modeling involves formal models created by statisticians to describe patterns in data. But we can also imagine the mind, *qua* statistical homunculus, as doing something like Bayesian analysis when it receives data from the senses. At any moment, the mind has some hypotheses about the world, with a certain degree of belief in each one. This entails degrees of belief about hypotheses, and degrees of belief about possible parameter values within each hypothesis. The senses then provide more data about the world, and the mind updates its beliefs. If the mind is Bayesian, then Equations 1 and 4 specify the updating of the belief probabilities. Furthermore, for any particular state of beliefs, the mind can generate predictions about the world when presented with stimulus x. If the mind is Bayesian, it will have beliefs about possible predicted values t as specified by Equation 2.

For theorists who wish to explore Bayesian models of cognition there are several challenges. Perhaps foremost among these challenges is specification of the hypotheses in the mind (over which it does Bayesian updating of belief probabilities). Once a theorist has posited particular model functions, then another challenge is showing that Bayesian updating of belief probabilities matches human learning. Research in the 1960's and 1970's (e.g., Edwards, 1968; Godden, 1976; Shanteau, 1975) tried to make the hypotheses utterly simple and explicit. For example, subjects were told the numbers of red and blue chips sampled so far from an unknown bag, and were asked to judge the probability that the chips came from either a hypothetical bag with 70% blue chips and 30% red, or from a hypothetical bag with 30% blue chips and 70% red. People did not adjust their judgments as extremely as prescribed by Equation 4. More recently, Kitzis, Kelley, Berg, Massaro, and Friedman (1998) found that a Bayesian updating model fit their data better than a number of non-Bayesian learning models, but still not very well: Humans showed over-sensitivity to recent trials and over-reliance on cues with relatively greater diagnosticity. In these studies the Bayesian models involved hypotheses with fixed (or punctate) parameter values and only Equation 4. Learning did not involve distributions of parameter values, and therefore never the complexity of invoking Equations 1 and 2.

Depending on the particular model form, and especially if the model involves distributions of parameter values, Bayesian updating itself can be computationally intensive. In these cases it can be very difficult to determine accurately the predictions of a Bayesian model, and it can cause eyebrows to be raised when asserting that the mind is capable of analogous computations. The present article describes a general way to reduce the computational demands of Bayesian updating when the model architecture consists of a feed-forward network of component functions. The result is not necessarily Bayesian overall, but this scheme exhibits several behaviors displayed by human learners which are difficult for previous Bayesian models to accommodate.

Trial order invariance

A characteristic of many recent Bayesian learning models is that they do not depend on trial order. Given two training items, the posterior probability distribution does not depend on the order in which the training items are provided. This trial-order invariance is desirable in cases when all data should be treated as equally relevant, regardless of order. In human learning this might not be how people actually treat data that are temporally distributed. We will see that the new method introduced later does not enjoy, or suffer from, trial-order invariance.

The standard Bayesian terminology for probability distributions over parameters, "prior" and "posterior," is misleading insofar as it connotes the passage of time. There is *no* time in the Bayesian formula that relates posterior to prior distributions. More accurate terminology would refer to the distribution of θ with particular data *excluded* versus the distribution of θ with those data *included*. With that caveat in mind, I will comply with the traditional terminology.

The reason that trial order has no impact in many Bayesian models is that the data are assumed to be drawn independently from a stationary probability model. The model function $p(y|\theta,x)$ is typically assumed to be independent of time or trial and independent of any data generated previously. Appendix A provides a mathematical derivation of trial order independence in Bayesian updating. Some specific Bayesian models do explicitly represent time, with probability distri-

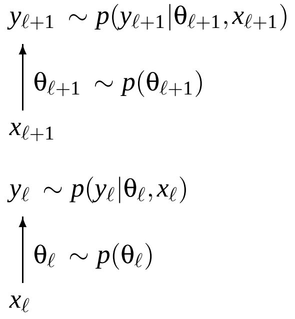
$y_{\ell+1} \sim p(y_{\ell+1}|\theta_{\ell+1}, x_{\ell+1})$
$\theta_{\ell+1} \sim p(\theta_{\ell+1})$
$x_{\ell+1}$
$y_\ell \sim p(y_\ell|\theta_\ell, x_\ell)$
$\theta_\ell \sim p(\theta_\ell)$
x_ℓ

Figure 1. Architecture of successive functions. Vertical arrows indicate a mapping from input to output within a layer, parameterized by θ. The notation "$\theta \sim p(\theta)$" means that θ is distributed according to the probability distribution $p(\theta)$. In the globally Bayesian approach, $x_{\ell+1} = y_\ell$. In the locally Bayesian approach, $x_{\ell+1} = \bar{y}_\ell$.

butions that are functions of time, but time is not inherent in the general Bayesian approach any more than spatial location. Obviously, if a Bayesian model makes its probability function an explicit function of time or trial, then the model will be able to show effects of time, which might or might not match human behavior. Models involving time can be complex, however, and so many Bayesian models avoid functions of time merely for convenience, not out of theoretical commitment.

Effects of trial order should not be confused with effects of number of trials. Even Bayesian models that are not sensitive to trial order are sensitive to the number of times that a datum has appeared. This is simply because the prior probability distribution is gradually overwhelmed by the accretion of data through training. For example, both Danks, Griffiths, and Tenenbaum (2003) and Courville, Daw, Gordon, and Touretzky (2004) described models that changed their behavior during training because a prior was gradually overcome by data, but in neither case were the models sensitive to the ordering of the training items.

Thus, in Bayesian models that have no time dependencies, trial order has no influence on the ultimate posterior probability. In particular, this means that any empirical phenomena that depend on training order, such as those phenomena explored in subsequent sections, will not be exhibited by such Bayesian models. Existing models of learning that are trial-order invariant include a special case of the Kalman filter (Dayan, Kakade, & Montague, 2000) discussed later, Bayesian neural networks (e.g., MacKay, 2003; Neal, 1996), sigmoid belief networks (Courville et al., 2004), and noisy-OR causal models (e.g., Sobel, Tenenbaum, & Gopnik, 2004; Tenenbaum & Griffiths, 2003). Trial-order invariance is a deficiency for many existing Bayesian models that are intended to address human and animal learning, which can be highly sensitive to trial order.

The architecture: Successive functions

For simplicity I will assume that the model of interest can be expressed as a succession of component functions. One function maps the stimulus representation to an internal representation, a second function maps that representation to another, and so on, until a final function maps the last internal representation to a response representation. Usually I will call each function a "layer," but occasionally I will refer to a function as a "module" or a "component" in the sequence.

The input vector for the ℓ^{th} layer is x_ℓ. This is displayed at the bottom of Figure 1. The probability of output vector y_ℓ is specified by the function for that module, and is denoted $p(y_\ell|\theta_\ell, x_\ell)$, where θ_ℓ is the parameter vector for the function. The theorist provides a distribution of prior probabilities of the parameter values, denoted $p(\theta_\ell)$. All these variables are denoted in the lower part of Figure 1. The next layer up is also shown in Figure 1, with subscripts of $\ell+1$. The ℓ^{th} layer's output provides the input to the $\ell+1^{st}$ layer. The final layer is indexed by $\ell = L$ and the first layer is indexed by $\ell = 1$. A stimulus-target pair, on which the sequence of modules is trained, is denoted $\langle t_L, x_1 \rangle$. On successive training trials, a sequence of such input-output pairs are presented.

As mentioned in the introduction, there are many examples of cognitive models that assume this sort of architecture. I am specifically interested in such architectures when applied to trial-by-trial learning.

Globally Bayesian updating

In the standard approach, all the layers are treated as one integrated model that maps x_1 to y_L, having parameters $\theta_L, \ldots, \theta_1$ with prior probability distribution $p(\theta_L, \ldots, \theta_1)$ over the joint parameter space.

Predicting the output for a given input

When provided with a particular value for the input x_1, we would like to know what the model predicts for the final output, y_L. The model functions do not compute a unique value of an output, instead they specify the probabilities of values of y_L. Because unique values of the parameters are not known, we marginalize across all the possible parameter values, just as in Equation 2, to get the probability distribution of output values:

$$p(y_L|x_1) = \int d\theta_L \ldots d\theta_1 \; p(y_L|\theta_L, \ldots, \theta_1, x_1) \; p(\theta_L, \ldots, \theta_1) \tag{6}$$

The actual computation of the integral in Equation 6 can be simplified, but my aim here is simply to point out that, conceptually, the integral is over the high-dimensional joint parameter space.

When we want a unique predicted value for the output, we use the expected (i.e., mean) value across possible output values, as was described for the general case in Equation 3.

For layers of successive functions, the marginal output is expressed as

$$\bar{y}_L = \int dy_L \, y_L \, p(y_L|x_1) \tag{7}$$

Equation 7 assumes that y_L is a metric variable, so it makes sense to add (i.e., integrate) different values of y_L. When y_L is instead a nominal variable, then Equation 7 does not apply, and the prediction for the output is left as Equation 6.

Parameter estimation and learning

When provided with a target output t_L, we would like to estimate parameters that accommodate that target. In the maximal likelihood estimation (MLE) approach to parameter estimation, we choose the single vector of values of $\theta_L, \ldots, \theta_1$ that maximizes the likelihood, $p(t_L|\theta_L, \ldots, \theta_1, x_1)$. In the Bayesian approach, on the other hand, we begin with a prior probability, $p(\theta_L, \ldots, \theta_1)$, over the parameter space and derive a posterior probability distribution according to Bayes' theorem (Eqn. 1), which becomes:

$$\begin{aligned} & p(\theta_L, \ldots, \theta_1|t_L, x_1) \\ & \quad = \frac{p(t_L|\theta_L, \ldots, \theta_1, x_1)\, p(\theta_L, \ldots, \theta_1)}{p(t_L|x_1)} \end{aligned} \tag{8}$$

where the denominator is determined by Equation 6 when $y_L = t_L$. This updating of the probability distribution over the parameter space is referred to as Bayesian "learning" of the parameters.

Computational demands

For either goal of predicting or learning, we need to compute the overall likelihood, $p(y_L|\theta_L, \ldots, \theta_1, x_1)$, which appeared in Equations 6 and 8. We only have, however, the single-layer expressions, $p(y_\ell|\theta_\ell, x_\ell)$, specified by the models in each layer. So we need to re-express the overall likelihood in terms of the individual layer functions.

In the present scenario, using layers of independent modules, the overall likelihood can be determined (in principle) by starting at the first layer and working through each successive layer. Thus, beginning with the first layer, we determine

$$\begin{aligned} & p(y_2|\theta_2, \theta_1, x_1) = \\ & \quad \int dy_1 \; p(y_2|\theta_2, y_1) \; p(y_1|\theta_1, x_1) \end{aligned} \tag{9}$$

The result of Equation 9 is to be thought of as a formula for a function of the variable y_2 (and of the variables θ_ℓ). We can then determine the formula for the probability distribution at the next layer:

$$\begin{aligned} & p(y_3|\theta_3, \theta_2, \theta_1, x_1) = \\ & \quad \int dy_2 \; p(y_3|\theta_2, y_2) \; p(y_2|\theta_2, \theta_1, x_1) \end{aligned} \tag{10}$$

We proceed up the layers until we get the general formula for the probability at the last layer, which we then evaluate with $y_L = t_L$.

This recursive procedure is tractable when the integral at each level can be analytically formulated. An example is when each probability density function is a linear transformation with Gaussian noise (e.g., Neapolitan, 2004, Ch. 4). If instead each integral must be numerically approximated, the situation becomes very computationally demanding. To wit, suppose that each variable y_ℓ is represented by a comb of V values over its range. Consider fixed, specific values for $\theta_\ell, \ldots, \theta_1$. Then at the ℓ^{th} layer we compute an array of V values of $p(y_\ell)$ by, at each value of y_ℓ, summing (i.e., integrating) the product $p(y_\ell|\theta_\ell, \ldots, \theta_1, y_{\ell-1})\, p(y_{\ell-1}|\theta_{\ell-1}, \ldots, \theta_1, x_1)$ over the V values of the previous layer's $y_{\ell-1}$. This summation is done recursively up the layers. If each parameter θ_ℓ is specified on a grid of P values, then that recursive summation must be carried out for every one of the P^L combinations of particular parameter values. For even modestly dense combs over variables y_ℓ and grids over parameters θ_ℓ, the combinatorics grow explosively.

The computational demands leave us with the following alternatives: 1. Restrict our model functions to forms that yield analytically tractable integrals. 2. Specify model functions however we desire, but approximate them with analytically tractable forms, and demonstrate that the approximations are good under the circumstances we use. 3. Use sparse combs and grids over the variables and show that the approximations are good under the circumstances we use. 4. Use sophisticated Monte Carlo sampling over the variables and show that the approximations are good under the circumstances we use. Much previous work has gone into each of these approaches (see, for example, textbooks by Gill, 2002; MacKay, 2003; Gelman, Carlin, Stern, & Rubin, 2004).

The new approach: Locally Bayesian updating

There is another approach to the problem: Jettison the goal of being globally Bayesian, and instead assume only that each module is Bayesian by itself. One motivation for this approach is that the computations for globally Bayesian updating might be prohibitive. A second motivation is that locally Bayesian learning can, in some circumstances, be faster than globally Bayesian learning. Examples are described later in the article. But even if special cases of globally Bayesian computations are tractable and fast, there is a third motivation. In the course of evolutionary design of the mind, there might be component functions that are used for various different activities. Each of these components might learn and develop and evolve somewhat independently of the others, to enhance flexibility and damage resistance. Yet each should be Bayesian in the context of its own interior environment. Each component only knows about the information it receives from its immediately contiguous neighbors, and within its myopic environment it should be Bayesian.

Notice that Bayesian learning (Equation 1) requires a specific input (x) and a specific teacher (t). An interior layer, however, receives a distribution of inputs (y from the previous layer) and no explicit teacher at all. To accomplish locally Bayesian learning for an interior layer, we will specify a particular input and teacher. Once we specify those values, then each layer will do Bayesian learning on its own. These locally Bayesian modules might or might not yield globally Bayesian behavior, but they can mimic some aspects of human behavior.

Specific input for locally Bayesian learning

Each module takes as its input the marginal output from the previous layer. The marginal output is a standard Bayesian approach to prediction, as was described in Equation 2. For this purpose, I here assume that y_ℓ is on a metric scale, and therefore can be integrated, or summed if it is discrete valued. Formally, the input to layer $\ell+1$ is the predicted (i.e., marginalized) value of the output of module ℓ:

$$\begin{aligned} x_{\ell+1} &= \bar{y}_\ell \\ &= \int dy_\ell \, y_\ell \; p(y_\ell|x_\ell) \\ &= \int dy_\ell \, y_\ell \int d\theta_\ell \; p(y_\ell|\theta_\ell,x_\ell) \; p(\theta_\ell) \\ &= \int d\theta_\ell \; p(\theta_\ell) \int dy_\ell \, y_\ell \; p(y_\ell|\theta_\ell,x_\ell) \qquad (11) \end{aligned}$$

Equation 11 is then applied recursively up the sequence of layers, so every layer has a specific input. (This use of the mean output for the input to the next layer assumes that the mean is a valid input for the next layer; i.e., that the mean lies in the domain of the next layer's function.) The form provided in the last line of Equation 11 is useful computationally, because it first finds the mean output for a specific hypothesized value of θ, and then integrates those mean outputs weighted by the probability of the hypotheses. To recapitulate: In the globally Bayesian approach, the input to a layer is the output value of the layer that feeds it, but the lower-layer output is probabilistically distributed. In the locally Bayesian approach, the input to a layer is the mean output value of the layer that feeds it, and the mean value is determinate.

The final layer variable, y_L, does not need to be metrically scaled because we do not need to marginalize its output for feeding another layer. When the final layer's output is nominally scaled, we cannot sum over values of y_L, and the output is

$$p(y_L) \;=\; \int d\theta_L \; p(y_L|\theta_L,\bar{y}_{L-1}) \; p(\theta_L) \qquad (12)$$

for each value of y_L.[1]

Specific target for locally Bayesian learning

A training item specifies the target vector at the final output layer, which is useful for global Bayesian updating (as in Equation 8). If, however, we want local updating within each layer, we need an analogous target vector for every interior layer. Clearly we should like the teacher t_ℓ for layer ℓ to be a value that makes the probability of the final teacher large. Indeed, we should like to find the value of t_ℓ that maximizes $p(t_L|t_\ell)$. Unfortunately, only the final layer has access to the external teacher. Therefore, we start at the last layer and determine a teacher for the penultimate layer, then for the previous layer, and so on down as many layers as needed.

Formally, when the ℓ^{th} layer has a target vector t_ℓ but the target $t_{\ell-1}$ for the layer below is unknown, we choose for the lower-layer target that value which maximizes the probability of the current-layer target:

$$\begin{aligned} t_{\ell-1} &= \underset{x^*_\ell}{\operatorname{argmax}} \; p(t_\ell|x^*_\ell) \\ &= \underset{x^*_\ell}{\operatorname{argmax}} \int d\theta_\ell \; p(t_\ell|\theta_\ell,x^*_\ell) \; p(\theta_\ell) \qquad (13) \end{aligned}$$

Equation 13 simply states that the target for the layer below is the input to the current layer that would maximize the probability of the target for the current layer. The variable x^*_ℓ is given a superscript star to distinguish it from the input value $x_\ell = \bar{y}_{\ell-1}$. Notice that Equation 13 can be recursively applied down the levels of modules: The "outside world" provides t_L for the last module, and then Equation 13 is recursively applied from the last module down to as many lower layers as desired. Whereas it might be desirable to find an x^*_ℓ value that *maximizes* the probability of t_ℓ as defined in Equation 13, in practice it may suffice to find an x^*_ℓ value that merely increases the probability of t_ℓ relative to $x_\ell = \bar{y}_{\ell-1}$.

Procedure for locally Bayesian learning

With targets determined by Equation 13 and inputs determined by Equation 11, we can do Bayesian updating within layers. But how should learning be interleaved with target backpropagation?

One candidate scheme is to propagate the inputs up all the layers, and propagate the teachers down all the layers, independently; i.e., using the same $p(\theta_\ell)$. After both the teachers and the inputs have been propagated, then simultaneously update the beliefs of all layers. Despite the simplicity of this simultaneous updating scheme, an infelicity is that the updating of layer ℓ changes the input $x_{\ell+1} = \bar{y}_\ell$ to the layer above, and so the already-executed learning of layer $\ell+1$ is no longer appropriate. The updating of layer ℓ also changes $t_{\ell-1}$ for the layer below, and so the already-executed learning of layer $\ell-1$ is no longer appropriate.

To address the problem of simultaneous updating, a second candidate scheme is to first propagate the teachers down all the layers, and then to update the beliefs of the layers one at a time from the lowest up, and then computing the output of each layer, one at a time, after its beliefs are updated. Thus, at the first layer, beliefs are updated using t_1 and x_1, and then, after updating $p(\theta_1|t_1,x_1)$, $\bar{y}_1$ is computed and used as input x_2 for learning in layer 2. This continues up the layers, each layer updating only after the layer that feeds it has learned and computed its output value. This scheme solves

[1] It may be possible to extend the approach to cases in which lower layer's outputs y_ℓ are nominal instead of metric. In this case, we might set $x_{\ell+1}$ to the most probable value of y_ℓ. When searching for a target that maximizes the probability of the target (as explained below), the search would have to explore the discrete space.

the problem of inappropriate inputs, but retains the problem of inappropriate teachers, i.e., updating of a layer changes the teacher for the layers below.

A third scheme addresses the problem of simultaneous updating from the opposite direction of the second scheme. In this third scheme, teachers are not backpropagated before learning. Instead, inputs are first propagated up all the layers, and then each layer, starting with the final layer, is Bayesian updated before it is used to determine a teacher value for the layer below it. In other words, learning starts with the last layer updating its beliefs using t_L and x_L. Then, with the updated distribution $p(\theta_L|t_L,x_L)$, the next lower layer teacher t_{L-1} is determined. Then $p(\theta_{L-1}|t_{L-1},x_{L-1})$ is updated, and those updated belief probabilities are used to determine t_{L-2}, and so forth. This scheme solves the problem of inappropriate teachers, but retains the problem of inappropriate inputs, i.e., updating a layer changes the inputs for the layers above.

In the absence of a computational reason to prefer one scheme over the other, I will let the typical task dynamics motivate my choice. In standard associative learning tasks, a trials consists of the following sequence of events. First, a stimulus, that is, x_1, appears. The learner is asked to predict the outcome, that is, the participant generates y_L. Then corrective feedback occurs, that is, t_L is provided. Presumably, after the feedback is presented, the learner then does some internal parameter adjustment. This sequence of events is best mimicked by the third scheme described above. The inputs are propagated first, and then the external teacher is supplied, and then learning is backpropagated down the layers. This is the approach taken in the simulations described later in the article. The other two schemes may turn out to be more appropriate for different applications, or yet other iterative updating schemes might ultimately prove to be best.[2]

In summary, learning proceeds as follows. First, input activation propagates up the layers with $x_\ell = \bar{y}_{\ell-1}$ and $\bar{y}_{\ell-1}$ defined by Equation 11. Then, for each layer, starting with the final layer and working down, belief probabilities for layer ℓ are updated according to Bayes theorem,

$$\begin{aligned} p(\theta_\ell|t_\ell,x_\ell) &= p(t_\ell|\theta_\ell,x_\ell)\,p(\theta_\ell)/p(t_\ell|x_\ell) \\ &= \frac{p(t_\ell|\theta_\ell,x_\ell)\,p(\theta_\ell)}{\int d\theta_\ell\; p(t_\ell|\theta_\ell,x_\ell)\,p(\theta_\ell)}, \end{aligned} \qquad (14)$$

and then the teacher for the next lower layer is determined using Equation 13 with the posterior distribution on θ_ℓ:

$$t_{\ell-1} \;=\; \underset{x_\ell^*}{\operatorname{argmax}} \int d\theta_\ell\; p(t_\ell|\theta_\ell,x_\ell^*)\; p(\theta_\ell|t_\ell,x_\ell). \qquad (15)$$

The updating of the parameters within a layer, based on targets specific to that layer, is what makes this approach "locally" Bayesian. This sort functional localization of Bayesian updating should not be confused with spatially local, but functionally parallel, Bayesian updating in models of pattern recognition (e.g., Bolle & Cooper, 1986). Notice also that what is being locally updated in the present scheme is the probability distribution over possible parameter values within a layer; this process is not local updating of a single parameter value as done in some analyses (e.g., Russell, Binder, Koller, & Kanazawa, 1995).

Trial order sensitivity

Each layer updates its parameter distribution according to Bayes theorem (Eqn. 14). If the model function in each layer has no time dependencies, then each individual layer will show no sensitivity to mere re-orderings of its inputs and targets. The overall system of time-independent functions can be sensitive to trial order, however, because the targets themselves (selected by Equation 15) depend on trial order.

In formal terms, consider the training items $\langle t_L^{(a)}, x_1^{(a)}\rangle$ and $\langle t_L^{(b)}, x_1^{(b)}\rangle$. Those external values do not change if the training order is changed, and a globally (time invariant) Bayesian model is not affected by changes in training order. But the locally Bayesian learner generates internal targets, $t_\ell^{(a)}$ and $t_\ell^{(b)}$, that depend on previous learning. If the training order is changed, then $t_\ell^{(a)}$ and $t_\ell^{(b)}$ might also change, and therefore the learned beliefs of the internal layers can depend on trial order. Examples of these internal targets will be described for particular applications in the simulations reported below.

An illustrative implementation

The remainder of the article illustrates locally Bayesian learning in a simple model. The model is simple enough that the analogous globally Bayesian model can also be simulated. The models are applied to the associative learning paradigms known as highlighting, backward blocking, unovershadowing, etc., to be described en route. The locally Bayesian learning model can (qualitatively) capture the human behaviors arising from these learning paradigms. The analogous globally Bayesian learning model does not exhibit highlighting, does not show a difference between forward and backward blocking, etc. Previous Bayesian learning models in the literature also do not exhibit highlighting. Previous models in the literature that do exhibit highlighting (e.g., the connectionist model EXIT, Kruschke, 2001a) do not show backward blocking or unovershadowing. Thus, the locally Bayesian learning model captures a set of phenomena that have not been spanned by previous models. Moreover, in some cases, the locally Bayesian model learns the training items faster than the analogous globally Bayesian model.

The model implemented here is intended to be an oversimplified architecture for illustrative purposes. The simple architecture permits a complete specification of a global hypothesis space, so that local- and global-learning versions of the model can be compared. I have also implemented a local Bayesian learning version of the EXIT model, which also shows all the same qualitative effects as the simple model described here. To save space, I will not further describe the locally Bayesian version of the EXIT model.

[2] There might be relations of these updating schemes to the Expectation Maximization algorithm (Dempster, Laird, & Rubin, 1977), but no formal relations have yet been worked out.

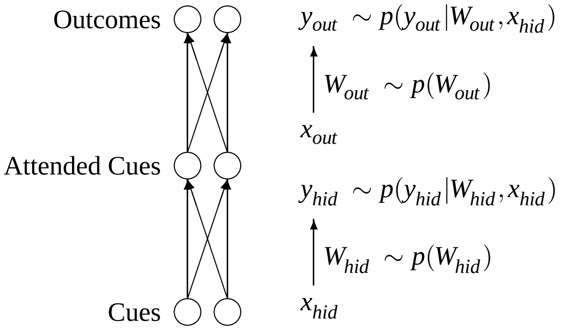
Outcomes
$y_{out} \sim p(y_{out}|W_{out},x_{hid})$
$W_{out} \sim p(W_{out})$
x_{out}
Attended Cues
$y_{hid} \sim p(y_{hid}|W_{hid},x_{hid})$
$W_{hid} \sim p(W_{hid})$
Cues
x_{hid}

Figure 2. Architecture for the simple model of associative learning. When locally Bayesian, the input to the outcome layer is the mean output of the hidden layer, i.e., $x_{out} = \bar{y}_{hid}$.

The models will be applied to simple associative learning experiments. In such experiments, the learner is shown cues that indicate outcomes. If the learners are rats, the cues might be tones or lights and the outcomes might be foot shocks or food pellets. If the learners are humans, the cues might be words on a computer screen and the outcomes might be different response buttons to click. On any trial of learning, cues are presented to the learner, who predicts an outcome, and who is then presented with corrective feedback. In category learning procedures, the feedback indicates which response would have been correct, not just whether the response was wrong.

When there are multiple cues presented on a trial, it is reasonable to suppose that the learner may selectively attend to some cues and ignore other cues. A major tradition in theories of learning is that people and animals do, in fact, learn to attend to cues that are diagnostic for correct responses, and learn to ignore cues that are irrelevant (e.g., Kruschke, 2003a; Mackintosh, 1975; Trabasso & Bower, 1968). Thus, when learning to associate cues with (overt) responses, people are also learning to associate cues with (covert) attentional distributions over those cues.

The model is a simplistic instantiation of the notion that cues are associated with attentionally filtered versions of the cues, which are then associated with outcomes. Figure 2 illustrates the model's architecture. The model has connectivity like a two-layer connectionist network. Variables involved with feeding the middle, a.k.a., "hidden" layer, are denoted by a subscript *hid*. Variables involved with feeding the outcome layer are denoted by a subscript *out*. Stimuli are represented in the network by corresponding cue nodes. A cue node has activation $x_{hid} = 1$ if the cue is present, and has activation zero otherwise. Cue activations are propagated to the hidden layer, which represents attentionally filtered cues, also with zero/one activations. The attended cue activations are then propagated to the outcome layer, which has nodes that represent the categorical choice options, again represented with zero/one activations.

From cues to attended cues

The hidden nodes are intended to represent attentionally filtered copies of the stimulus cues. Therefore each hidden node corresponds to an input node, and there are as many hidden nodes as input nodes. The activation of a hidden node is much like the standard connectionist network: The net input to a node is computed by summing over the weighted incoming connections, and then the probability of becoming activated is a squashing function of the net input.

Formally, let the stimulus cues be denoted by the column vector x_{hid}, with N components corresponding to the N cues. The weights going to the hidden nodes, from the cues, are denoted by the matrix W_{hid}. The j^{th} row of W_{hid}, denoted $W_{hid,j}$, contains the weights that converge upon the j^{th} hidden node. The net input to the j^{th} hidden node is the weighted sum of the cue activations, which can be expressed as the matrix product $W_{hid,j}x_{hid}$.

The hidden node activations also form a column vector, denoted y_{hid}, consisting of 1's and 0's. A hidden node has activation 1 if the corresponding cue is being attended to, and has activation 0 if the corresponding cue is being ignored. The probability that the j^{th} hidden node is activated is defined as

$$p(y_{hid,j}{=}1|W_{hid},x_{hid}) = \text{sig}(W_{hid,j}x_{hid})^c \qquad (16)$$

where $\text{sig}(x) = 1/(1+\exp(-x))$ is the well-known sigmoid function. The sigmoid function is raised to the power $c = 6$ (an arbitrary value) in Equation 16 so that the probability of being activated is nearly zero when the net input is zero. Equation 16 is assumed to apply to each hidden node independently, so that the probability of any particular vector of hidden node activations is the product of the probabilities of the individual node activations.

Any particular set of weight values is a hypothesis for the mapping from cues to attention. The model is provided with a set of (fixed) weight matrices that define its hypothesis space. Because the hidden nodes are intended to represent corresponding cues, the weight to the j^{th} hidden node from the j^{th} cue is constrained to be positive, which causes the hidden node to tend to be activated when the corresponding cue is present. In the demonstrations reported below, the connections to the j^{th} hidden node from the j^{th} cue were allowed to have values of either 4 or 6, because when $\text{net} = 6$, $\text{sig}(\text{net})^6$ is nearly a probability of 1. The "lateral" connections, to a hidden node from some non-corresponding cue node, were allowed to be inhibitory or neutral, having values of either -4 or 0. As an example of a hypothesis, a hidden node could have a weight of 6 from the corresponding cue node, a weight of 0 from a second cue, and a weight of -4 from a third cue. The inhibitory weight from the third cue reduces the probability that the hidden node "attends" to the first cue when the third cue is present.

The hypothesis space consists of all possible combinations of the weight values. If, for example, there are 3 cues, then there are 9 hidden weights and $2^9 = 512$ hypotheses. Notice that the hypothesis space is unbiased: All combinations of lateral inhibition are represented in the space, so any cue node has equal opportunity to inhibit any other hidden node.

The untrained network begins with a prior probability on the weight matrices, and learning consists of changing those belief probabilities. In the simulations presented below, the prior distribution is uniform: Every available hypothesis has equal probability. Because every hypothesis in the space has its mirror opposite also in the space, the uniform prior yields unbiased hidden node probabilities that merely probabilistically copy the input cues.

When propagating activation up the network for locally Bayesian learning, the mean hidden node activation (corresponding to Equation 11) is

$$\bar{y}_{hid} \quad = \quad \sum_{W_{hid}} p(W_{hid}) \sum_{y_{hid}} y_{hid} \; p(y_{hid}|W_{hid}, x_{hid}) \qquad (17)$$

where the first sum is over all weight matrices in the hypothesis space and the second sum is over all 2^N vectors $y_{hid} \in \{0,1\}^N$ that make up the hidden activation space.

From attended cues to outcomes

The outcome nodes represent the possible response categories. In general, there is one outcome node per response category. Associative weights, denoted W_{out}, go to the outcome nodes from the hidden nodes. The k^{th} outcome node computes its net input by summing over the weighted hidden activations: $net_{out,k} = W_{out,k} x_{out}$ (where $x_{out} = \bar{y}_{hid}$). One and only one outcome node gets an activation of 1 while the other outcome nodes get an activation of 0. The probability that the k^{th} outcome node is activated is defined as $p(y_{out,k}{=}1|W_{out}, x_{out}) = \exp(net_{out,k}) / [\sum_j \exp(net_{out,j})]$. This formula is simply the well-known softmax function from the connectionist literature (Bridle, 1990), and is also an often-used exponentiated version of Luce's choice rule (Luce, 1959).

In the particular applications here, there are only two outcome categories. In this situation, the second outcome node is redundant with the first outcome node (because the outcomes are mutually exclusive), and so the architecture can be simplified such that there is only one outcome node. When the single outcome node has value 1, the first outcome is chosen, and when the outcome node has value 0, the other outcome is chosen. The softmax function on two nodes is algebraically equivalent to a sigmoid function on a single node. Thus, in the simulations reported below, the probability of activating the single outcome node is

$$p(y_{out}{=}1|W_{out}, x_{out}) = \mathrm{sig}(net_{out}). \qquad (18)$$

In the simulations reported below, the model was provided with a set of fixed outcome weight matrices, W_{out}, that collectively define its hypothesis space for mapping attended cues to outcomes. Each individual weight was allowed to be inhibitory, neutral, or excitatory. For simplicity, I chose values of -5, 0 and 5, because when net $= 5$, sig(net) is a probability of nearly 1, and when net $= -5$, sig(net) is a probability of nearly 0. When there are N hidden nodes, there are 3^N combinations of outcome weights; i.e., there are 3^N hypotheses in the hypothesis space for mapping attended cues to outcomes.

As is typical in applications of Bayesian connectionist networks (e.g., MacKay, 1992, 2003; Rumelhart, Durbin, Golden, & Chauvin, 1995), I specify a Gaussian prior on the outcome-weight hypothesis space. In situations when the two categorical outcomes should have no prior bias, the prior probability of W_{out} is set proportional to $\prod_i \mathrm{norm}(w_i; 0, 5)$, where w_i is the i^{th} component of W_{out} and $\mathrm{norm}(w; \mu, \sigma)$ is the normal probability density of w for mean μ and standard deviation σ (and covariances assumed to be all zero). Thus, the hypothesis space begins by favoring hypotheses with neutral (zero weight) connections, and by symmetrically doubting hypotheses that contain positive or negative associations.

The model is also applied to situations in which the two outcomes are not initially unbiased. Consider, for example, the situation of a rat having to learn an association between a tone and a foot shock. The outcome categories are presence or absence of shock. Initially, there should be little expectation of foot shock when the tone occurs. Therefore the model should have prior probabilities that favor hypotheses of no outcome. This is achieved by setting the prior probabilities proportionally to normal densities centered on -5 instead of zero, that is, the prior probability of a hypothesis is $\prod_i \mathrm{norm}(w_i; -5, 5)$.

Locally Bayesian learning

When a stimulus is presented, activation propagates up the network according to Equations 16, 17 and 18. When the correct outcome, t_{out}, is presented, the outcome layer does Bayesian updating of its hypothesis space. Formally, the updated probability of a particular hypothesis W'_{out} is

$$p(W'_{out}|t_{out}, x_{out}) \quad = \quad \frac{p(t_{out}|W'_{out}, x_{out})\; p(W'_{out})}{\sum_{W_{out}} p(t_{out}|W_{out}, x_{out})\; p(W_{out})} \qquad (19)$$

where the sum in the denominator is over all outcome weight matrices in the hypothesis space. This is simply Equation 14 re-written in the notation of this specific model architecture.

After the outcome hypothesis space is updated, it is used to propagate a target to the hidden space. Rewriting Equation 15 in the notation of the specific model architecture yields:

$$t_{hid} \quad = \quad \underset{x^*_{out}}{\mathrm{argmax}} \sum_{W_{out}} p(t_{out}|W_{out}, x^*_{out})\; p(W_{out}|t_{out}, x_{out}). \qquad (20)$$

The target for the hidden layer, t_{hid}, is going to be used for Bayesian updating of the hidden layer, and therefore t_{hid} must be in the range of values for which $p(t_{hid}|W_{hid}, x_{hid})$ is defined. Therefore t_{hid} must be in $\{0,1\}^N$, and the maximization in Equation 20 explores all $x^*_{out} \in \{0,1\}^N$.

Once the hidden target is found, the hidden-layer hypothesis space is updated, using Bayes formula analogous to Equation 19 with all the *out* subscripts replaced by *hid* subscripts.

The analogous globally Bayesian model

One of the motivations for illustrating the locally Bayesian approach with such a simple model is that the corresponding globally Bayesian model is small enough to be easily simulated. Therefore the behaviors of the locally and globally Bayesian models can be directly contrasted.

In the analogous globally Bayesian model, each hypothesis consists of a specific combination of hidden weight matrix and outcome weight matrix. The global hypothesis space was constructed by crossing every hidden-weight matrix in the locally Bayesian model with every outcome-weight matrix. When there are N cues, the globally Bayesian model has $2^{(N^2)} \times 3^N$ hypotheses, whereas the locally Bayesian model has a total of $2^{(N^2)} + 3^N$ hypotheses. For example, when $N = 3$, the globally Bayesian model has 13,824 hypotheses, whereas the locally Bayesian model has 539 hypotheses. The prior probability on a hypothesis in the global model was set to the product of the prior probabilities of each corresponding hypothesis in the local model: $p_{global}(W_{out}, W_{hid}) = p_{local}(W_{out})\, p_{local}(W_{hid})$. This assignment makes the marginal priors of each hidden weight value identical for the local and global models.

The probability of each specific outcome, for a specific hypothesis, is generated according to Equation 9, which can be re-written in the specific model's notation as

$$p(y_{out}|W_{out}, W_{hid}, x_{hid}) = \sum_{y_{hid}\in\{0,1\}^N} p(y_{out}|W_{out}, y_{hid})\; p(y_{hid}|W_{hid}, x_{hid}) \tag{21}$$

where the component probabilities were defined in Equations 16 and 18. To compute the marginal probability of each specific outcome, we marginalize across all possible hypotheses as in Equation 6, where the integral over parameter values becomes a sum over discrete hypothesized weight matrices. To compute the expected value of the outcome, we then marginalize across possible outcome values as in Equation 7. When a teacher t_{out} is supplied for the outcome, then Equation 21 provides the likelihood of each hypothesis, $p(t_{out}|W_{out}, W_{hid}, x_{hid})$. Those likelihoods are used in Bayes' theorem to update the global hypothesis space as expressed earlier in Equation 8, where the denominator is computed from Equation 6 as a sum over discrete hypothesized weight matrices instead of an integral over continuous parameter values.

Application to highlighting

In this and the following sections, the model will be applied to various phenomena in associative learning that are challenging for either time-independent Bayesian models or error-driven models in the Rescorla-Wagner and connectionist traditions. Trial order effects are difficult for many Bayesian models to address but natural for error-driven models, whereas retrospective revaluation of absent cues is difficult for error-driven models to address but natural for Bayesian models.

Table 1
Canonical highlighting design.

Phase	# blocks	Items × Frequency	
First	N_1	I.PE→E ×2	
Second	N_2	I.PE→E ×3	I.PL→L ×1
Third	$N_3 = N_2 + N_1$	I.PE→E ×1	I.PL→L ×3
Test		PE.PL→? (L) I→? (E)	

Note: An item is shown in the format, Cues→Correct Response. In the test phase, typical response tendencies are shown in parentheses.

Table 1 shows a canonical *highlighting* design. The learner first sees trials of cues I and PE indicating outcome E, denoted I.PE→E in Table 1. In the second and third phases of training, trials of I.PL→L are intermixed. Notice that cue I is an Imperfect predictor because both outcomes E and L can occur when I occurs. Cue PE is a Perfect predictor of the Earlier trained outcome E, and cue PL is a Perfect predictor of the Later trained outcome L. If people learn the simple underlying symmetry of the cue-outcome correspondences, then when they are tested with cue I by itself, they should choose outcomes E and L equally often. In fact, there is a strong tendency to choose outcome E (shown in the bottom row of Table 1). This response bias is not a general primacy effect, however, because when people are tested with the pair of cues PE and PL, they prefer outcome L. Apparently, cue PL has been highlighted during learning I.PL→L, so that cue I is not associated strongly with L but PL is.

The canonical highlighting design equalizes the long-run base rates of the early and late outcomes. Notice in the table that when $N_3 = N_2 + N_1$, the total number of I.PE→E trials is $3N_1 + 4N_2$, which equals the total number of I.PL→L trials. Thus, the base rates of E and L trials are equal. This equality of base rates distinguishes highlighting from the inverse base rate effect reported by Medin and Edelson (1988), which uses only the second phase of Table 1, i.e., $N_1 = 0$ and $N_3 = 0$. The equality of base rates emphasizes that highlighting is an order-of-learning effect, not a base rate effect. It is only by virtue of the fact that the I.PE cases are learned before the I.PL cases that asymmetric test responding occurs at all. If the I.PE and I.PL cases were intermixed equally throughout training, they would be structurally equivalent and no such highlighting effect could be meaningfully assayed (except for differences in acquisition order within individual subjects).

Highlighting or the inverse base rate effect have been obtained in many different experiments using different stimuli, procedures, and cover stories, such as fictitious disease diagnosis (Kruschke, 1996; Medin & Edelson, 1988), random word association (Dennis & Kruschke, 1998), and geometric figure association (Fagot, Kruschke, Depy, & Vauclair, 1998). Many other published experiments have obtained the inverse base rate effect for different relative frequencies and numbers of training blocks (e.g., Juslin, Wennerholm, & Winman, 2001; Medin & Bettger, 1991; Shanks, 1992). I have run several (unpublished) experiments in my lab in which $N_1 = 0$ and $N_2 = N_3$, and in all of these experiments robust highlighting has been obtained. Highlighting has not yet been observed in animal learning, though to my knowledge it has been sought in only one study (Fagot et al., 1998).

Highlighting is not Bayesian. The canonical design is a critically difficult case for time-independent Bayesian approaches because highlighting is a trial-order effect. Any Bayesian model that treats the training items exchangably will, by definition, fail to show highlighting. Appendix B provides a Bayesian derivation that $p(E|I) = p(E)$ for any values of N_1, N_2, and N_3. With additional assumptions about unobserved contingencies, another derivation in Appendix B shows that $p(E|PE.PL) = p(E)$ for any values of N_1, N_2, and N_3.

Highlighting has been explained by rapid shifts of attention during learning, and the learning (i.e., retention) of those shifts. The theory has been implemented in error-driven, connectionist models called ADIT and EXIT (e.g., Kruschke, 1996, 2001a, 2005; Kruschke, Kappenman, & Hetrick, 2005). The present model captures some of the same ideas as the EXIT model, but with weight changes driven by Bayesian updating and with attentional shifts driven by maximization of outcome probability. As mentioned earlier, I have also simulated a Bayesian version of EXIT, with results similar to those of the simpler model reported here.

Simulation results. Figure 3 shows graphically the results of training in the highlighting procedure. The upper row shows results from the locally Bayesian model, and the lower row shows results from the globally Bayesian model. I simulated a simple case in which seven trials of I.PE→E were followed by seven trials of I.PL→L. The exact number of trials and their exact relative order is not important to the qualitative outcome, as long as some trials of I.PE→E occur first. Therefore I front loaded all the I.PE→E trials so that the training lists would be easy to read in the left panels of Figure 3.

The right panels of Figure 3 show the behavior of the models after training. Notice that accuracy is good on the training items: There is a very high probability of responding E to I.PE, and a very low probability of responding E to I.PL. The locally Bayesian model shows robust highlighting, with $p(E) > 50\%$ for cue I alone, and $p(E) < 50\%$ for cue pair PE.PL. The globally Bayesian model shows no highlighting, however, with $p(E) = 50\%$ for both I and PE.PL. This lack of highlighting by the globally Bayesian model is exactly what we should expect, based on the Bayesian analyses of the highlighting paradigm discussed earlier.

Figure 3 also shows aspects of the posterior probabilities on the hypothetical weights. The graphs denote the three cue nodes as PE, I and PL. The corresponding hidden nodes are denoted hidPE, hidI and hidPL. The outcome node is denoted E, but it also represents outcome L when the node's value is zero. A connection to the outcome node from the hidden node hidPE is denoted E←hidPE. A connection to the hidden node hidPE from the cue node I is denoted hidPE←I.

The second panels from the left in Figure 3 show the marginal belief probabilities for values of the outcome weights. For example, the marginal probability of value -5 on weight E←hidPL is the sum of the posterior probabilities of all hypotheses that have a value of -5 on weight E←hidPL. For locally Bayesian learning, we see that the posterior probabilities on the output weights are asymmetric for hidPE and hidPL. There is high probability that E←hidPL is -5, but not as high a probability that E←hidPE is $+5$. The output weights are also asymmetric from hidI: There is higher probability that E←hidI is $+5$ than -5. For globally Bayesian learning, however, no such asymmetry occurs. Globally Bayesian learning accurately reflects the underlying symmetry of the training cases.

The third panels from the left in Figure 3 show the marginal belief probabilities for values of the hidden weights. For locally Bayesian learning, the distributions are asymmetric. In particular, the weight hidI←PL has very high probability on value -5, which means that PL inhibits hidI, whereas the weight hidI←PE has very high probability on value 0, which means that PE does not inhibit hidI. For globally Bayesian learning, no such asymmetry exists. To reiterate, globally Bayesian learning accurately reflects the symmetry of the training cases.

The locally Bayesian model exhibits highlighting because of the hidden targets it generates. When presented with a case of I.PL→L (after earlier training on I.PE→E), the best target for the hidden layer sets hidI at zero. Thus, the hidden layer learns to map I.PL to hidPL (not to hidI.hidPL). In the language of attentional learning theory (Kruschke, 2003a), the model has rapidly shifted attention to PL away from I, and then has learned to reproduce that shift in response to the stimulus I.PL. In more Bayesian terms, the model has generated an internal target that is maximally consistent (or least inconsistent) with its current beliefs, and then done Bayesian learning with that target. The locally Bayesian model first shifts the data (i.e., its internal targets) to best fit its beliefs, and only then changes its beliefs to accommodate the (shifted) data.

Figure 3 also reveals that the locally Bayesian model learns the training items faster than the globally Bayesian model. Accuracy on the I.PE and I.PL training items is better for the locally Bayesian model than for the globally Bayesian model. This advantage for the locally Bayesian model appears to be only slight in Figure 3 because the response proportions are compressed against the limits of the possible range, but the the accuracy advantage for the locally Bayesian model is quite large on the initial trials of both phases. The locally Bayesian model improves its accuracy faster than the globally Bayesian model on every training trial.

This faster learning by the locally Bayesian model might

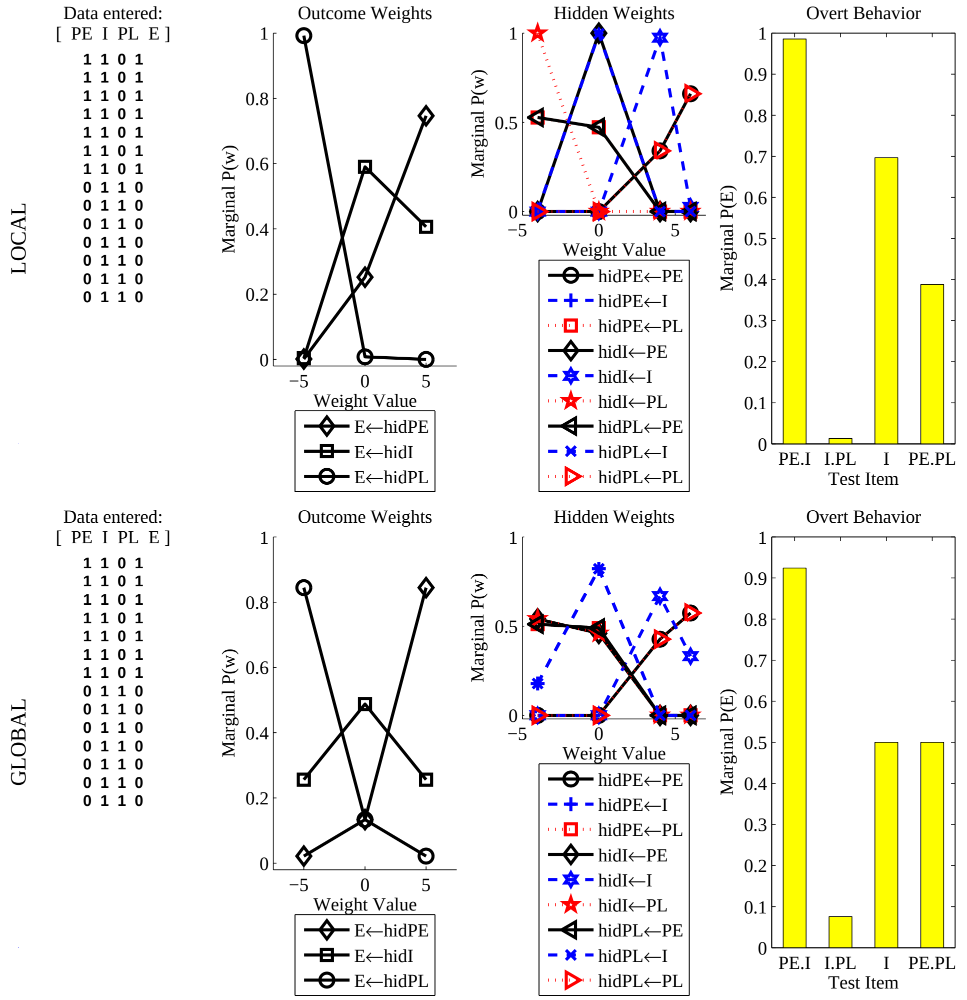

Figure 3. The models trained in the highlighting procedure. *Upper Row:* The locally Bayesian model. *Lower row:* The globally Bayesian model.

seem unexpected, because globally Bayesian models are supposed to be "optimal" learners. The locally and globally Bayesian models have exactly analogous hypothesis spaces, exactly analogous priors, and the same training items. So how can the globally Bayesian model, which accurately learns the underlying symmetry of the items, learn more slowly than the locally Bayesian model?

At least part of the answer is that the globally Bayesian model retains some belief in hypotheses that are eliminated by the internal targets in the locally Bayesian model, and the lingering uncertainty of the global model leaves it less accurate. Specifically, during the early training phase (I.PE→E), the globally Bayesian model retains some belief in the joint hypothesis that (a) hidI is inhibited by PE and (b) hidPE is inhibited by I and (c) hidI and hidPE both inhibit outcome E. In other words, the globally Bayesian model retains some belief in the joint hypothesis that cues I and PE prevent each other from inhibiting outcome E. This hypothesis (among others) dilutes the marginal accuracy when tested with I.PE. The locally Bayesian model, on the other hand, quickly squelches any beliefs in inhibition among I and PE. This is because, in early training, the hidden target specifies full activation of both hidI and hidPE, which is inconsistent with any inhibition to hidI from PE or to hidPE from I. The robust average activation of hidI and hidPE also quickly squelches belief in inhibitory links to outcome E from hidI or hidPE.

In the late phase of training (I.PL→L), the locally Bayesian model retains and enhances its early lead by specifying a hidden target that is maximally consistent with its earlier learning. For I.PL→L, the hidden target puts zero activation on hidI because the currently believed hypotheses assert that hidI indicates outcome E, not outcome L. This shift of activation away from hidI allows the model to retain high accuracy on I.PE while quickly learning I.PL without interference from cue I. This acceleration of learning the later cases, caused by the shift of activation away from hidI, is directly analogous to the acceleration of learning in the EXIT model emphasized by Kruschke (2003b). The EXIT model learns faster when there are learned shifts of attention than when there are not shifts of attention, because interference between training items is reduced by the attentional shifts.

Application to blocking and backward blocking

Blocking (Kamin, 1968) occurs when the early phase of learning has trials of A→X and the later phase has trials of A.B→X. Notice that in the second phase, the same outcome is indicated by an additional cue. In subsequent tests, B elicits a weaker X response than it would have if only the A.B→X trials had been trained. Thus, the previous training of A→X seems to have mitigated, or blocked, learning about B in the subsequent A.B→X trials. Blocking is a crucial phenomenon for all models of learning to address, because it is observed in a wide variety of procedures and species, and it appears to disconfirm any model that merely counts co-occurrences of cues and outcomes (but cf. Miller & Matzel, 1988).

Backward blocking is an analogous phenomenon that occurs when the phases of training are reversed. The first phase involves A.B→X trials and the later phase involves A→X trials. Despite the fact that B is absent in the second phase, it loses strength. Typically, however, the amount of reduction in backward blocking is weaker than the amount of reduction in forward blocking (see e.g., Beckers, De Houwer, Pineño, & Miller, 2005; Kruschke & Blair, 2000; Lovibond, Been, Mitchell, Bouton, & Frohardt, 2003; Pineño, Urushihara, & Miller, 2005; Shanks, 1985). This asymmetry in strengths of forward and backward blocking is a trial order effect that is challenging for extant time-independent Bayesian approaches.

There have been several previous theories of blocking and backward blocking (for reviews, see De Houwer & Beckers, 2002b; Dickinson, 2001). One type of theory involves error-driven associative learning, whereas another type of theory involves Bayesian learning. The error-driven associative learning models descend from the Rescorla-Wagner (1972) model. The Rescorla-Wagner model handily accounts for blocking, but because it assumes absent cues have zero influence on learning, it cannot account for backward blocking. Extensions of the model, that assume absent cues have a negative impact on learning, can account for backward blocking or other effects (Dickinson & Burke, 1996; Ghirlanda, 2005; Markman, 1989; Tassoni, 1995; Van Hamme & Wasserman, 1994). The models assert that only absent cues that are expected to be present should have negative impact, but the exact computations regarding which cues are expected, and their magnitude of negativity, have been left unspecified.

The Bayesian models of backward blocking (e.g., Gopnik et al., 2004; Sobel et al., 2004; Tenenbaum & Griffiths, 2003) show the effects by shifting belief probability over hypotheses about cue-outcome correspondences. For example, suppose the model has three hypotheses: $A \Rightarrow X$ (and $B \not\Rightarrow X$), $B \Rightarrow X$ (and $A \not\Rightarrow X$), and $A \vee B \Rightarrow X$. There is a belief probability for each of the three hypotheses, and, crucially, the sum of the belief probabilities must always be 1.0 because the hypotheses are mutually exclusive and exhaustive. Therefore, if belief in any one hypothesis increases, belief in the other hypotheses must decrease. The model accounts for backward blocking because in the second phase, as belief in $A \Rightarrow X$ increases, belief in $B \Rightarrow X$ decreases, so that subsequent testing with B alone shows reduced strength of X responding. This is the everyday logic of exoneration, mentioned in the introduction of the article: If A is responsible, then B is exonerated. Unfortunately, because these Bayesian models are trial-order invariant, these models show no difference in strength between forward blocking and backward blocking. The Bayesian approach has been applied to situations in which human learning occurs in just a few trials, unlike the error-driven associative models which originally focussed on animal learning that occurs across many trials. The extension of the Bayesian approach, to phenomena of highlighting and relative magnitudes of forward and backward blocking, is a different goal than that of previous Bayesian modelers.

The locally Bayesian model introduced here, however, is sensitive to trial order. The Bayesian updating of belief in associative hypotheses generates backward blocking just as in globally Bayesian approaches. In the locally Bayesian model, however, the selection of asymmetric targets in the hidden layer generates stronger forward blocking than backward blocking. Figure 4 shows the results of the locally Bayesian model for backward and forward blocking. The top row shows the state of the model after a few trials of training on A.B→X. Notice in the right panel the percentage of X responses given to B alone: $p(X|B) \approx .77$. This level of responding is the baseline against which forward and backward blocking are to be judged.

The middle row of Figure 4 shows the locally Bayesian model after backward blocking, that is, after training continued with A→X. Notice in the right panel that $p(X|B)$ has dropped to about .61 despite the fact that B never occurred in the second phase of training. Notice also that the weight to X from hidB (denoted X←hidB) has it average belief probability shifted down dramatically: In the top row the modal X←hidB value is 5, whereas in the middle row the modal value is 0. This backward blocking is the result of Bayesian belief updating, analogous to that reported by previous researchers (e.g., Tenenbaum & Griffiths, 2003).

The bottom row of Figure 4 shows results for forward blocking in the locally Bayesian model. The left panel shows that the same items were trained as in backward blocking (cf. the middle row); merely their order was reversed so that all the A→X items came first instead of last. The right panel shows that $p(X|B) \approx .35$, which is clearly lower than that observed after backward blocking. In other words, blocking is stronger in forward blocking than in backward blocking.

The locally Bayesian model shows stronger forward blocking because the hidden targets suppress hidB during A.B training. This suppression can be observed in the bottom row's third panel, which graphs the hidden weights. The modal value for hidB←A is -4, which means that A is inhibiting hidB. The suppression of hidB is beneficial because after A→X training, there is significant belief in all outcome-weight hypotheses that have positive values for X←hidA, including the specific hypothesis that X←hidA is positive $(+5)$

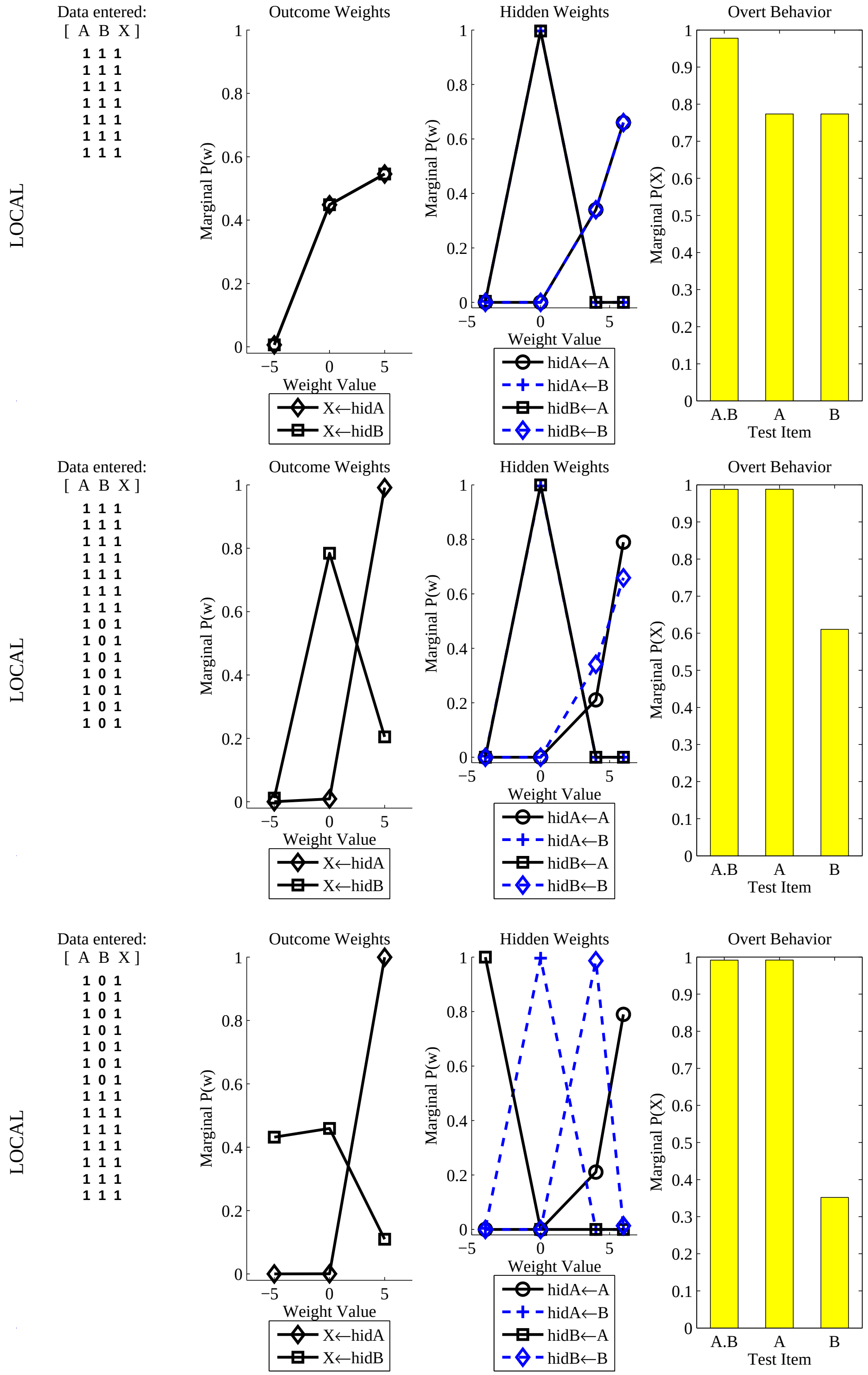

Figure 4. The locally Bayesian model applied to backward or forward blocking. *Top row:* After early training in backward blocking (or unovershadowing). *Middle row:* After backward blocking. *Bottom row:* After forward blocking.

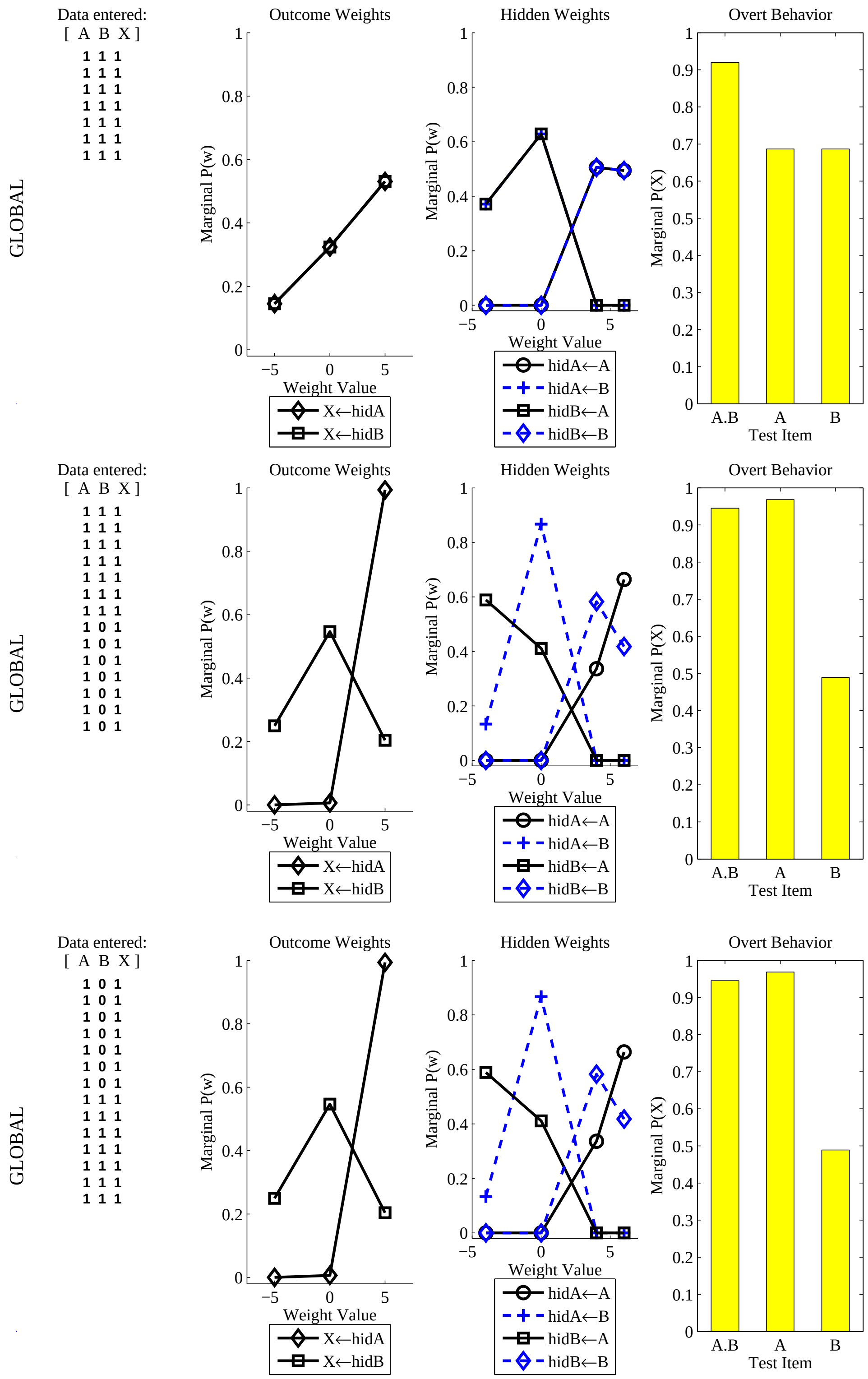

Figure 5. The *globally* Bayesian model applied to backward or forward blocking. *Top row:* After early training in backward blocking (or unovershadowing). *Middle row:* After backward blocking. *Bottom row:* After forward blocking.

and X←hidB is negative (−5). If hidB is allowed to be active when cues A.B appear, outcome X is inhibited by the currently believed hypothesis that X←hidB is negative (−5). Therefore, it is maximally consistent with current beliefs to suppress hidB in the hidden layer target.

Figure 5 shows behavior of the analogous globally Bayesian model. The top row shows the state of the model after the first phase of backward blocking. The degree of responding to B-alone is the baseline against which forward and backward blocking are to be judged. The middle row shows results from backward blocking, and the bottom row shows results from forward blocking. There is no difference in overt behavior, or marginal weights, between forward and backward blocking. This lack of difference is expected, of course, because the globally Bayesian model should be invariant under changes in trial order, and indeed it is. This insensitivity is undesirable as a model of human performance, which, as described earlier, can show greater forward blocking than backward blocking.

A comparison of Figures 5 and 4 reveals that the locally Bayesian model learns its training items faster than the globally Bayesian model. Notice the accuracy of the models on the training item, A.B. The locally Bayesian model shows higher accuracy than the globally Bayesian model. This advantage for the locally Bayesian model is quite strong in the early trials and persists throughout training. Thus, despite the fact that the globally Bayesian model learns optimally over the joint hypothesis space, the locally Bayesian model learns the training items faster.

In summary, the locally Bayesian model shows backward blocking that is difficult for error-driven associative models, and the locally Bayesian model shows sensitivity to trial order that is difficult for globally Bayesian models. The locally Bayesian model also learns faster than the analogous globally Bayesian model.

There is no claim here that this model comprehensively explains the myriad results surrounding blocking and backward blocking. Despite the strong attentional learning in forward blocking, there is no account here of learned attention in backward blocking (Kruschke & Blair, 2000). There is no account of the dependence of backward blocking on within-compound associations (Dickinson & Burke, 1996; Wasserman & Berglan, 1998). There is no account of second-order blocking (De Houwer & Beckers, 2002a), or of the influence of additive targets (Lovibond et al., 2003), or of spontaneous recovery from blocking (Pineño et al., 2005). Thus, the claims here are modest: What has been shown is one way to combine Bayesian and attentional approaches to ameliorate some of their individual inadequacies. The general discussion explores potential expansions of the Bayesian framework to address some of the more elaborate phenomena. Blocking, as a behavioral effect, is probably generated by many different underlying mechanisms. Any model of all the effects cited above will probably involve a combination of several mechanisms.

Application to unovershadowing and backward conditioned inhibition

Overshadowing is the phenomenon that after training with A.B→X, responses to *B* alone are weaker than if *B* were trained by itself, without being accompanied by *A*. Apparently the presence of *A* has overshadowed *B*. In *unovershadowing,* a.k.a. recovery/release from overshadowing, after the A.B→X training there are trials of A→ ¬X, i.e., cue *A* alone leading to the *absence* of outcome *X*. When *B* is then tested alone, it elicits outcome *X* more strongly, despite the fact that it never occurred in the later phase of training (Beckers et al., 2005; Kaufman & Bolles, 1981; Larkin, Aitken, & Dickinson, 1998; Lovibond et al., 2003; Matzel, Schachtman, & Miller, 1985; Melchers, Lachnit, & Shanks, 2004; Wasserman & Berglan, 1998). Unovershadowing is the Holmesian logic of eliminating the impossible, mentioned in the introduction of this article: When cue A is not responsible, then cue B must be.

Figure 6 shows the results of the local Bayesian model applied to unovershadowing. Notice that the strength of responding to B alone (right-most panel) is higher than after the first phase of training, which can be reviewed in the top row of Figure 4. In other words, there is robust unovershadowing. Notice also in the middle-left panel that the probability mass for weights to X from hidB (X←hidB) has shifted to the right (compared with Figure 4), i.e., the mean weight has increased, despite the fact that B did not appear in later training.

Unlike other extant Bayesian models, the locally Bayesian attention model predicts different behaviors when the phases of training are reversed. In reversed unovershadowing, A→ ¬X is followed by A.B→X. In the locally Bayesian model, final responses to A.B are enhanced, and responses to A are even lower, than for unovershadowing. This is because on A.B→X trials, the hidden target sets hidA to zero, and so the model learns to believe a negative weight on hidA←B. Essentially, when cues A.B are presented, the hidden layer only "sees" hidB, and so the response to A.B is virtually as strong as the response to B alone. In regular unovershadowing, however, the model (in its present form) cannot retrospectively learn a negative weight on hidA←B, and so the impact of A continues to have influence during later A.B tests. Thus, this particular simplistic model predicts that responding to A.B should be strong after reversed unovershadowing, but weaker after unovershadowing. I am not aware of any published data that directly address this prediction, because the A.B combination typically is not tested at the end of unovershadowing. The main point here is to illustrate again the fact that the locally Bayesian model is sensitive to trial order. The particular predictions of this simplistic implementation are not essential to the success or failure of the general framework.

The globally Bayesian model also exhibits robust unovershadowing, of course, as can be seen in the lower row of Figure 6. The test performance on cue B-alone is stronger than after the first phase, shown in the top row of Figure 5. Unlike the locally Bayesian model, the globally Bayesian model pre-

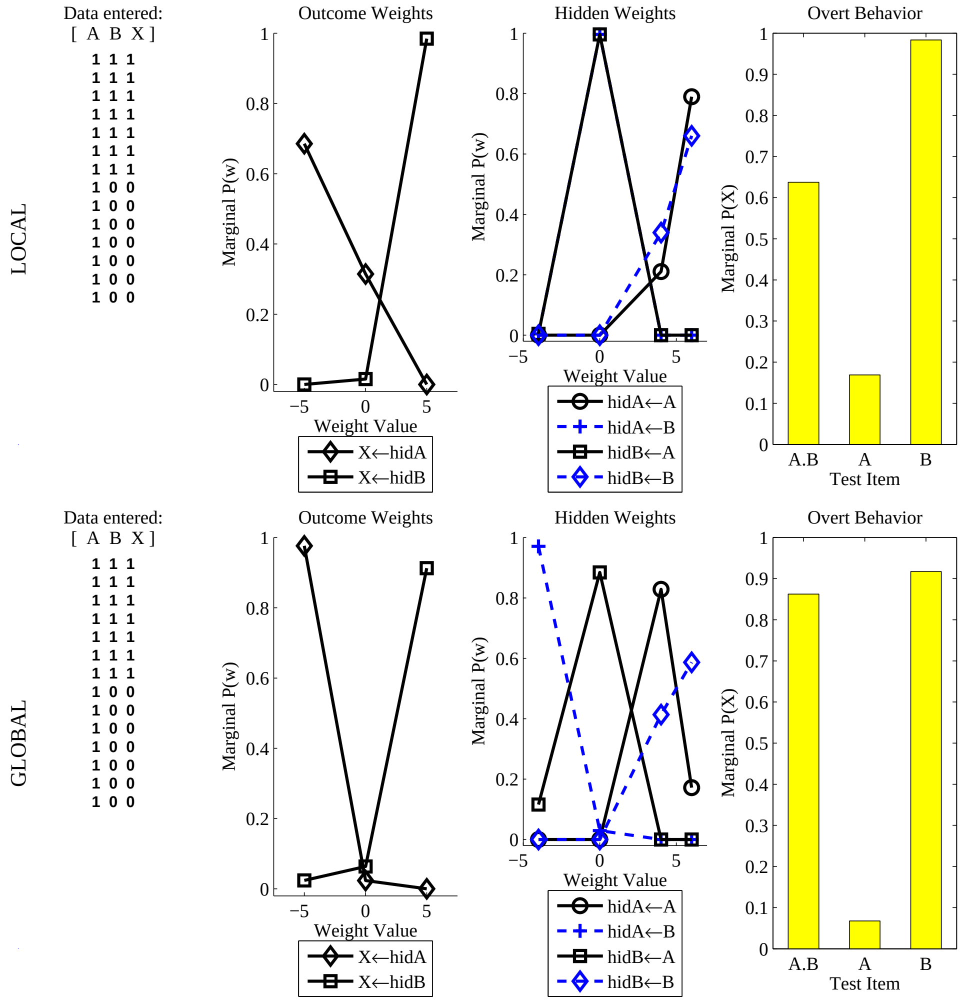

Figure 6. The models after later training in unovershadowing. Top row is locally Bayesian, to be compared with the top row of Figure 4. Bottom row is globally Bayesian, to be compared with the top row of Figure 5.

dicts equal performance for unovershadowing and reversed unovershadowing, because the globally Bayesian model is not sensitive to trial order.

Interestingly, the globally Bayesian model shows selectively faster learning during the second phase of unovershadowing than the locally Bayesian model. Accuracy on the second-phase item, A→ ¬X, grows faster for the globally Bayesian model than for the locally Bayesian model. This result can be seen in the right panels of Figure 6, where $p(X|A)$ is smaller for the global model than for the local model. The reason the global model learns this case faster is that it has a hypothesis in its joint hypothesis space that is well-tuned to the training case, but which is not available to the locally Bayesian model with its separate hypothesis spaces. The global joint space has hypotheses in which the weight to X from hidA (X←hidA) is negative and the weight to hidA from B (hidA←B) is negative. These hypotheses retain some belief during the first phase of training in the global model (because the presence of B in early training inhibits hidA, so the negative weight to X from hidA is not felt). That combination of weights is then perfect to accommodate the training item in the second phase, which needs an inhibitory weight from hidA to X. So the global model, with its joint hypotheses, is poised to learn the second phase quickly. The marginal weights in the lower row of Figure 6 do indeed show large probabilities loaded onto negative values of X←hidA and hidA←B. The locally Bayesian model, on the other hand, does not allow the double negative combination (i.e., negative X←hidA and negative hidA←B) to survive the first phase of training, because the layers operate separately. During the first phase, the locally Bayesian model shifts belief away from negative weights in the outcome layer and in the hidden layer, and therefore the model is at a disadvantage when confronting the second phase of learning.

In summary, the locally Bayesian model shows robust unovershadowing, which is difficult for error-driven connectionist models (including the Rescorla-Wagner model) to address. The locally Bayesian model predicts differences between unovershadowing and reversed unovershadowing because of trial-order sensitivity. The unovershadowing paradigm is one in which the analogous globally Bayesian model learns faster than the locally Bayesian model.

In *conditioned inhibition*, the first phase of training involves trials of A→X. The second phase of training has trials of A.B→ ¬X. The result is that *B* becomes an inhibitor of response X. In *backward conditioned inhibition*, the phases of training are reversed. *B* becomes an inhibitor in this case too (Chapman, 1991; Larkin et al., 1998; Melchers et al., 2004).

Backward conditioned inhibition is the same structure as unovershadowing, but with the roles of the outcomes reversed: What was a present outcome is now an absent outcome and vice versa. Conditioned inhibition is trickier to assess behaviorally, however, than unovershadowing. Whereas unovershadowing results in an increase in a response, which can be directly observed, backward conditioned inhibition results in a lack of response, which can only be indirectly observed. Traditional indirect tests of conditioned inhibition include the summation and retardation tests (Rescorla, 1969). Fortunately, when assessing the model, we do not need to rely merely on overt responses to infer hidden inhibitory links, because we can peer inside the model and see the actual associative strengths.

Results of the model applied to backward conditioned inhibition are shown in Figure 7. Cue E is an excitor and cue I is an inhibitor (not to be confused with the imperfectly predictive cue in highlighting). As expected, robust backward conditioned inhibition is obtained, for both locally and globally Bayesian models. Cue I becomes a strong inhibitor despite its absence in the later phase of training. The right panel shows that the response to cue I has dropped well below its prior baseline level. The middle-left panel shows that the probability mass on the weight to X from hidI (X←hidI) has shifted to extreme negative values relative to its prior distribution, despite the fact that cue I was absent during later training.

The lower row of Figure 7 shows results from the globally Bayesian model. The globally Bayesian model learns the second phase of backward conditioned inhibition faster than the locally Bayesian model. The explanation for the advantage is analogous to the case of unovershadowing. In the joint hypothesis space of the global model, hypotheses with negative hidE←I weights and positive X←hidE weights survive the first phase of training. These hypotheses are then poised to accommodate the demands of the second phase. No analogous combination exists in the local model that can survive the first phase of training. In the local model, all positive weights to X from hidE are squelched in the first phase, and so the local model begins the second phase at a disadvantage.

In summary, the cases of unovershadowing and backward conditioned inhibition have reiterated points made for highlighting and blocking while also revealing more about the relation of local and global models. The reiterated points are that the Bayesian models exhibit retrospective revaluation effects that error-driven connectionist models find difficult, and the locally Bayesian model shows trial order effects that the globally Bayesian model cannot. The newly revealed point is that the globally Bayesian model can sometimes learn later items faster than the locally Bayesian model, when the global model retains joint hypotheses that the local model is not able to sustain because its layers operate separately. Thus, the local model can learn faster than the global model when the hidden targets reduce interference between new and old items, but the global model can learn faster when it has a joint hypothesis that fits the items well but which the local model cannot sustain.

Discussion

Attention is crucial

Elsewhere (e.g., Kruschke, 2003a) I have argued that an essential mechanism in human associative learning is rapid shifting of attention across cues and the learning (retention) of those shifts. For a given stimulus, the cognizer learns associations from the stimulus to an allocation of attention across the stimulus cues, and the cognizer learns associations from the attentionally filtered cues to the overt responses or outcomes. An illustration of these two layers of associations appeared in Figure 2. A key part of the learning process is the allocation of attention, for which there is no external teacher. Instead, the learner must infer an attentional allocation. What has been offered in the present article is a locally Bayesian approach to learning the associations in the two layers, along with a way of generating an allocation of attention.

The simplistic locally Bayesian model illustrated above can only make probabilistic copies of the input cues at its hidden layer. Its hidden layer cannot represent cue combinations. Therefore, the model cannot learn non-linear mappings from cues to outcomes, such as the exclusive-or (XOR). In principle, the model could be expanded such that its hidden layer includes higher-order cue combinations. One variation of that approach is for the hidden node hypothesis space to include a random smattering of weighted cue combinations, much like the typical random starting weights of traditional or Bayesian backpropagation networks (MacKay, 2003; Neal, 1996; Rumelhart et al., 1995; Rumelhart, Hinton, & Williams, 1986) or the random covering map in the original ALCOVE model (Kruschke, 1992). To qualitatively reproduce the simulation results reported above, I believe that the hypothesis space must contain hypotheses that implement the notion of selective attention, whereby cues can be selectively ignored or attended. To exhibit a highlighting effect, the model should be able to suppress its internal representation of cue I when presented with I.PL, and enhance its internal representation of PL. To exhibit strong forward blocking, the model should be able to enhance its internal representation of A when presented with A.B, and suppress its internal representation of B.

Relations to some other models

The idea of attentionally filtered cues was implemented by the hidden layer in the simplistic model above. Another

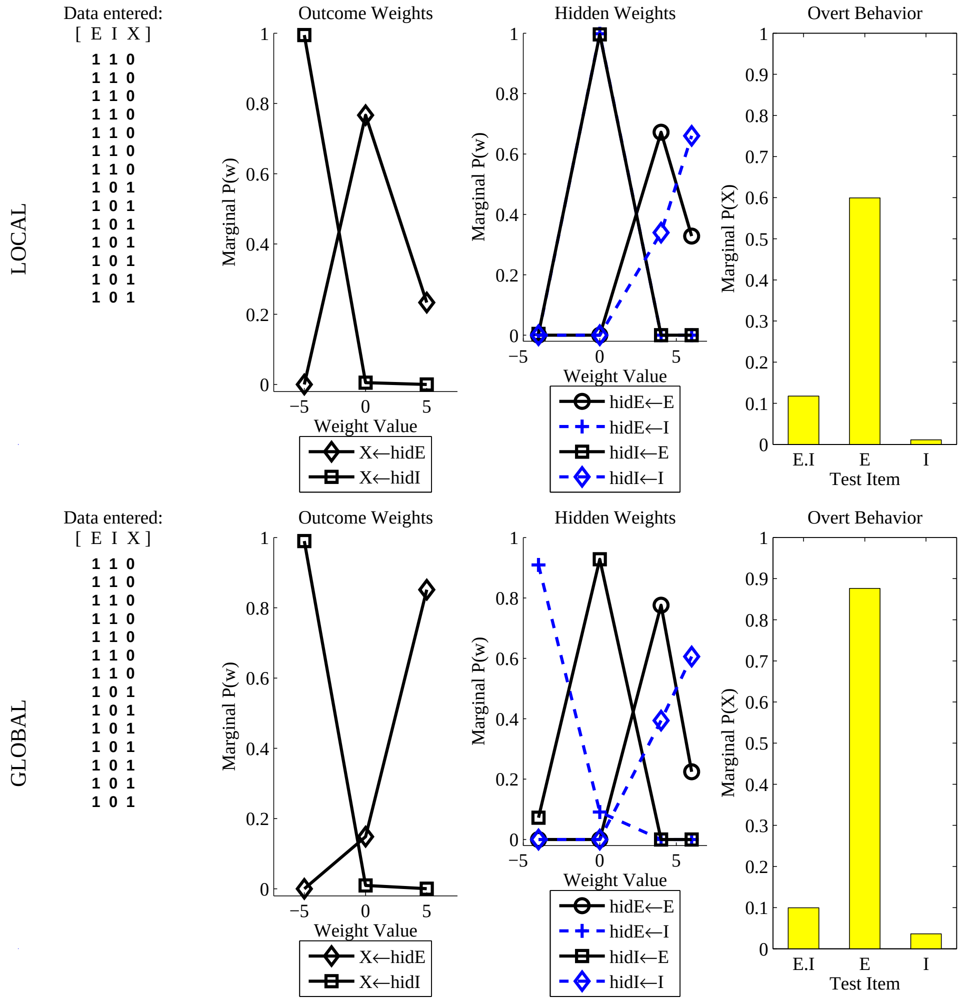

Figure 7. Results from backward conditioned inhibition for the locally Bayesian model (top row) and the globally Bayesian model (bottom row).

way to implement attentional filtering is to represent attention as multiplicative gates on cue activations, and have the model learn about hypothetical weights from cue configurations (a.k.a. exemplars) to attention gates. There is one layer of weights from exemplars to attention gates, and a second layer of weights from attentionally gated cues to outcomes. The two layers of weights are supplied with spaces of hypothetically possible sets of weights, and the degree of belief in each hypothesis is updated trial by trial. This amounts to a locally Bayesian implementation of the EXIT model (Kruschke, 2001a, 2001b). I have simulated the locally Bayesian EXIT model and found it to produce behaviors very much like the behaviors of the simple model above.

The top layer of the simplistic locally Bayesian model is closely related to a Kalman filter, which was introduced to associative learning researchers by Sutton (1992) and has been used to model some aspects of attention in learning by Dayan and collaborators (e.g., Dayan & Kakade, 2001; Dayan et al., 2000; Dayan & Yu, 2003; Kakade & Dayan, 2002). In a Kalman filter, continuous-scale outcomes are computed as a weighted sum of input cues. The weighting coefficients have prior distributions defined as multivariate normal. The Kalman filter uses Bayesian updating to adjust the probability distribution on the weights (Meinhold & Singpurwalla, 1983). Because the model is linear, the posterior distributions on the weights are also multivariate normal, and the Kalman filter equations elegantly express the posterior mean and covariance as a simple function of the prior mean and covariance. There are some similarities to the upper layer of the previous section's simplistic locally Bayesian model, wherein outcomes are computed as a weighted sum of (hidden) cues, the weighting coefficients have prior distributions defined as multivariate normal, and the distributions are updated in a Bayesian fashion. One difference between the models is that the Kalman filter can add a constant amount of noise variance to the weight distributions on every trial. Because of the accumulation of noise across trials, the Kalman filter can exhibit some trial order effects. Simulations presented by previous researchers (Dayan & Kakade, 2001; Dayan et al., 2000; Dayan & Yu, 2003) have apparently set the added noise close to zero, and so trial order effects are modest. When there is no added noise, then of course the Kalman filter shows no difference between forward and backward blocking. Even with added noise, the Kalman filter does not exhibit highlighting (verified by simulations not shown). It may be that more sophisticated models of temporal change (e.g., Dayan & Yu, 2003; Steyvers & Brown, 2005) could show highlighting, but this prospect awaits future research.

The locally Bayesian model extends the Kalman-filter approach by pre-pending an attentional learning layer. Whereas the Kalman filter learns about the cues as they are actually presented, the upper layer of the locally Bayesian model learns only about attentionally filtered cues at the hidden layer. The attentional filtration depends on the temporal order of training items. The temporal dependencies of the two models are not incompatible; future extensions of the models could incorporate both the noise accumulation of the Kalman filter model with the attentional selection of the locally Bayesian model.

The present model's notion of attention is quite different than what Dayan et al. refer to as attention in the Kalman filter. In their approach, the posterior variance on a cue's weight is the attention paid to that cue. Higher variance on a cue's weight denotes higher uncertainty about the cue's impact on the outcome, and uncertainty is supposed to elicit attention in learning. Posterior variances on associative weights can be seen, roughly, in the graphs of outcome weight probabilities. In Figure 4, for example, the bottom row shows the state of the locally Bayesian model after (forward) blocking. The graph of outcome weights shows the weight to X from hidA (X←hidA) has virtually all of its probability mass loaded over a value of $+5$. This distribution has small variance, hence small uncertainty and small "attention" in the sense used by Dayan et al. The weight to X from hidB (X←hidB), however, has its probability mass distributed over all three values of -5, 0 and $+5$. This distribution has higher variance, hence higher uncertainty and higher "attention" in the usage of Dayan et al. My use of the term "attention" is rather different. In the locally Bayesian model, attention is the activation of the cues in hidden layer. During the prediction phase of a trial, before an external teacher is supplied, attention is the mean activation of the hidden layer. During the learning phase, after corrective feedback is supplied, attention is the target at the hidden layer. In the bottom row of Figure 4, the graph of the hidden weight distribution shows that hidB is strongly inhibited by A (i.e., hidB←A has most of its probability mass on a value of -4). The self connection to hidB from B also has most of its probability mass on its smallest allowed value. These weights imply that cue B is largely ignored, not attended to. Cue A, on the other hand, is attended to. Thus, attention as internal cue activation in the locally Bayesian model is quite distinct from attention as weight uncertainty in the Kalman filter model.

In the locally Bayesian model, cues are thought of as inputs, and outcomes are outputs to be predicted. Courville and colleagues (Courville et al., 2004; Courville, Daw, & Touretzky, 2005) instead conceptualized both the cues and outcomes as effects to be predicted by latent causes. In their approach, a hypothesis is a set of weights from latent causes to cues and outcomes, with the probability of each effect being determined by a sigmoidal function of the summed weights from activated latent causes. The hypothesis space consists of many weight combinations, and Bayesian learning shifts belief probability among the hypotheses. Courville et al. (2004) showed that the approach can account for the dependency of conditioned inhibition on the number of trials of training, by virtue of the prior probabilities being gradually overwhelmed by training data. Estimating the Bayesian probabilities is computationally intensive, and their approach involves no temporal encoding, so it suffers from trial-order invariance and can show none of the trial-order dependencies discussed above.

In principle, the latent-cause approach might be applied to each layer in a two-layer architecture with locally Bayesian learning. One set of latent causes would generate the cues and the attentionally filtered cues, while another set of latent causes would generate attentionally filtered cues and outcomes. Targets for the attentionally filtered cues would be selected as in the locally Bayesian scheme explicated above. Presumably such an architecture would generate trial-order phenomena comparable to the simplistic model reported above.

Dynamic creation of hypotheses. The simple model simulated above is not able to exhibit some phenomena related to blocking procedures, such as the dependency of backward blocking on memory for cue combinations (Aitken, Larkin, & Dickinson, 2001; Dickinson & Burke, 1996; Wasserman & Berglan, 1998). This inability in the model is a limitation imposed by the use of simplistic, non-configural associative hypotheses. Future versions of the model might include representations of cue combinations, selectively recruited as needed. In other words, the model must have the right kind of hypotheses, generated on the fly.

Two existing approaches to dynamic hypothesis recruitment include the rational model of categorization by Anderson (1991), and the DAC5 model by Verschure and Althaus (2003). Rigorous predictions for the DAC5 model rely on full-scale robotic simulations far beyond the scope of this article. Both models lack selective attention to cues, however, and probably fail to show highlighting in the canonical design. Anderson (1990, pp. 117-120) applied the rational model to the inverse base rate effect reported by Medin and Edelson (1988), which emerges when using only the second phase of the canonical design in Table 1, where I.PE→E is three times as frequent as I.PL→L throughout the training. The rational model can capture the ordinal trends of the inverse base rate effect, but relies on the categories having different base rates: "[feature] mismatches will be weighed more seriously for high-frequency diseases" (Anderson, 1990, p. 119). Specifically, for probe PE.PL, the missing feature I mismatches both categories, but is weighed more heavily for the high-frequency E category. This difference will not occur in the canonical highlighting design, wherein categories have equal frequencies.

The framework I am expounding here suggests an extension of the rational model that might accommodate highlighting and other attentional effects. Layers of rational models could be used in succession, the first rational model learning to map cues to attentional allocations, and the second rational model learning to map attentionally filtered cues to outcomes. This approach is an especially intriguing prospect for future research, because of its ability to recruit hypotheses in each layer on the fly.

Conclusion

The locally Bayesian attention model achieves trial-order effects by generating internal target data that depend on trial order, without having any time-sensitive functions in its individual layers. The internal targets are selected to be maximally consistent with beliefs learned up to that moment. When learning any new cue-outcome datum, the model first generates internal representations that are least inconsistent with its current beliefs before updating its beliefs. This procedure qualitatively reproduces several challenging phenomena in human associative learning. To sum up, the locally Bayesian model changes the data to fit its beliefs before changing its beliefs to fit the data. Alas, people seem to behave that way too.

References

Aitken, M. R. F., Larkin, M. J. W., & Dickinson, A. (2001). Reexamination of the role of within-compound associations in the retrospective revaluation of causal judgements. *The Quarterly Journal of Experimental Psychology*, *54B*, 27–51.

Anderson, J. R. (1990). *The adaptive character of thought*. Hillsdale, NJ: Erlbaum.

Anderson, J. R. (1991). The adaptive nature of human categorization. *Psychological Review*, *98*(3), 409–429.

Beckers, T., De Houwer, J., Pineño, O., & Miller, R. R. (2005). Outcome additivity and outcome maximality influence cue competition in human causal learning. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *31*, 238–249.

Bolle, R. M., & Cooper, D. B. (1986). On optimally combining pieces of information, with application to estimating 3-D complex-object position from range data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *PAMI-8*(5), 619–638.

Bridle, J. S. (1990). Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition. In F. Fogleman Soulie & J. Herault (Eds.), *Neurocomputing: Algorithms, architectures and applications* (pp. 227–236). Berlin: Springer-Verlag.

Chapman, G. B. (1991). Trial order affects cue interaction in contingency judgment. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *17*, 837–854.

Courville, A. C., Daw, N. D., Gordon, G. J., & Touretzky, D. S. (2004). Model uncertainty in classical conditioning. In S. Thrun, L. K. Saul, & B. Schölkopf (Eds.), *Advances in neural information processing systems* (Vol. 16, pp. 977–984). Cambridge, MA: MIT Press.

Courville, A. C., Daw, N. D., & Touretzky, D. S. (2005). Similarity and discrimination in classical conditioning: A latent variable account. In L. K. Saul, Y. Weiss, & L. Bottou (Eds.), *Advances in neural information processing systems* (Vol. 17). Cambridge, MA: MIT Press.

Danks, D., Griffiths, T. L., & Tenenbaum, J. B. (2003). Dynamical causal learning. In S. Becker, S. Thrun, & K. Obermayer (Eds.), *Advances in neural information processing systems* (Vol. 15, pp. 67–74). Cambridge, MA: MIT Press.

Dayan, P., & Kakade, S. (2001). Explaining away in weight space. In T. Leen, T. Dietterich, & V. Tresp (Eds.), *Advances in neural information processing systems* (Vol. 13, pp. 451–457). Cambridge, MA: MIT Press.

Dayan, P., Kakade, S., & Montague, P. R. (2000). Learning and selective attention. *Nature Neuroscience*, *3*, 1218–1223.

Dayan, P., & Yu, A. J. (2003). Uncertainty and learning. *IETE (Institution of Electronics and Telecommunication Engineers, India) Journal of Research*, *49*, 171–182.

De Houwer, J., & Beckers, T. (2002a). Second-order backward blocking and unovershadowing in human causal learning. *Experimental Psychology*, *49*, 27-33.

De Houwer, J., & Beckers, T. (2002b). A review of recent developments in research and theories on human contingency learning. *The Quartely Journal of Experimental Psychology*, *55B*, 289–310.

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Jounral of the Royal Statistical Society*, *39*, 1–38.

Dennis, S., & Kruschke, J. K. (1998). Shifting attention in cued recall. *Australian Journal of Psychology*, *50*, 131–138.

Dickinson, A. (2001). Causal learning: Association versus computation. *Current Directions in Psychological Science*, *10*, 127–132.

Dickinson, A., & Burke, J. (1996). Within-compound associations mediate the retrospective revaluation of causality judgements. *Quarterly Journal of Experimental Psychology: Comparative & Physiological Psychology*, *49B*, 60–80.

Eckstein, M. P., Abbey, C. K., Pham, B. T., & Shimozaki, S. S. (2004). Perceptual learning through optimization of attentional weighting: Human versus optimal Bayesian learner. *Journal of Vision*, *4*, 1006–1019.

Edwards, W. (1968). Conservatism in human information processing. In B. Kleinmuntz (Ed.), *Formal representation of human judgment.* New York: Wiley.

Fagot, J., Kruschke, J. K., Depy, D., & Vauclair, J. (1998). Associative learning in humans (homo sapiens) and baboons (papio papio): Species differences in learned attention to visual features. *Animal Cognition*, *1*, 123–133.

Gelman, A., Carlin, J. B., Stern, H. S., & Rubin, D. B. (2004). *Bayesian data analysis, 2nd ed.* Boca Raton, Florida: CRC Press.

Ghirlanda, S. (2005). Retrospective revaluation as simple associative learning. *Journal of Experimental Psychology: Animal Behavior Processes*, *31*, 107–111.

Gill, J. (2002). *Bayesian methods for the social and behavioral sciences*. Boca Raton, Florida: CRC Press.

Godden, D. (1976). Transition structure versus commitment in sequential subjective probability revision. *Acta Psychologica*, *40*, 21–28.

Gopnik, A., Glymour, C., Sobel, D. M., Schulz, L. E., Kushnir, T., & Danks, D. (2004). A theory of causal learning in children: Causal maps and bayes nets. *Psychological Review*, *111*, 3–32.

Juslin, P., Wennerholm, P., & Winman, A. (2001). High level reasoning and base-rate use: Do we need cue competition to explain the inverse base-rate effect? *Journal of Experimental Psychology: Learning, Memory and Cognition*, *27*, 849–871.

Kakade, S., & Dayan, P. (2002). Acquisition and extinction in autoshaping. *Psychological Review*, *109*, 533–544.

Kamin, L. J. (1968). "Attention-like" processes in classical conditioning. In M. R. Jones (Ed.), *Miami symposium on the prediction of behavior: Aversive stimulation* (pp. 9–33). Coral Gables, FL: University of Miami Press.

Kaufman, M. A., & Bolles, R. C. (1981). A nonassociative aspect of overshadowing. *Bulletin of the Psychonomic Society*, *18*, 318–320.

Kitzis, S. N., Kelley, H., Berg, E., Massaro, D. W., & Friedman, D. (1998). Broadening the tests of learning models. *Journal of Mathematical Psychology*, *42*, 327–355.

Kruschke, J. K. (1992). ALCOVE: An exemplar-based connectionist model of category learning. *Psychological Review*, *99*, 22–44.

Kruschke, J. K. (1996). Base rates in category learning. *Journal of Experimental Psychology: Learning, Memory and Cognition*, *22*, 3–26.

Kruschke, J. K. (2001a). Toward a unified model of attention in associative learning. *Journal of Mathematical Psychology*, *45*, 812-863.

Kruschke, J. K. (2001b). The inverse base rate effect is not explained by eliminative inference. *Journal of Experimental Psychology: Learning, Memory and Cognition*, *27*, 1385–1400.

Kruschke, J. K. (2003a). Attention in learning. *Current Directions in Psychological Science*, *12*, 171–175.

Kruschke, J. K. (2003b). Attentional theory is a viable explanation of the inverse base rate effect: A reply to Winman, Wennerholm, and Juslin (2003). *Journal of Experimental Psychology: Learning, Memory and Cognition*, *29*, 1396-1400.

Kruschke, J. K. (2005). Learning involves attention. In G. Houghton (Ed.), *Connectionist models in cognitive psychology.* London: Psychology Press.

Kruschke, J. K., & Blair, N. J. (2000). Blocking and backward blocking involve learned inattention. *Psychonomic Bulletin & Review*, *7*, 636-645.

Kruschke, J. K., Kappenman, E. S., & Hetrick, W. H. (2005). Eye gaze and individual differences consistent with learned attention in associative blocking and highlighting. *Journal of Experimental Psychology: Learning, Memory and Cognition*, *31*, 830–845.

Larkin, M. J. W., Aitken, M. R. F., & Dickinson, A. (1998). Retrospective revaluation of causal judgments under positive and negative contingencies. *Journal of Experimental Psychology: Learning, Memory, & Cognition*, *24*, 1331-1352.

Lee, M. D. (2004). A Bayesian analysis of retention functions. *Journal of Mathematical Psychology*, *48*, 310–321.

Lovibond, P. F., Been, S.-L., Mitchell, C. J., Bouton, M. E., & Frohardt, R. (2003). Forward and backward blocking of causal judgment is enhanced by additivity of effect magnitude. *Memory & Cognition*, *31*, 133–142.

Luce, R. D. (1959). *Individual choice behavior*. New York: Wiley.

MacKay, D. J. C. (1992). The evidence framework applied to classification networks. *Neural Computation*, *4*(5), 698–714.

MacKay, D. J. C. (2003). *Information theory, inference & learning algorithms*. Cambridge, UK: Cambridge University Press.

Mackintosh, N. J. (1975). A theory of attention: Variations in the associability of stimuli with reinforcement. *Psychological Review*, *82*, 276–298.

Markman, A. B. (1989). LMS rules and the inverse base-rate effect: Comment on Gluck and Bower (1988). *Journal of Experimental Psychology: General*, *118*, 417–421.

Marr, D. (1982). *Vision*. San Francisco: W. H. Freeman.

Matzel, L. D., Schachtman, T. R., & Miller, R. R. (1985). Recovery of an overshadowed association achieved by extinction of the overshadowing stimulus. *Learning and Motivation*, *16*, 398–412.

Medin, D. L., & Bettger, J. G. (1991). Sensitivity to changes in base-rate information. *American Journal of Psychology*, *104*, 311–332.

Medin, D. L., & Edelson, S. M. (1988). Problem structure and the use of base-rate information from experience. *Journal of Experimental Psychology: General*(117), 68–85.

Meinhold, R. J., & Singpurwalla, N. D. (1983). Understanding the Kalman filter. *American Statistician*, *37*(2), 123–127.

Melchers, K. G., Lachnit, H., & Shanks, D. R. (2004). Within-compound associations in retrospective revaluation and in direct learning: A challenge for comparator theory. *Quarterly Journal of Experimental Psychology: Comparative & Physiological Psychology*, *57B*, 25–53.

Miller, R. R., & Matzel, L. D. (1988). The comparator hypothesis: A response rule for the expression of associations. In G. H. Bower (Ed.), *The psychology of learning and motivation: Advances in research and theory* (Vol. 22, pp. 51–92). San Diego, CA: Academic Press.

Myung, I. J., & Pitt, M. A. (1997). Applying Occam's razor in modeling cognition: A Bayesian approach. *Psychonomic Bulletin and Review*, *4*, 79–95.

Neal, R. M. (1996). *Bayesian learning for neural networks*. New York: Springer.

Neapolitan, R. E. (2004). *Learning Bayesian networks*. Upper Saddle River, NJ: Pearson Prentice Hall.

Palmer, S. E. (1999). *Vision science*. Cambridge, MA: MIT Press.

Pineño, O., Urushihara, K., & Miller, R. R. (2005). Spontaneous recovery from forward and backward blocking. *Journal of Experimental Psychology: Animal Behavior Processes*, *31*, 172–183.

Regier, T., & Gahl, S. (2004). Learning the unlearnable: The role of missing evidence. *Cognition*, *93*, 147–155.

Rescorla, R. A. (1969). Pavlovian conditioned inhibition. *Psychological Bulletin*, *72*, 77–94.

Rescorla, R. A., & Wagner, A. R. (1972). A theory of Pavlovian conditioning: Variations in the effectiveness of reinforcement and non-reinforcement. In A. H. Black & W. F. Prokasy (Eds.), *Classical conditioning II: Current research and theory* (pp. 64–99). New York: Appleton-Century-Crofts.

Rumelhart, D. E., Durbin, R., Golden, R., & Chauvin, Y. (1995). Backpropagation: The basic theory. In Y. Chauvin & D. E. Rumelhart (Eds.), *Backpropagation: Theory, architectures, and applications* (pp. 1–34). Hillsdale, NJ: Erlbaum.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning internal representations by back-propagating errors. In D. E. Rumelhart & J. L. McClelland (Eds.), *Parallel distributed processing* (Vol. 1). Cambridge, MA: MIT Press.

Russell, S. J., Binder, J., Koller, D., & Kanazawa, K. (1995). Local learning in probabilistic networks with hidden variables. In *Proc. fourteenth international joint conference on artificial intelligence* (p. 1146-1152). San Francisco: Morgan Kaufmann.

Shanks, D. R. (1985). Forward and backward blocking in human contingency judgement. *Quarterly Journal of Experimental Psychology: Comparative & Physiological Psychology*, *37B*, 1–21.

Shanks, D. R. (1992). Connectionist accounts of the inverse base-rate effect in categorization. *Connection Science*, *4*, 3–18.

Shanteau, J. (1975). Averaging versus multiplying combination rules of inference judgment. *Acta Psychologica*, *39*, 83–89.

Sobel, D. M., Tenenbaum, J. B., & Gopnik, A. (2004). Children's causal inferences from indirect evidence: Backwards blocking and Bayesian reasoning in preschoolers. *Cognitive Science*, *28*, 303–333.

Steyvers, M., & Brown, S. (2005). Prediction and change detection. In ** (Ed.), *Advances in neural information processing systems* (Vol. **, p. **). **: **.

Steyvers, M., Tenenbaum, J. B., Wagenmakers, E. J., & Blum, B. (2003). Inferring causal networks from observations and interventions. *Cognitive Science*, *27*, 453–489.

Sutton, R. S. (1992). Gain adaptation beats least squares? In *Proceedings of the seventh annual Yale workshop on adaptive and learning systems* (p. 161-166). New Haven, CT: Yale University.

Tassoni, C. J. (1995). The least mean squares network with information coding: A model of cue learning. *Journal of Experimental Psychology: Learning, Memory, & Cognition*, *21*(1), 193–204.

Tenenbaum, J. B., & Griffiths, T. L. (2003). Theory-based causal inference. In S. Becker, S. Thrun, & K. Obermayer (Eds.), *Advances in neural information processing systems* (Vol. 15, pp. 35–42). Cambridge, MA: MIT Press.

Trabasso, T., & Bower, G. (1968). *Attention in learning: Theory and research*. New York: Wiley.

Van Hamme, L. J., & Wasserman, E. A. (1994). Cue competition in causality judgments: The role of nonpresentation of compound stimulus elements. *Learning & Motivation*, *25*(3), 127–151.

Van Wallendael, L. R., & Hastie, R. (1990). Tracing the footsteps of Sherlock Holmes: Cognitive representations of hypothesis testing. *Memory & Cognition*, *18*, 240–250.

Verschure, P. F. M. J., & Althaus, P. (2003). A real-world rational agent: Unifying old and new AI. *Cognitive Science*, *27*, 561–590.

Wasserman, E. A., & Berglan, L. R. (1998). Backward blocking and recovery from overshadowing in human causal judgement: The role of within-compound associations. *Quarterly Journal of Experimental Psychology: Comparative & Physiological Psychology*, *51B*, 121–138.

Appendix A
Trial order invariance

The insensitivity of Bayesian updating to the ordering of data is an elementary result. I provide a proof here, however, in the interest of explicitness. The result follows from Equation 1 and the assumed time-independence of data. To reduce clutter in this derivation, I will suppress the variables x and M from Equation 1. When Equation 1 is applied successively to data $t^{(1)}$ and $t^{(2)}$, in that order, we get

$$\begin{aligned} p(\theta|t^{(1)}) &= p(t^{(1)}|\theta)p(\theta)/p(t^{(1)}) && (22) \\ p(\theta|t^{(2)},t^{(1)}) &= p(t^{(2)}|\theta,t^{(1)})p(\theta|t^{(1)})/p(t^{(2)}|t^{(1)}) && (23) \end{aligned}$$

where, in Equation 23, $p(\theta|t^{(2)},t^{(1)})$ denotes the probability of θ after receiving data $t^{(1)}$ followed by $t^{(2)}$ in that order. Tackling each term of the right-hand side of Equation 23 in turn, we note first that $p(t^{(2)}|\theta,t^{(1)}) = p(t^{(2)}|\theta)$ because $t^{(2)}$ does not depend on $t^{(1)}$. The next term on the right-hand side of Equation 23 is $p(\theta|t^{(1)})$, which we note is given by Equation 22. Finally, for the last term on the right-hand side of Equation 23, we note that $p(t^{(2)}|t^{(1)}) = p(t^{(2)})$ because $p(t^{(2)})$ is assumed to be independent of $p(t^{(1)})$. Plugging those back into Equation 23 yields

$$\begin{aligned} p(\theta|t^{(2)},t^{(1)}) &= p(t^{(2)}|\theta,t^{(1)})p(\theta|t^{(1)})/p(t^{(2)}|t^{(1)}) \\ &= p(t^{(2)}|\theta)\frac{p(t^{(1)}|\theta)p(\theta)}{p(t^{(1)})}\Big/p(t^{(2)}) \\ &= \frac{p(t^{(2)}|\theta)}{p(t^{(2)})}\frac{p(t^{(1)}|\theta)}{p(t^{(1)})}p(\theta) \\ &= \frac{p(t^{(1)}|\theta)}{p(t^{(1)})}\frac{p(t^{(2)}|\theta)}{p(t^{(2)})}p(\theta) \\ &= p(\theta|t^{(1)},t^{(2)}) && (24) \end{aligned}$$

where the last equality comes from simply going backwards through the previous equalities with the order of $t^{(1)}$ and $t^{(2)}$ reversed.

Appendix B
Highlighting is not Bayesian

Beyond its dependency on trial order, there is another way to illustrate the non-Bayesian nature of highlighting. A Bayesian analysis of the highlighting situation suggests that what the learner should do, when tested with cue I by itself or cues PE.PL, is respond with the overall base rates of the outcomes. In the canonical design, $p(E) = p(L)$, and so the Bayesian analysis predicts no preference for E over L, unlike human responding. A formal derivation for this fact follows next.

By Bayes' theorem, the probability of outcome E given cue I is

$$\begin{aligned} p(E|I) &= p(I|E)p(E)/p(I) \\ &= p(I|E)p(E)/[p(I|E)p(E)+p(I|L)p(L)] \\ &= p(E)/[p(E)+p(L)] \\ &= p(E) && (25) \end{aligned}$$

Thus, as claimed, a Bayesian responder would not prefer E over L in the canonical design.

For an analysis of the Bayesian response when cues PE.PL are presented, we proceed as follows. First, by Bayes' theorem,

$$\begin{aligned} &p(E|PE.PL) \\ &\quad= p(PE.PL|E)p(E)/p(PE.PL) \\ &\quad= \frac{p(PE.PL|E)p(E)}{p(PE.PL|E)p(E)+p(PE.PL|L)p(L)} && (26) \end{aligned}$$

Because the combination PE.PL is never seen in training, we must make assumptions about its probability of occurrence. One reasonable assumption is that $p(PE.PL|E) = p(PE.PL|L) = \delta > 0$, where δ is a small, perhaps infinitesimal, value. Alternatively, we could assume that cues occur independently for each outcome, that is, $p(PE.PL|E) = p(PE|E)p(PL|E)$ and $p(PE.PL|L) = p(PE|L)p(PL|L)$, and that $p(PL|E) = p(PE|L) = \delta > 0$. Under either assumption, Equation 26 becomes

$$\begin{aligned} & p(E|PE.PL) \\ = \quad & \frac{p(PE.PL|E)p(E)}{p(PE.PL|E)p(E) + p(PE.PL|L)p(L)} \\ = \quad & p(E)/[p(E)+p(L)] \\ = \quad & p(E) \end{aligned} \qquad (27)$$

Again, as claimed, a Bayesian responder would not prefer E over L in the canonical design.

This analysis suggests that no time-independent Bayesian model could account for highlighting. The derivation of Equation 27 did require some extra assumptions, however, that a defender of the Bayesian approach might discover reasons to reject. The derivation of Equation 25 required no such extra assumptions, and highlighting in the canonical training procedure is inherently a trial-order effect.