Cognitive Science 32 (2008) 1349 –1375

ISSN: 0364-0213 print / 1551-6709 online
DOI: 10.1080/03640210802463724

Parameters, Predictions, and Evidence in Computational Modeling: A Statistical View Informed by ACT–R

Rhiannon Weaver

Department of Statistics, Carnegie Mellon University

Received 30 August 2007; received in revised form 3 September 2008; accepted 8 September 2008

Abstract

Model validation in computational cognitive psychology often relies on methods drawn from the testing of theories in experimental physics. However, applications of these methods to computational models in typical cognitive experiments can hide multiple, plausible sources of variation arising from human participants and from stochastic cognitive theories, encouraging a "model fixed, data variable" paradigm that makes it difficult to interpret model predictions and to account for individual differences. This article proposes a likelihood-based, "data fixed, model variable" paradigm in which models are treated as stochastic processes in experiments with participant-to-participant variation that can be applied to a broad range of mechanistic cognitive architectures. This article discusses the implementation and implications of this view in model validation, with a concrete focus on a simple class of ACT-R models of cognition. This article is not intended as a recipe for broad application of these preliminary, proof-of-concept methods, but as a framework for communication between statisticians searching for interesting problems in the cognitive modeling sphere, and cognitive modelers interested in generalizing from deterministic to stochastic model validation, in the face of random variation in human experimental data.

Keywords: Model comparison; ACT–R; Likelihood-based methods; Bayesian statistics; Computational cognitive modeling; Cognitive architectures

1. Introduction

In computational psychology, current methods and critiques for inference and model validation draw from methods developed for the physical sciences, but in practice these methods can be restrictive in their interpretations for experimental settings subject to multiple sources of variation. Expressing computational cognitive models as stochastic models, by specifying the probability relation between observed data, unobservable parameters, and theoretical assumptions, provides a framework that can uncover these sources of variation and guide

Correspondence should be sent to Rhiannon Weaver, CIC 3109, Carnegie Mellon University, 4500 5th Avenue, Pittsburgh, PA 15213-2612. E-mail: rlweaver@stat.cmu.edu

methodology for addressing them. In this article, I present such a stochastic view of the Adaptive Control of Thought–Rational (ACT-R) cognitive architecture (Anderson & Lebiere, 1998) as a first step in a statistical standpoint on parameter estimation, predictions, and cognitive model validation using evidence from human experimental data.

In much of the computational psychological modeling community, evaluation methodology involves comparison of summary measures calculated from a human population with the same measures calculated from simulated data produced by a model. The goal is to provide useful simulations that also validate the underlying theory. Many computational models also have a set of parameters that may be tuned in order to give a model some flexibility in matching human data. Comparisons between model and data are evaluated based on goodness of fit; Schunn and Wallach (2001) survey common methods.

Reliance on goodness-of-fit metrics has been criticized in the past. Roberts and Pashler (2000) noted that fit metrics alone do not take into account the flexibility or complexity of a model. For example, critics of ACT, a predecessor to ACT-R, noted that it was such a flexible theory as to allow for the modeling of the same set of results with two models that made contradictory assumptions (Wexler, 1978). Similar criticism has also been leveled at other complex models in psychology and social science (e.g., MacCallum, Wegener, Uchino, & Fabrigar, 1993). In ACT-R modeling, the concept of *zero-parameter fits*—setting parameter values *a priori* to plausible values—arose as an attempt to combat overly flexible models by restricting the range of a model's predictions.

In a broader context, the deterministic view of a computational model maps a fixed set of inputs (parameters) to a fixed set of outputs (predictions), so restricting the set of plausible inputs constrains the resulting set of observed outputs. However, this viewpoint also restricts model interpretability and characterization of a model's predictions. Although summary statistics calculated from model simulations are treated as measuring the fixed constants that comprise model predictions, in reality these summaries are averages of multivariate, correlated measures across multiple simulations of individual actions. And fixing parameters at a single value restricts interpretaion of those parameters to universal constants, generalizeable to any population or individual. This makes it difficult to discuss multiple, plausible sources of variation in both models and human data.

It is clear that a stochastic computational model can produce different sets of observable behavior from one simulation run to another, even when parameter values are fixed. However, it is also plausible to attribute variation in human data to factors other than probabilistic mechanisms, for example, variation in parameter values:

- From participant to participant.
- From task to task regardless of participant.
- From task to task within-subjects.
- Fithin task and within-subjects.

These scenarios are gaining more attention in the ACT–R community with models applied to interpret individual traces of behavior in cognitive assessment and diagnosis (for example, Koedinger & MacLaren, 2002; Pavlik & Anderson, 2004), and in more traditional contexts (for example, Daily, Lovett, & Reder, 2001) with models studying individual differences

that cannot be addressed with zero-parameter fits. Relaxing the deterministic constraints on models and their predictions invites broader questions as well. Can a computational model make strong stochastic predictions when structured parameter variation is introduced into the model? What kind of human experimental data is needed to validate these predictions?

As a cognitive theory progresses, estimation and validation methods must also adapt in order to measure and control multiple sources of variation in observed data. Criticism of current methods is not new, and rigorous statistical methods have started to appear in the literature (e.g., Pitt, Myung, & Zhang, 2002). I believe a comprehensive approach requires a formal description of models as stochastic processes by which an individual participant generates individual data. In statistical terms, the mathematical link between the traces of action observed in a human experiment, and the model theorized to have produced them, is called the *likelihood function* (e.g., see Casella & Berger, 2001, chap. 6).

In this article, I outline the basic framework for applying likelihood-based analysis to a constrained class of ACT-R models. Section2 describes the current methodology for computational model evaluation and comparison in the context of experimental physics, and shows how these methods can obscure strong relationships in the data and the theory in some typical ACT-R experiments. Section3 defines the set of ACT-R models considered in this preliminary research and develops the mathematical form of the likelihood function for those models. Section 4 presents proof-of-concept examples of likelihood-based Bayesian estimation and model selection, concluding with a comment on scalability and a suggestion for experimental design in complex settings under one set of assumptions for individual differences.

I do not intend for this article to provide a recipe for broad application of likelihood-based techniques in the modeling community. Writing ACT-R models as stochastic processes shows how the statistical theory applies for these cognitive models, and provides a guide for developing the likelihood function for other architectures such as EPIC (Kieras & Meyer, 1997) or Soar (Laird, Newell, & Rosenblum, 1987) that allow for both stochastic mechanisms and parametric flexibility in model building. However, implementing likelihood-based methods for complex models is a difficult task that often requires adaptations on a model-by-model basis to obtain good results. Nevertheless, I hope this article can lay some groundwork for communication between statisticians searching for interesting applications in the cognitive modeling sphere, and cognitive modelers searching for model validation methods that accommodate multiple sources of variation.

2. The legacy of the measurement model

Statisticians view models that aim to explain how a system "works" as *substantive* (equivalently, *mechanistic*) models (Cox, 1990; Lehmann, 1990). Computational models developed with cognitive architectures are examples of these. For empirical forecasting, a "black box" model is as good as its predictions alone. However, for substantive models, the processes behind those predictions attempt to provide an interpretable and accurate, if abstracted, reflection of the way observed data are generated. These mechanisms posit a scientific theory that gathers evidence from data. In the face of conflicting evidence, the mechanisms should be adapted or discarded.

When observed data are used to learn about parameters or to validate model structure, statisticians ask, "How much information do the data contain about the model?" This is a subtly different question than, "How many data sets does this model fit?," which often arises in discussions of the strength of scientific theories (e.g., Cutting, 2000). In the first case, the data are considered fixed, and the model unknown or variable, and in the second, the model is fixed, and the data are variable. It also reflects two different methodological views of variability, in terms of *measurement* versus *random variation*. In this section, I outline the major differences between these views, and I re-examine current experimental methodology in light of its genesis in measurement and its applicability to random variation.

2.1. Measurement and random variation

Data collection methodology for computational psychological modeling has historically drawn from theory testing methods in the physical sciences, which rely heavily on a *measurement model* of variability. Bevington and Robinson (1992) described the model in detail; I outline the relevant features in the following.

In an experiment, a quantity of interest (mass, pressure, volume, etc.) cannot be measured exactly, subject to the precision of instrumentation. To increase precision, multiple measurements are taken and averaged together to yield a *data point*. Because experimental conditions can be tightly controlled, the main source of variability among these measurements can be attributed to random, independent, and unbiased error from the apparatus. Thus the data point, though subject to a margin of error, is an unbiased estimate of the underlying fixed quantity. Measurement errors are often modeled as continuous, Gaussian fluctuations, independent across multiple quantities of interest.

In contrast to measurement error is the more general statistical concept of *random variation*. In this scenario, a large component of variability is due to factors other than measurement error; for example, fluctuations from participant to participant. This variability is often left uncontrolled in experiments, either due to practicalities or limits in generalizability or because individual differences are of express interest.

Random variation can arise due to heterogeneity in an experimental population, or to a non-deterministic theory that posits probabilistic relationships as generating mechanisms for observed data. Holland (1990) called these the "random sampling" and "stochastic subjects" rationales. The quantity of interest in a model of random variation is an entire distribution in the population, and the distributions of multiple quantities of interest are generally not independent. Furthermore, the sample average is harder to interpret as estimating a fixed quantity of interest; instead, it estimates the population mean, which is one of many possible summarizations of the target distribution.

When the measurement model fails, what is an interpretable data point? Suppose we have the task of measuring the trade-off between speed and accuracy (recorded as either a success or failure) for a cognitive task. In this case, for a single participant, we cannot obtain a measurement of speed independent from a measurement of accuracy. Instead, each repetition of the task produces a latency value coupled with either a success or failure. If we obtain one repetition for each of 50 participant, there are now two different sources of variation contributing to the calculated data points. The first results from aggregating accuracy and

latency across successes and failures, and the second results from aggregating these values across participants who *a priori* may be more or less susceptible to making errors as they increase speed. The contribution of these individual differences to the "error" in the data points makes it more difficult to use these summaries to test certain hypotheses; for instance, whether a logarithmic within-subjects relationship exists between the two quantities.

Although responses averaged across participants are considered as the basic data point through much of the computational modeling literature, it is more enlightening to consider them as *statistics*—summary measures calculated over the domain of individual responses. The utility of these statistics depends on the experimental setting and on the modeling goals.

2.2. ACT–R and experimental data

The ACT–R architecture represents a participant's cognitive decision making, visual understanding, and motor action as a result of repetitions of a probabilistic cycle of procedural action and fact recall (see section 3.2 for details). Though ACT-R's structure and theoretical mechanisms differ from EPIC and Soar, the architectures are similar in that they include stochastic mechanisms in the theory; they traditionally account for two cognitive measures (i.e., performance time and accuracy),[1] and they simulate behavior on an individual level, producing traces that can be interpreted as a participant's decisions and actions leading to completion of a task.

Despite this, in current practice, data is often aggregated across participants and summarized assuming the measurement model of variability before being used to validate a model. An example is the serial digit recall experiment described by Anderson, Bothell, Lebiere, and Matessa (1998), where 62 participants were asked to memorize and recall a string of M grouped digits. Fig. 1 shows an example of the ACT–R memory structure for $M = 7$ with a $3 + 4$ grouping. Let D_{im} be the digit recalled by participant i on list item m (D_{im} could be "nil" if the participant recalled no digit) with associated latency L_{im}, and define the function $c(D_{im})$ as

$$c(D_{im}) = \begin{cases} 1 & D_{im} \text{ is correct} \\ 0 & \text{otherwise.} \end{cases}$$

An experiment yields a total of $14 (= 2M)$ data points. For each position m in the string, a pair $(\overline{c(D)}_m, \overline{L}_m)$ is recorded, where $\overline{c(D)}_m$ is the sample average, $(1/62)\sum_{i=1}^{62} c(D_{im})$, and $\overline{L}_m$ is the sample average, $(1/62)\sum_{i=1}^{62} L_{im}$. These data points are compared with the equivalent

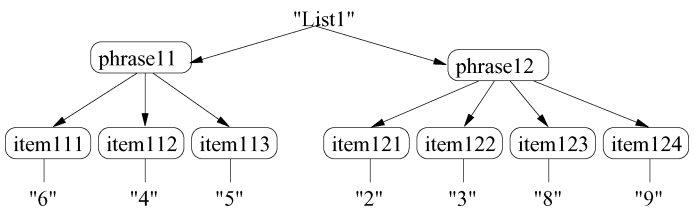

Fig. 1. Representation of serial list memory objects in a digit recall experiment, as modeled by Anderson and Matessa (1997).

data points $(\overline{c(D)}^*_m, \overline{L}^*_m)$ calculated from ACT–R model simulations. Model validation is achieved to the extent that the ACT–R model simulates patterns in $(\overline{c(D)}^*_m, \overline{L}^*_m)$ consistent with $(\overline{c(D)}_m, \overline{L}_m)$.

This data summarization method arises throughout the ACT–R community (e.g., see Baker, Corbett, & Koedinger, 2001; Ball, Gluck, Krusmark, & Rodgers, 2003; Daily et al., 2001; Fleetwood & Byrne, 2001; and others) and in the computational psychological community as a whole. For example, in their discussion of good fits and theory testing in psychology, Roberts and Pashler (2000) noted that data is often represented as a graph with two functions—"observed" and "simulated" (p. 358). The authors also noted that in many cases "the variability of the data (e.g., between-subject variation) is unclear," and that "adding error bars may not solve this problem; it is variability on the constrained dimension or dimensions that matters" (p. 359).

The measurement model influences both the methods and the critical discussion. Data points are calculated assuming independent errors, with variability conceptualized as flexibility of these data points in matching deterministic model predictions. Roberts and Pashler (2000) noted the inadequacy of this view, but they restricted discussions of the dimensions constrained by the model to the usual space of summary statistics of aggregated human data, as opposed to probability distributions of individual actions.

For example, to determine the scope of a model, Roberts and Pashler (2000) suggested measuring the model's predictions associated with each plausible value in the flexible parameter space. Although it is a systematic approach to account for between-subject parametric variation, the definition of a prediction is still imprecise. Suppose the prediction is a pair of arithmetic means, representing average accuracy and average latency for the hypothetical speed and accuracy task in section 2.1. In this prediction space, the proposed measure of scope cannot distinguish between parameter settings that lead to a 50% failure rate that takes an average of 20 sec of processing time regardless of outcome, and parameter settings that lead to a 50% failure rate that takes an average of 10 sec for successes and 30 sec for failures. This problem does not arise if the model's prediction is defined as the joint distribution of latency and response, associated with a single realization of the task for an individual participant.

Variability of simulated data is often discussed in the context of Monte Carlo error in estimating a deterministic model prediction. Baker, Corbett, and Koedinger (2003) explained how repeated simulations reduce the margin of error of the sample average, and thus pinpoint simulated data points very accurately. However, these sample averages still estimate numerical characterizations of probability distributions. Schunn and Wallach (2001) suggested extending the method to comparison of higher order moments of model and data distributions (e.g., standard deviations), but this does not address the problem fully because the comparison methods still assume independent measurement error among any of these quantities. Further, it raises the question of how many summary measures are needed to validate a model, without taking into account the assumptions of the underlying mechanisms to determine whether these measures are supportive of the research goals.

For example, according to the model structure in Fig. 1, groups of digits are related due to their parent phrase. Under ACT-R's theory of memory activation, the presence of a reference to the parent phrase spreads activation among its children in a fact recall (Anderson, 1983).

The result for the digit recall experiment affects positional confusions; a participant should be more likely to confuse items within the same group, rather than across groups. This is a prediction of the ACT–R model that can provide evidence supporting the hierarchical memory structure of Fig. 1 (e.g., as opposed to a simple linear linked list of seven items), but it cannot be observed from the summary statistic $\overline{c(D)}_m$.

In fact, standard metrics for comparing human data with model simulations can indicate very strong relationships even when it is impossible for the model to account for some individual responses. As an example, Baker et al. (2003) built several ACT–R models to explain two kinds of conceptual errors students make when asked to construct a scatterplot. For data visualization, scatterplots are used to graphically display two quantitative variables. Students were given a task with two quantitative variables and one categorical variable as a distractor, and were directed to produce a scatterplot. In a "variable choice" error, students chose the categorical distractor along with one quantitative variable and produced a bar chart. In a "nominalization" error, students chose both quantitative variables, but treated the numerical values for one axis as categorical labels, and used the remaining quantitative variable to produce the equivalent of a bar chart. Table 1 summarizes their results in the form of observed percentages of each of these errors under several scaffolding conditions.

One model's simulated percentages for each type of error are shown in parentheses. The model, called KNOW-IT-ALL, assumed that students could use either knowledge of scatterplots as visualization tools for quantitative data or knowledge of properties of quantitative variables to perform the task. If a student did not remember the purpose of scatterplots as quantitative visualization tools, they could still use their knowledge of quantitative variables to avoid a nominalization error. Despite high correlation, the percentages in the final row of the table correspond to structural zeros in the ACT–R model; it could not simulate this error, despite observation of the error in four of the five scaffolding conditions.

The structural impossibilities hidden in these data points raise questions about the strength of evidence they contain for model validation and parameter estimation. If a model is unable

Table 1
Percentages of different behaviors in scatterplot construction as reported by Baker, Corbett, and Koedinger (2003)

Accuracy of Responses in Scatterplot Construction					
	Scaffolding Condition				
Proportion of Observations	1	2	3	4	5
Variable choice error	.15 (.03)	.27 (.14)	.08 (.17)	.27 (.17)	.07 (.15)
Correct axis variables (CAV)	0 (.05)	.73 (.77)	.80 (.81)	.73 (.74)	.77 (.83)
% of CAV observations					
with correct representation	na	.74 (.81)	.74 (.80)	.84 (.80)	.79 (.82)
with x axis nominalized only	na	.16 (.19)	.17 (.20)	.16 (.20)	.13 (.18)
with y axis nominalized only	na	0 (0)	0 (0)	0(0)	0(0)
with both axes nominalized	na	.05 (0)	.09 (0)	0 (0)	.08 (0)

Note. ACT-R predictions (model KNOW-IT-ALL) are shown in parentheses beside each observation. $r = 0.983$.

to reproduce an individual's responses, it is difficult to determine how those responses provide information about the most likely values of the model's tuneable parameters related to the data generation process. And when the model can reproduce an individual's responses, it should also be able to provide a distribution of likely unobservable actions that could have generated those responses.

Focus on the measurement model has encouraged a "model fixed, data variable" paradigm in computational psychology that makes it difficult to attribute individual actions to the underlying generating process, to measure individual differences, and to estimate likely values of flexible parameters. Though ACT-R is the focus of this article, the critique applies more broadly; human diversity may be fertile ground for some very flexible cognitive theories, but criticism is at least partially driven by data summarization methods that cannot accommodate multiple sources of variation, and thus make it more difficult to validate the strong assumptions of a stochastic theory. In the next sections, I use a constrained class of ACT-R models to motivate development of substantive computational models as stochastic processes, and I show how a likelihood-based model addresses these concerns.

3. A stochastic model of ACT–R decision-making processes

The ACT–R architecture has both a symbolic layer and a sub-symbolic layer. Fig. 2 shows the symbolic layer, comprising four types of objects:

- Memory object: This is either a *chunk*, several related categorical attributes representing a fact; or a *production* representing a procedural action and a set of conditions under

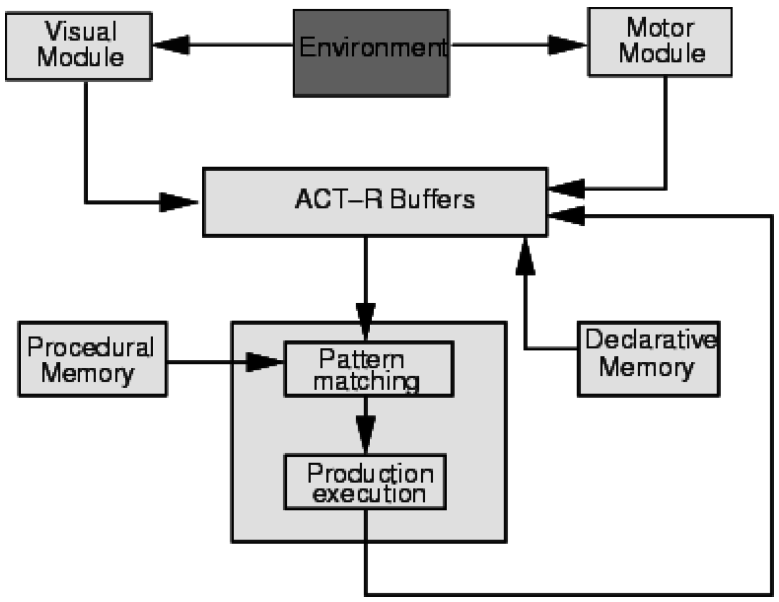

Fig. 2. ACT–R symbolic architecture. Courtesy of the ACT–R Web site at http://act-r.psy.cmu.edu/about

which the action can be executed. A production that executes its actions is said to *fire*.

- Module: This is an encapsulated system component devoted to performing specific actions and processing specific kinds of information. For instance, the *declarative memory module* holds a collection of chunks that can be referenced for aid in solving cognitive tasks. The *procedural memory module* holds a collection of productions that can fire.
- Buffer: This is an interface to a module that can hold one chunk. I define the contents of all buffers in the system as the model's *running state*. Productions modify the running state by modifying the attributes of chunks that reside in buffers or by initiating actions that cause chunks to be moved in or out of buffers. A production that requests a chunk to be placed into the declarative memory buffer is said to initiate a *retrieval*.
- Pattern matcher: This is a device that evaluates conditional statements against an object as true or false. ACT–R has a pattern matcher that evaluates production conditions against the running state to determine which productions can fire, and one that evaluates retrieval requests against a set of chunks to determine which chunks can be retrieved.

The sub-symbolic layer is a set of mechanisms governing three basic ways that symbolic objects interact:

- Probabilistic selection: These are mechanisms describing how the model selects among a set of competing productions to fire (conflict resolution), or among a set of competing chunks to be retrieved (fact selection).
- Latency: These are mechanisms determining the amount of time that it takes a procedural action, fact recall, and motor or audio–visual response to complete execution.
- Learning: These are mechanisms describing how memory objects merge, and how parameter values change, with time and experience.

For this preliminary research, I impose the following restrictions on the symbolic and sub-symbolic architecture:

1. Symbolic structure is restricted to a set of pre-specified objects in the procedural and declarative memory modules. Productions can access two buffers: a buffer holding one pre-specified chunk designated GOAL and the declarative memory buffer, which can hold one chunk designated RETRIEVAL.
2. The sub-symbolic architecture includes conflict resolution for productions, fact selection for retrievals, and latency rules for those actions. In particular, rules for learning are not used.[2]
3. The model is *synchronous*; it satisfies any outstanding retrieval requests before initiating the next production conflict resolution.

As a motivating example, consider a very simple cognitive task that has two outcomes, success or failure, in an experiment with I independent participants. Suppose Model A represents this task as a purely procedural task, as follows:

- Declare chunk ℓ_1^A with attribute STATE.
- At the start of the experiment:

  - Set STATE = "failure" in chunk ℓ_1^A.
  - Set GOAL = ℓ_1^A.
- Declare production P_1^A with the rule:
  - IF STATE = "failure" in the GOAL chunk:
  - THEN set STATE = "success."

Compare this to Model B, which uses both procedural action and fact recall:

- Declare chunk ℓ_1^B with attributes STATE and STEP.
- Declare chunk ℓ_2^B with attribute VALUE.
- At the start of the experiment:
  - Set STATE = "failure" in chunk ℓ_1^B.
  - Set STEP = "first" in chunk ℓ_1^B.
  - Set VALUE = "success" in chunk ℓ_2^B.
  - Set GOAL = ℓ_1^B.
- Declare production P_1^B with the rule:
  - IF STATE = "failure" and STEP = "first" in the GOAL chunk:
  - THEN retrieve chunk ℓ_2^B into the RETRIEVAL buffer, and set STEP = "second."
- Declare production P_2^B with the rule:
  - If STATE = "failure" and STEP = "second" in the GOAL chunk, and RETRIEVAL = ℓ_2^B:
  - THEN set STATE = VALUE, where VALUE is the attribute of the RETRIEVAL chunk.

At the end of the experiment, we observe the overall latency, L_i, and accuracy D_i for each participant, defined in model terms as follows:

$$D_i = \begin{cases} 1 & \text{if GOAL STATE = "success"} \\ 0 & \text{otherwise.} \end{cases}$$

These models admit very limited actions. In Model A, either P_1^A fires, producing a success; or P_1^A does not fire, producing a failure (section 3.2 explains how a production action or chunk retrieval can fail). For Model B, a success occurs when P_1^B fires, ℓ_2^B is retrieved, and P_2^B fires. However, the process can fail in an attempt of any of these actions, yielding three ways to generate $D_i = 0$. Models A and B are the simplest of ACT–R "models," but their actions are the building blocks of the stochastic process that occurs in sequence for all ACT–R models under constraints 1 through 3.

A stochastic process is defined mathematically by the specification of two elements:

- The *state space* that describes the range of values the process can take at any step k.
- The *transition probability distribution* that outlines the probability mechanisms for changes of state.

In section 3.1, I use the symbolic structure of ACT-R models satisfying 1 through 3 to derive the state space for the resulting stochastic process; and in section 3.2, I use the sub-symbolic structure to derive its transition probability distribution.

3.1. Symbolic structure and the state space

For ACT-R models satisfying 1 through 3, the contents of the GOAL and RETRIEVAL buffers define the running state, and the actions that generate transitions between stable running states can be described by a simple process I call a *decision cycle*. In this cycle, a single production is chosen to fire. This production may or may not request retrieval of a chunk from declarative memory into a buffer. If no productions match the running state, or if no matching production succeeds in firing, the model terminates. Similarly, if no chunks match a production retrieval request, or no chunk is successfully retrieved from a request, the model terminates.

The running state is a subset of the *symbolic state* of the model. In addition to the running state, the symbolic state also includes the following:

- All productions in procedural memory.
- All chunks in declarative memory.
- The production that fired and the chunk (if any) that was retrieved in the last completed decision cycle.

A model simulation also has a *temporal state* associated with any symbolic state, represented by a collection of quantitative variables:

- The production overhead time associated with the last completed decision cycle.
- The chunk recall time (could be 0 if no request was made) associated with the last completed decision cycle.
- The timestamp associated with the transition to the current symbolic state.

The progression of symbolic states describes an individual's "state of mind" as he or she attempts a cognitive task, with latency arising from the associated temporal states. Let us collect this sequence into a vector $\boldsymbol{\eta} = (\eta_0, \ldots, \eta_K)$, where each η_k records the current stable symbolic and temporal state at step k of the decision-making process. In this section, I suppress the subject index i for ease of notation. From a particular state η_k, a decision cycle is initiated and completes its actions. The result is a new state η_{k+1}, arising from the fired production's symbolic changes in chunk attributes or buffer contents and the timing cost of those actions.

Fig. 3 displays this model graphically. The box at the left encompasses the hidden progression of states η_k. Although these states and transitions are unobserved, we observe partial byproducts of them: a sequence of choices, $D_1, \ldots, D_m$, and possibly latency times between choices, $L_1, \ldots, L_m$. Each observation (D_m, L_m) has the form of a timestamp–action pair. We may also observe cues or other conditions set by the experimenter (represented by the box labeled "Cue").

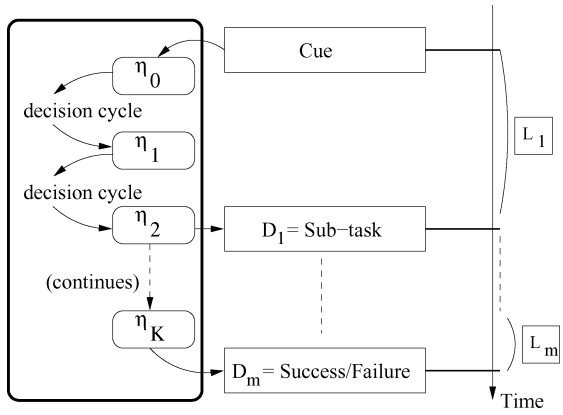

Fig. 3. A general cognitive task modeled by ACT–R. Observable data (square boxes) can be either categorical response data or latency associated with responses. Observable data is generated by the underlying, unobservable transition process (round boxes), where state-to-state transitions occur with decision cycles.

Let the progression $\eta_1, \ldots, \eta_k$ that generates (D_1, L_1) through (D_m, L_m) be the participant's *path*. When all states do not admit output, call the sequence of states between outputs a *path segment*.

From this depiction we see that ACT–R models fall into a class of discrete time, dynamic system models in which observed data is produced serially by an unobserved stochastic process. A general example is shown in Fig. 4, with square boxes representing observed quantities, round boxes representing unobserved quantities, and arrows indicating conditional dependence. The dependence structure in both Fig. 3 and Fig. 4 also indicate that the unobserved process is a Markov process; η_k depends only on η_{k-1}.[3] The Markov property is easy to achieve when there is flexibility in defining the state variables; if a system can be shown to have higher order dependencies among η_k and $\eta_1, \ldots, \eta_{k-2}$, the state variables collected in η can be adapted or expanded to retain dependence only on η_{k-1}.

Different canonical modeling paradigms arise from different assumptions on the format of η and y and on the relations between η_k, η_{k+1}, and y_k. For example, a *linear dynamical system* (e.g., Roweis & Gharamani, 1999) assumes both η_k and y_k are multivariate normal variables related by linear transformations. For the ACT–R model, however, the symbolic element of η_k is a collection of categorical variables that change discrete values over time, and

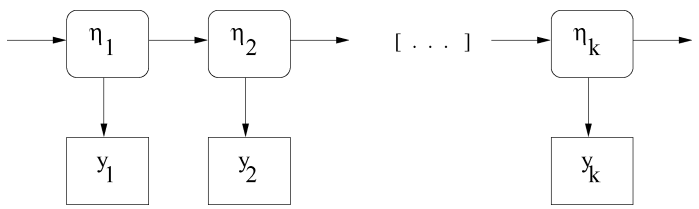

Fig. 4. Generic model of a dynamical system with unobserved data η generated via a Markov process, and observed data y generated at each step conditional on η.

the continuous temporal elements have no restrictions to Gaussian distributions. Whereas y_k is observable for every state in Fig. 4, in Fig. 3 only some states are linked to observed data, which may be an action or a timestamp–action pair.

Suppose η_k admits the mth timestamp–action pair. Because the temporal state contains full timing information, the latency L_m is a deterministic function of η_k. Let us further assume that the action D_m is a subset of the symbolic elements of η_k (e.g., a choice encoded as a chunk attribute). This ensures that (D_m, L_m) is a deterministic function of η_k, and it relegates the stochastic element of an ACT-R model to the transitions from η_{k-1} to η_k.

3.2. Sub-symbolic structure and transition probabilities

The sub-symbolic mechanisms for the decision cycle mediate transitions of the multidimensional states η_k. For ACT–R models satisfying 1 through 3, these mechanisms govern conflict resolution among productions and fact selection among chunks (transitions of symbolic state), and elapsed time for production action and chunk retrieval (transitions of temporal state).

Suppose a model has $\mathbb{J}$ productions and $\mathbb{L}$ chunks. If more than one production matches the model's running state, the system chooses production j to fire based on its *utility*:

$$U_j = \rho_j G - C_j,$$

where G, C_j, and ρ_j are sub-symbolic parameters. When more than one chunk matches a production retrieval request, the system chooses chunk ℓ to be recalled based on its *activation*:

$$A_\ell \approx \beta_\ell - d \log t_\ell - \sum_n PM_{n\ell},$$

where β_ℓ is a baseline, d is a decay rate, t_ℓ is the length of time since chunk ℓ was last recalled into a buffer, and the term $\sum_n PM_{n\ell}$ is a partial matching penalty for chunk ℓ's attributes against the retrieval request.

Probabilistic selection (conflict resolution for productions and fact selection for chunks) is performed by adding logistic noise to the utility or activation values of competing objects and selecting the competitor with the highest resulting value. Sub-symbolic parameters s_u (utility) and s_a (activation) control the level of noise added in this probabilistic step. The action is successful only if the selected value is higher than a *threshold*, denoted τ_u for utility and τ_a for activation.

For notational purposes, we can view thresholds as parameters associated with "virtual" symbolic elements; τ_u is the utility of a production matching any running state that halts decision cycles when it fires, and τ_a is the activation of a virtual chunk that matching any retrieval request that halts decision cycles when it is recalled. In later sections, I label these virtual elements production 0 and chunk 0.

For transitions of temporal state, the overhead time, T^o, accrued when production j fires, is

$$T^o = C_j.$$

The cost C_j defaults to 50 msec, the "atom of cognition" for ACT–R, but it can be adjusted if j admits other actions not explicitly modeled; for example, motor responses (e.g., van Rijn, Someren, & van der Maas, 2000).

The retrieval time, T^r, accrued when chunk ℓ to be recalled into a buffer is

$$T^r = Fe^{-(A_\ell + \nu_\ell)},$$

where F is a sub-symbolic parameter, and $A_\ell + \nu_\ell$ is the noisy activation value for chunk ℓ ($\nu_\ell \sim \text{logistic}(0, s_a)$).

In the stochastic ACT-R process, these mechanisms define a transition probability distribution, $P(\eta_{k+1}|\eta_k)$, over the symbolic and temporal state space. But these mechanisms are also functions of the sub-symbolic parameters, some of which are fixed *a priori*, whereas others are treated as tuning parameters (e.g., s_u, s_a, τ_u, or τ_a). Let us collect these parameters together into a vector, θ. To express the dependence mathematically, we write the transition probability as $P(\eta_{k+1}|\eta_k, \theta)$.

Elements of both θ and $\boldsymbol{\eta}$ can be unknown; I distinguish θ as elements that are not determined from the symbolic or temporal state. For example, partial matching penalties in A_ℓ are calculated directly from the symbolic state; similarly, t_ℓ is calculated from the temporal state (in fact, for models that use activation decay, we must expand the temporal state to include a "time since last recall" value for each chunk in declarative memory in order to preserve the Markov property). These are deterministic functions of $\boldsymbol{\eta}$. However, the decay rate d and baseline β_ℓ are not fixed by any η_k, and so are considered parameters in θ. In models where learning occurs, these parameters may be updated deterministically with state transitions; currently, I consider values in θ as static.

To derive $P(\eta_{k+1}|\eta_k, \theta)$, we first examine the probability distributions that arise from each of the sub-symbolic mechanisms (following Anderson & Lebiere, 1998, Appendix A, pp. 89–92). The distribution that describes the maximum value among competing logistic variables does not have a closed form. An expression for the approximate probability that production j will fire from a set M of competitors is

$$p_f(j|M, \eta, \theta) = \frac{e^{U_j/\sigma_u}}{e^{\tau_u/\sigma_u} + \sum_{m \in M} e^{U_m/\sigma_u}}, \tag{1}$$

where $\sigma_u = \sqrt{2}s_u$. Similarly, the approximate probability that chunk ℓ will be recalled by production j in a set N of matching chunks is

$$p_r(\ell|N, j, \eta, \theta) = \frac{e^{A_\ell/\sigma_a}}{e^{\tau_a/\sigma_a} + \sum_{n \in N} e^{A_n/\sigma_a}}, \tag{2}$$

where $\sigma_a = \sqrt{2}s_a$. When production j fires, ACT-R theory provides no source of variation in the overhead time, T^o, given fixed cost C_j. But it is plausible to observe minute variations in these completion times, even for one individual in repeated trials of a simple mastered task (imagine a participant repeatedly snapping his or her fingers). Thus, it seems reasonable to allow T^o some flexibility, conditional on C_j. In this research, I describe T^o as a scaled

Gamma($2000C_j$, 2) variable, with the following distribution function:

$$f_o(T^o = t|j, \theta) = \left(\frac{1}{1000}\right) \frac{1}{2^{2000C_j}\Gamma(2000C_j)} (1000t)^{2000C_j - 1} \exp[-(1000t)/2] \quad (3)$$

For $C_j = 0.05$, T^o has a tight bell-shaped distribution with mean 0.05. Also, as a connection with the underlying neural processes that the production abstracts, the Gamma distribution can be interpreted as the summed completion time for a long series of very short tasks. For notational ease, I associate the completion time T^o for the "virtual" production 0 (threshold failure) with a "cost" parameter C_0 in Equation 3; in modeling, C_0 can be set equal to τ_u.

The recall time of chunk ℓ retrieved by production j from a set N of competitors is approximated using a Weibull variable with shape parameter $\sqrt{2}/\sigma_a$ and scale parameter F/e^{μ}, where $\mu = \sum_{i \in N} A_{\ell_i}$. The distribution function is

$$f_r(T^r = t|\ell, N, \eta, \theta) = \frac{F\sigma_a}{\exp(\mu)\sqrt{2}} \left(\frac{t\sigma_a}{\sqrt{2}}\right)^{\frac{F}{\exp(\mu)} - 1} \exp\left[-\frac{t^{\frac{F}{\exp(\mu)}}\sigma_a}{\sqrt{2}}\right]. \quad (4)$$

Each decision cycle corresponds to the product of up to four probability distributions, associated with production selection and overhead (Equations 1 and 3), and if a chunk is requested, a retrieval and recall time (Equations 2 and 4). Let Z_k be an indicator variable, equal to 1 if step k initiates a retrieval, and 0 otherwise. Then the transition probability $P(\eta_{k+1}|\eta_k, \theta)$ has the following general form:

$$\begin{aligned} &P(\eta_{k+1}|\eta_k, \theta) \\ &\quad = p_f(j_k|M_k, \eta_k, \theta) f_o(T_k^o|j_k, \theta)[p_r(\ell_k|N_{j_k}, j_k, \eta_k, \theta) f_r(T_k^r|\ell_k, N_{j_k}, \eta_k, \theta)]^{Z_k}. \end{aligned} \quad (5)$$

3.3. Likelihood for ACT-R models

Let y_{ik} represent the observation for each state η_{ik}– no data, D_{im}, or (D_{im}, L_{im}) for the m th observed action. Let $\mathbf{y} = (y_{11}, \ldots, y_{1K_1}, \ldots, y_{I1}, \ldots, y_{IK_I})$, for I independent participants in an experiment. Similarly, define $\boldsymbol{\eta}$ as the collection of all path data for these participants. The likelihood function, $\mathcal{L}(\mathbf{y}; \boldsymbol{\eta}, \theta)$, is the joint probability distribution of $\mathbf{y}$, viewed as a function of the unobserved states and parameters. For any modeling paradigm described by Fig. 4, it is easy to express this joint probability for the i th subject:

$$\begin{aligned} \mathcal{L}(y_i; \eta_i, \theta) &= P(\eta_{i1}, \ldots, \eta_{iK}, y_{i1}, \ldots, y_{iK_i}|\theta) \\ &= P(\eta_{i1}|\theta) \prod_{k=1}^{K_i - 1} P(\eta_{i,k+1}|\eta_{ik}, \theta) \prod_{k=1}^{K_i} P(y_{ik}|\eta_{ik}, \theta). \end{aligned} \quad (6)$$

The likelihood of all data is the product of each individual sequence:

$$\mathcal{L}(\mathbf{y}; \boldsymbol{\eta}, \theta) = \prod_{i=1}^{I} P(\eta_{i1}) \prod_{k=1}^{K_i - 1} P(\eta_{i,k+1}|\eta_{ik}, \theta) \prod_{k=1}^{K_i} P(y_{ik}|\eta_{ik}, \theta), \quad i = 1, \ldots, I.$$

As we have defined the ACT-R process, η_{ik} either admits y_{ik} or not, so the probability terms $P(y_{ik}|\eta_{ik}, \theta)$ reduce to

$$P(y_{ik}|\eta_{ik}, \theta) = \begin{cases} 1 & \text{if } \eta_{ik} \text{ admits } y_{ik} \\ 0 & \text{otherwise} \end{cases}.$$

This implies that when $\mathcal{L}(\mathbf{y}; \boldsymbol{\eta}, \theta)$ is non-zero, it is equal to the product of ACT-R transition probabilities (Equation 5) for each path for each participant. This model for observed timestamp–action pairs accounts for different decisions among participants by including the expression for each individual's path. Furthermore, at the sub-symbolic layer, any element of θ can easily accommodate individual differences. For example, τ_u can be expanded to $(\tau_{u1}, \ldots, \tau_{uI})$, such that each τ_{ui} enters into the model only in the expression for participant i's path.

Consider Model A defined in section 3 which has the single production P_1^A that fires to produce a success. Using P_0^A to represent the virtual utility threshold production (which fires to produce a failure), an expression for the likelihood is

$$\begin{aligned}\mathcal{L}(\mathbf{y}; \boldsymbol{\eta}, \theta) = \prod_{i=1}^{I} &\left[p_f\left(P_1^A|\{P_0^A, P_1^A\}, \theta\right) f_o\left(L_i|P_1^A, \theta\right)\right]^{D_i} \\ &\times \left[p_f\left(P_0^A|\{P_0^A, P_1^A\}, \theta\right) f_o\left(L_i|P_0^A, \theta\right)\right]^{1-D_i}.\end{aligned}$$

For each participant, a success ($D_i = 1$) contributes the first term of the equation to the likelihood, whereas a failure ($D_i = 0$) contributes the second.

It is impractical to summarize all possible paths in one expression for the more complicated structure of Model B. Model B has two productions, P_1^B and P_2^B, in addition to the utility threshold P_0^B, as well as a retrieval chunk, ℓ_2^B, and retrieval threshold, ℓ_0^B, with a structure of four possible paths—one leading to success and three leading to failure (at the failure of any of the 3 symbolic actions that compete with thresholds to succeed). But the likelihood of any one path is again the product of transition probabilities. For example, for a single participant i with ($D_i = 1, L_i = l$), where l is calculated from temporal state elements $(T^o, T^r)_{i1}$ and $(T^o, 0)_{i2}$, we have

$$\begin{aligned}&\mathcal{L}(D_i = 1, L_i = l; (T^o, T^r)_{i1}, (T^o, 0)_{i2}, \theta) \\ &\quad = p_f\left(P_1^B|\{P_0^B, P_1^B\}, \theta\right) f_o\left(T_1^o|P_1^B, \theta\right) \\ &\quad \times p_r\left(\ell_2^B|\{\ell_0^B, \ell_2^B\}, P_1^B, \theta\right) f_r\left(T_1^r|\ell_2^B, \{\ell_0^B, \ell_2^B\}, \theta\right) \\ &\quad \times p_f\left(P_2^B|\{P_0^B, P_2^B\}, \theta\right) f_o\left(T_2^o|P_2^B, \theta\right).\end{aligned}$$

Note that $\mathcal{L}(\mathbf{y}; \boldsymbol{\eta}, \theta)$ is written in terms of individual timestamp–action pairs (D_{im}, L_{im}) and their relationship to paths and sub-symbolic parameters. The mathematical structure of $\mathcal{L}(\mathbf{y}; \boldsymbol{\eta}, \theta)$ dictates which data aggregations can be made to (D_{im}, L_{im}) in order to use all information from the likelihood in learning about the model's structure. In a "data fixed, model variable" scenario, information is the extent that the observed quantities constrain the set of likely unobserved quantities. The most precise bounds on unobserved quantities are attained when all individual observed data is retained. Certain summary statistics can also

provide bounds as precise as the individual data (these are called *sufficient statistics*; e.g., Casella & Berger, 2001, chap. 6), but the form of these statistics depends on the mathematical structure of the likelihood function. Without foreknowledge of this structure, it is difficult to find summaries of the data that do *not* relax these constraints. Derivation of sufficient statistics for ACT-R is beyond the scope of this article. But in general, the safest way to avoid information loss in an experiment is to retain all individual data.

4. Applications of the likelihood

Bayesian applications of the likelihood are attractive for ACT-R because they can incorporate expert opinion and strong prior information for θ to help with model interpretation while allowing for individual differences. Bayesian methods also fit naturally into the "data fixed, model variable" view of model evaluation using human data.

Section 4.1 introduces parameter estimation in the form of posterior inference—exploring the distribution of likely values for unobserved components of θ and $\boldsymbol{\eta}$, given evidence in $\mathbf{y}$. For model comparison, section 4.2 introduces the Bayes factor (Kass & Raftery, 1995), which is the *a posteriori* evidence of the model structure given evidence in $\mathbf{y}$. Section 4.3 provides a simple example, and section 4.4 gives a comment on scalability and experimental design.

4.1. Posterior inference

Model validation may require using $\mathbf{y}$ to learn about $\boldsymbol{\eta}$, θ, or both. Traditionally, ACT-R modeling has focused on estimating θ; however, as focus shifts to individual traces of action in the stochastic ACT-R process, we can look to $\boldsymbol{\eta}$ to provide an interpretable summary of likely cognitive action at the production level for each individual.

Let $\pi(\theta)$ be the prior information about sub-symbolic parameters θ, expressed as a probability distribution. For example, production cost C_j may default to a tight distribution around 50 msec to guide the production's interpretation as the "atom of cognition," whereas a value such as τ_u may have a diffuse distribution reflecting prior uncertainty. The quantity of interest for Bayesian analysis is the joint posterior distribution of the unobserved values given the following observations:

$$p(\theta, \boldsymbol{\eta}|\mathbf{y}) = \frac{\mathcal{L}(\mathbf{y}; \boldsymbol{\eta}, \theta)\pi(\theta)}{m(\mathbf{y})}. \tag{7}$$

The numerator is the product of $\pi(\theta)$ and the likelihood function. The denominator is the marginal probability of $\mathbf{y}$ given model structure and prior assumptions:

$$m(\mathbf{y}) = \int \left[\int \mathcal{L}(\mathbf{y}; \boldsymbol{\eta}, \theta) d\boldsymbol{\eta} \right] \pi(\theta) d\theta. \tag{8}$$

The posterior distribution expresses uncertainty in terms of the model structure and parameters as opposed to the space of new data sets, yielding a model-space viewpoint for diagnostics and substantive validation. The distribution of likely $\boldsymbol{\eta}$ given $\mathbf{y}$ is a direct link between observations and the data generating process. Also, in addition to point estimates of θ (e.g., using posterior

means), the *a posteriori* joint correlation structure of any number of flexible parameters can be directly explored.

Often, the integral $m(\mathbf{y})$ cannot be calculated directly, and so Equation 7 does not have a closed form. For example, the expression for ACT-R models satisfying the simplifying conditions 1 through 3 is as follows:

$$m(\mathbf{y}) = \int \prod_{i=1}^{I} \left\{ \sum_{\boldsymbol{\eta} \in \mathcal{S}_{D_i}} \left[\prod_{m=1}^{M_i} \int_{\boldsymbol{\eta}_m \in \mathcal{T}_{L_{im}}} \left\{ \prod_{k=1}^{K_m} P(\eta_{mk} | \eta_{m,k-1}, \theta) \right\} d\boldsymbol{\eta}_m \right] \right\} \pi(\theta) d\theta.$$

Here, $\mathcal{S}_{D_i}$ is the set of all paths generating symbolic action D_{i1}, ..., D_{im}, and $\mathcal{T}_{L_{im}}$ is the space of all temporal state variables that sum to L_{im} in path segment m. For each participant i, we must first find $\mathcal{S}_{D_i}$ (a difficult task in its own right). Then, for each path segment $\boldsymbol{\eta}_{im}$ of $\boldsymbol{\eta} \in \mathcal{S}_{D_i}$, we integrate the product of transition probabilities over the constrained temporal state space $\mathcal{T}_{L_{im}}$, and sum the resulting values from all path segments for all $\boldsymbol{\eta} \in \mathcal{S}_{D_i}$. Taking the product of this calculation over participants yields the inner integral from Equation 8, which then must be integrated with respect to the flexible sub-symbolic parameters in θ and their prior distributions. This calculation does not have a closed form, even for simple ACT-R models.

In the event that $m(\mathbf{y})$ cannot be calculated analytically, the posterior distribution can still be explored by obtaining a sample $(\theta, \boldsymbol{\eta})_1, \ldots, (\theta, \boldsymbol{\eta})_B$ from $p(\theta, \boldsymbol{\eta}|\mathbf{y})$ using Markov chain Monte Carlo (MCMC) methods (e.g., Gilks, Richardson, & Spiegelhalter, 1996). MCMC methods bypass the need for calculating $m(\mathbf{y})$ by starting with an initial value $(\theta, \boldsymbol{\eta})_0$ and obtaining $(\theta, \boldsymbol{\eta})_b$ for $b = 1, ..., B$ by iterating between the following stochastic steps in a Markov chain:

1. Data augmentation: For each participant i, sample a new path $(\boldsymbol{\eta}_i)_b$ from the conditional distribution $p(\boldsymbol{\eta}_i|\theta_{b-1}, \mathbf{y}_i)$.
2. Parameter updating: Sample a new sub-symbolic parameter vector θ_b from the conditional distribution $p(\theta|(\boldsymbol{\eta})_b, \mathbf{y})$.

Statistical theory (e.g., Tanner & Wong, 1987) proves that this iterative algorithm will eventually converge so that successive $(\theta, \boldsymbol{\eta})_b$ are drawn from $p(\theta, \boldsymbol{\eta}|\mathbf{y})$. These values can be used to estimate any number of posterior measures; for example, the marginal probability $m(\mathbf{y})$ that could not be calculated directly. As we see in the next section, this probability is a critical component in the calculation of Bayes factors for model comparison.

MCMC was recognized as a useful way to estimate both model parameters θ and unobservable subject-specific quantities (of which paths $\boldsymbol{\eta}_i$ are an example) by Patz and Junker (1999). Although standard MCMC techniques (see Gilks et al. (1996) chap. 5) can be applied directly to the parameter updating step for general ACT-R models, the data augmentation step is difficult to implement due to the high-dimensional state space that includes both categorical and continuous quantities and to the complex dependencies among (D_{im}, L_{im}) and the underlying path segment that generated it.

The data augmentation step requires sampling a path from $\mathcal{S}_{D_i}$ for each participant i, with weight proportional to the product of its transition probabilities given θ_{b-1} *and* $\mathbf{y}_i$. Accuracy data in $\mathbf{y}_i$ merely restricts the symbolic elements in $\mathcal{S}_{D_i}$; transition probabilities given $\mathbf{y}_i$ are proportional to Equation 5 when accuracy alone is observed. However, latency observations

Table 2
Parameter values for Model *A* and Model *B* in a simulation study with 100 participants for each model

Model *A*		Model *B*	
σ_u	1.00	σ_u	1.00
τ_u	0.00	τ_u	0.00
ρ_1	0.85	σ_a	0.20
C_1	0.05	τ_a	1.00
G	1.00	C_1	0.05
C_0	0.05	ρ_1	0.85
		C_0	0.05
		ρ_2	0.85
		C_2	0.05
		β_2	1.00
		G	1.00
		F	1.00

also penalize longer path segments m when L_{im} is small, and vice versa. Mathematically, this is a re-weighting of Equation 5 within path segments, proportional to the restrictions of $\mathcal{T}_{L_{im}}$. This re-weighting adds another layer of difficulty to the data augmentation step, beyond the task of efficiently exploring the symbolic states in $\mathcal{S}_{D_i}$.

A general solution does not yet exist for implementing the data augmentation step in ACT-R; currently, I have approached the problem on a model-by-model basis. For example, consider the model KNOW-IT-ALL developed for the scatterplot construction task in Table 1. This model was validated using accuracy data only; it had only one starting state, η_0, and all data was recorded after the model terminated. This allowed for a depth-first search of all possible model states, yielding 2, 772 distinct states, with paths consisting of progressions of approximately 30 states. Paths were sampled by matching the observed data for each individual i to a set $\mathcal{E}_{D_i}$ of states that admitted i's responses upon termination. At each iteration, a termination state η_{iK_i} was sampled from $\mathcal{E}_{D_i}$. A candidate path $\boldsymbol{\eta}_i^c$ was generated from $\mathcal{S}_{D_i}$ using a *proposal distribution*, $q(\boldsymbol{\eta}|\theta_{b-1})$, equivalent to stochastically propagating decisions backward through states η_{iK_i-1}, $\eta_{iK_i-2}\ldots$ and so on, until reaching η_0. To sample the candidate path with weight proportional to its transition probabilities given θ_{b-1}, $\boldsymbol{\eta}_i^c$ was accepted as $(\boldsymbol{\eta}_i)_b$ with probability equal to r, where

$$r = \min\left(1, \frac{\mathcal{L}(\mathbf{y}_i; \boldsymbol{\eta}_i^c, \theta_{b-1}) q((\boldsymbol{\eta}_i)_{b-1} \mid \theta_{b-1})}{\mathcal{L}(\mathbf{y}_i; (\boldsymbol{\eta}_i)_{b-1}, \theta_{b-1}) q(\boldsymbol{\eta}_i^c \mid \theta_{b-1}).}\right)$$

If $\boldsymbol{\eta}_i^c$ was rejected, then $(\boldsymbol{\eta}_i)_b$ was set to $(\boldsymbol{\eta}_i)_{b-1}$. This is an example of a *Metropolis–Hastings* sampling algorithm (see Gilks et al., (1996) chap. 1.3.3.), tailored to the structure of model KNOW-IT-ALL. Using the acceptance ratio r can be shown to sample new paths with the correct weights as the MCMC algorithm converges.

4.2. Model comparison and selection

Model selection can be used to test the merits of competing theories or to adapt and refine an existing model. Both of these goals involve summarizing or exploring complex model structures, with difficulties arising in measuring complexity. The number of free parameters in θ does not accurately reflect the dimension of model flexibility when these parameters are subject to dependencies and informative priors (e.g., Hodges & Sargent, 2001), and the path structure $\boldsymbol{\eta}$ is similar to a random effect in a diagnostic assessment model, as the number of paths grows with the number of individuals (e.g., Patz, Junker, Johnson, & Mariano, 2002).

A criterion that is well-suited to these complications for overall model comparison is the Bayes factor (Kass & Raftery, 1995). Let A and B be two competing models, equally likely *a priori*, with marginal likelihood $m(\mathbf{y}|A)$ and $m(\mathbf{y}|B)$ given $\mathbf{y}$. The Bayes factor $\mathcal{B}_{AB}$ is defined as

$$\mathcal{B}_{AB} = \frac{m(\mathbf{y}|A)}{m(\mathbf{y}|B)}.$$

A value $\mathcal{B}_{AB} > 10$ indicates strong preference for Model A.

The marginal likelihood measures model adaptability without computing an effective number of parameters, as is necessary with comparison criteria such as minimum description length (Barron, Rissanen, & Yu, 1998) or Bayesian Information Criterion (Schwarz, 1978). It provides a "data fixed, model variable" view of Roberts and Pashler's (2000) notion of strong support for a theory; the value of $m(\mathbf{y})$ is high when the posterior distribution of θ is similar to the prior distribution $\pi(\theta)$ in areas where $\pi(\theta)$ contains strong information. Penalties are incurred with large likelihood on a space of low prior weight (an uninterpretable model) or with loosely constrained, flexible posterior distributions that can match any observation (an unfalsifiable model). As noted earlier, $m(\mathbf{y})$ can be estimated using MCMC samples $(\theta, \boldsymbol{\eta})_1, \ldots, (\theta, \boldsymbol{\eta})_B$; for example, with the harmonic mean estimator (Kass & Raftery, 1995, section 4.3):

$$\hat{m}(\mathbf{y}) = \left\{ \frac{1}{B} \sum_{b=1}^{B} \mathcal{L}(\mathbf{y}; (\theta, \eta)_b)^{-1} \right\}^{-1}. \tag{9}$$

4.3. A simple example

To illustrate posterior inference and model selection in a simple setting, simulations of $I = 100$ participants were produced for each of Models A and B from section 3 using the sub-symbolic parameter values described in Table 2. Denote these data $\mathbf{y}^A$ generated from Model A; and $\mathbf{y}^B$ generated from Model B. The observations are summarized in Fig. 5.

Success rates were 0.74 for $\mathbf{y}^A$ and 0.27 for $\mathbf{y}^B$ Average latency across all participants and outcomes was 0.0508 sec for $\mathbf{y}^A$ and 0.3501 sec for $\mathbf{y}^B$ Latency distributions are shown in Fig. 5 with *density plots*—smoothed histograms that have been scaled on the y axis so the area under the curve is equal to 1. Like probability distributions, peaks on a density plot represent modes in the observed data. The distributions for $\mathbf{y}^A$ reflect Model A's simple Gamma assumption

Table 3
Results of model selection performed on data y^A simulated using Model A and data y^B simulated using Model B.

Bayes factors (Log Scale) for Selecting Model A Over Model B		
$\log(\frac{m(\mathbf{y}\mid A)}{m(\mathbf{y}\mid B)})$	$y_i = D_i$	$y_i = (D_i, L_i)$
$\mathbf{y} = \mathbf{y}^A$	22.7	418.1
$\mathbf{y} = \mathbf{y}^B$	61.8	−15381.3

Note. Models were fit to data using either accuracy only (column 1) or accuracy and latency (column 2). Positive numbers indicate a preference for Model A, whereas negative numbers indicate a preference for Model B. A magnitude greater than 3 on the log scale indicates strong evidence for the chosen model.

for both successes and failures. The distribution of success times in $\mathbf{y}^B$ are generally longer due to the time needed for chunk retrieval, and the distribution of failure times is bimodal because failure can occur either before or after the chunk retrieval.

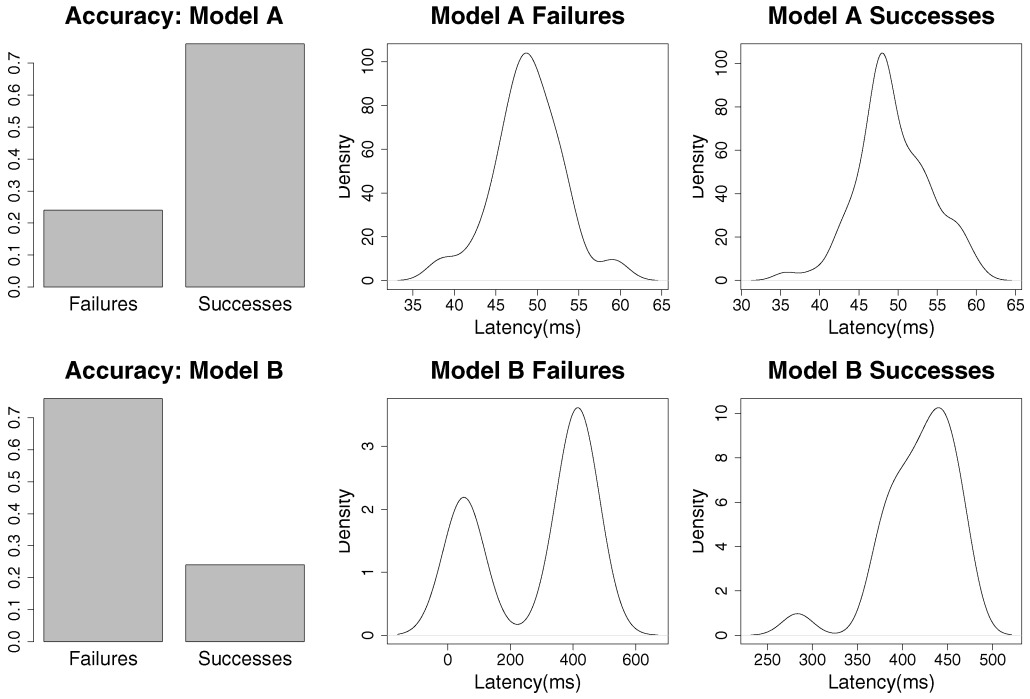

Fig. 5. Summary of simulated data for 100 participants for Models A and B with parameters set as listed in Table 2. Successes and failures are represented using bar charts. Latency values are represented using density plots (see text for definition). MS = milliseconds.

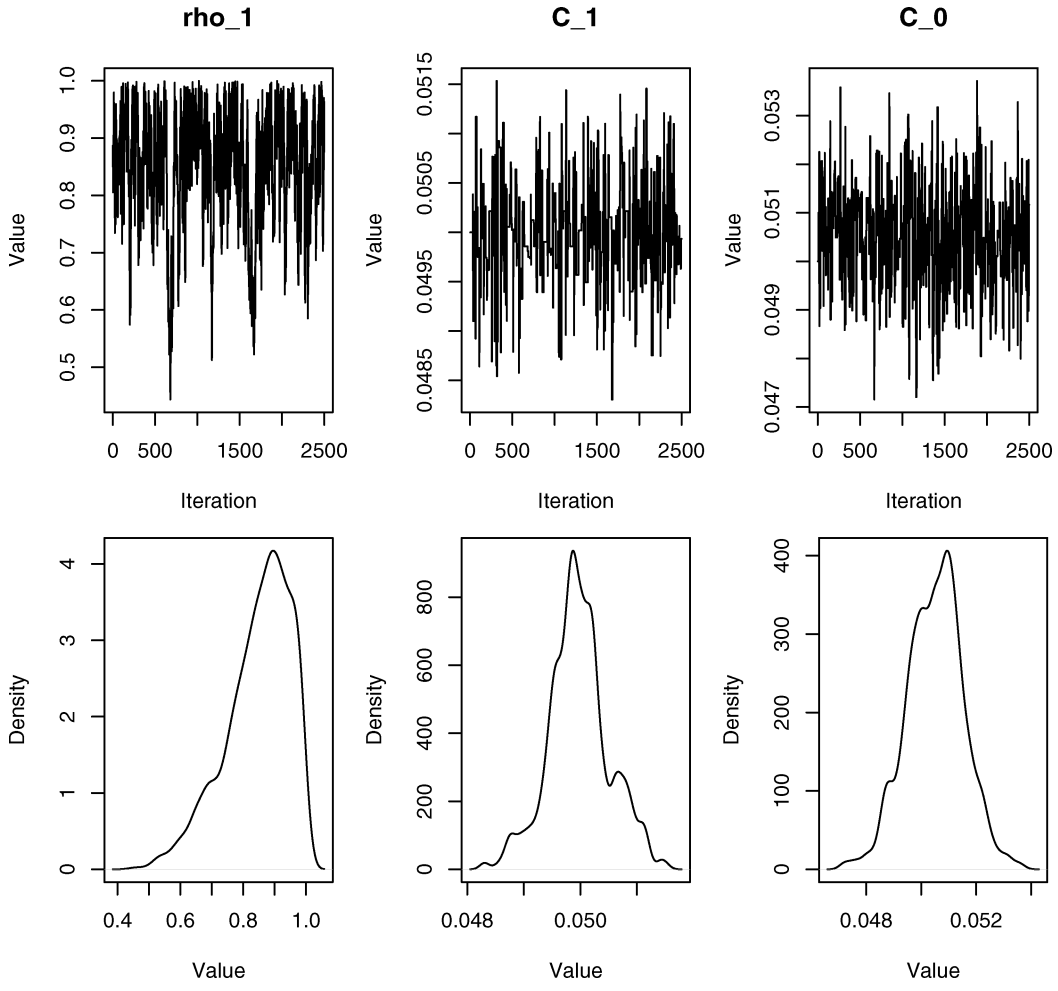

Fig. 6. Markov chain Monte Carlo iterations (top) and density plots (bottom) for posterior distributions (given $\mathbf{y}^A$) of three parameters allowed to vary in Model A.

I use Model A and $\mathbf{y}^A$ as the simplest example for posterior inference for θ. Because path information in A is completely determined by $\mathbf{y}$, the MCMC algorithm requires only the parameter updating step. As a first experiment, a uniform(0, 1) prior was set for ρ_1, and weak normal priors were set for C_1 and C_0, with all other sub-symbolic parameters fixed at the values in Table 2. Fig. 6 shows the results of 2,500 iterations of the parameter updating step. Time series plots of iterations (top row) indicate that the algorithm is stable, and density plots (bottom row) show that the posterior distributions are uni-modal and peaked near the values from Table 2.

In a second experiment, all six sub-symbolic parameters were allowed to vary, with diffuse prior information. Fig. 7 shows time series plots of 2,500 iterations of the parameter updating step. Although latency values strongly identify the cost parameters, the accuracy data in $\mathbf{y}^A$ does not contain enough information to distinguish τ_u, σ_u, G, and ρ_1 simultaneously.

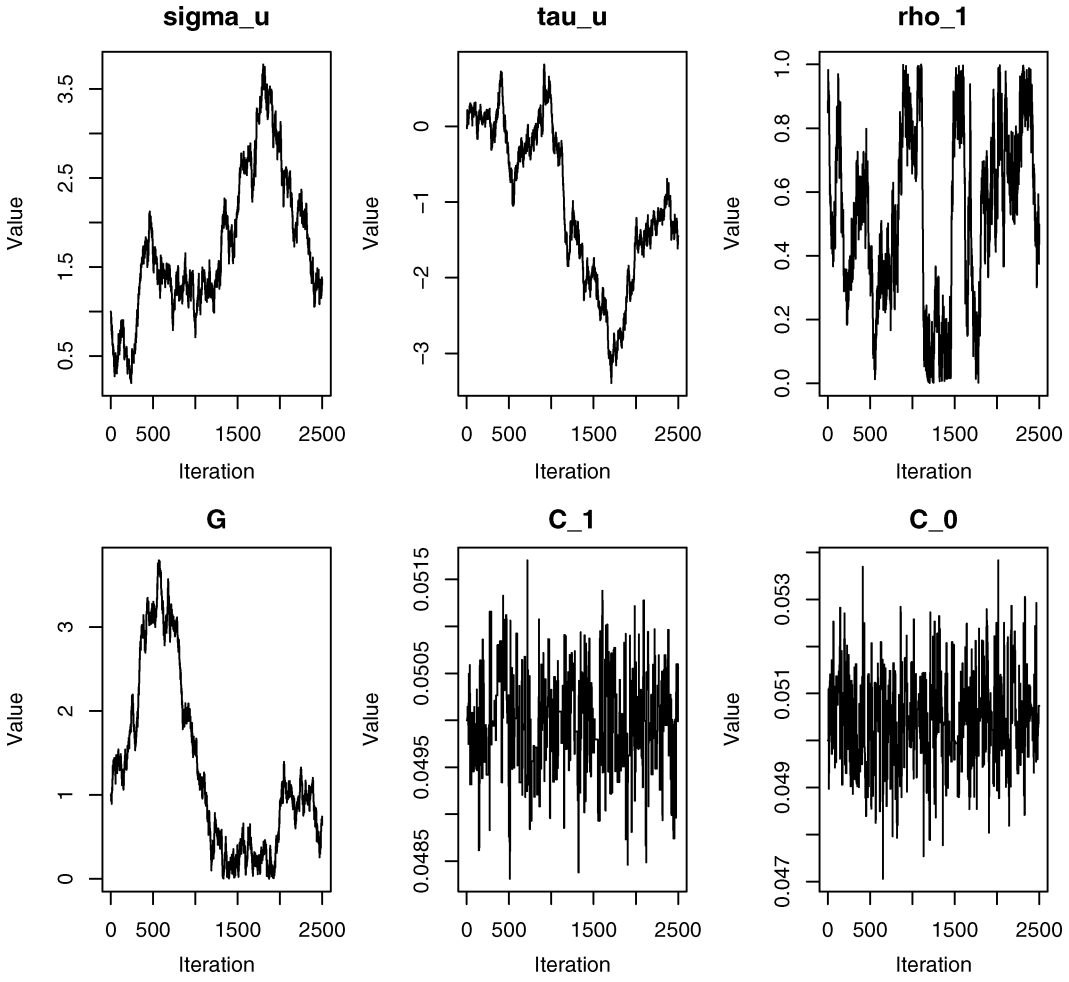

Fig. 7. Markov chain Monte Carlo iterations of posterior distributions (given $\mathbf{y}^A$) when all parameters in Model A are allowed to vary.

The determining factor for interpretability is instead the ability of $p_f(P_1^A|\{P_0^A, P_1^A\}, \theta)$ (Equation 1) to match the observed proportion of successes, 0.74. Despite the wild fluctuations in these four parameters from iteration to iteration, the posterior distribution of the probability of success, estimated by calculating $p_f(P_1^A|\{P_0^A, P_1^A\}, \theta_b)$ at each iteration b, is relatively constrained and peaked near 0.74 (Fig. 8).

For a Model selection experiment, models A and B were each fit to both $\mathbf{y}^A$ and $\mathbf{y}^B$ to determine which model each data set supported best. Two experiments were run for each model and data pair: one using only accuracy data and one using both accuracy and latency. For each experiment, the utility threshold τ_u was allowed to vary for Model A, whereas the utility noise σ_u, activation threshold τ_a, and baseline activation β_2 for the retrieval chunk were allowed to vary for Model B. This gave both models the flexibility to match any observed

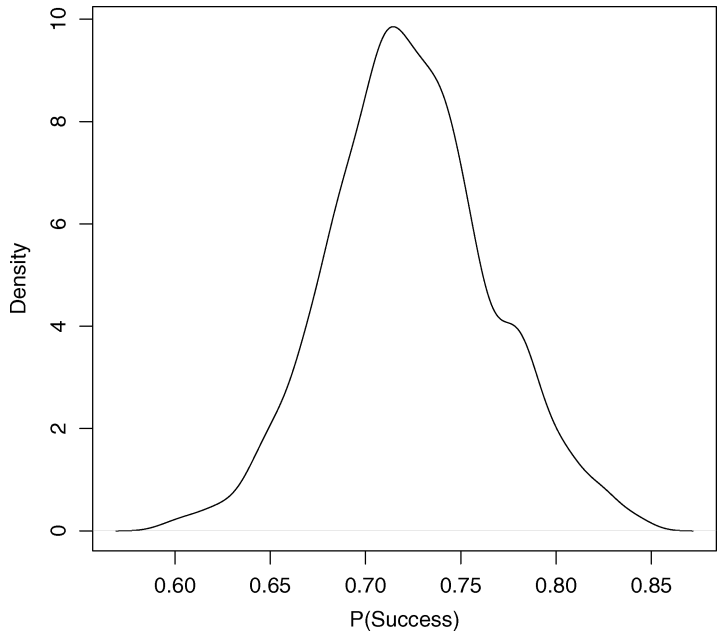

Fig. 8. Density plot for the posterior probability of a success, $P(D_i = 1)$, in Model A given $\mathbf{y}^A$ when all six parameters are free to vary. Despite lack of identifiability among σ_u, τ_u, G, and ρ_1, the posterior distribution for $P(D_i = 1)$ is stable and centered around the observed population proportion of 0.74.

proportion of successes in the data. When latency was used, the cost parameters C_0 and C_1 were allowed to vary in both models, as well as the overhead cost C_2 for Model B. Parameters that were not allowed to vary were fixed at the values in Table 2.

Diffuse priors were used for τ_u (in both models), β_2, σ_u, and τ_a to reflect prior uncertainty. Tight priors were used on C_0 and C_1 in both models, as well as C_2 in Model B, to reflect ACT-R assumptions. In Model B, it was necessary to devise a method for the data augmentation step in the MCMC algorithm. In this case, the sets $\mathcal{S}_{D_i}$ were easy to characterize, with only four possible paths of at most three state transitions. The simple model structure facilitated implementation of an otherwise inefficient Metropolis–Hastings sampling algorithm for re-weighting transition probabilities and sampling temporal variables conditional on $\mathcal{T}_{L_{i1}}$.

MCMC steps were run for 5,000 iterations in each experiment. Marginal likelihood was computed using the harmonic mean estimator (Equation 9) on the last 4,500 iterations, based on visual inspection of the iterations to indicate convergence. Table 3 shows the estimated Bayes factors (log scale) for each experiment.

Model A is favored for both $\mathbf{y}^A$ and $\mathbf{y}^B$ when only accuracy is used, but Model A is penalized severely on $\mathbf{y}^B$ when latency is included. With estimation of τ_u, Model A has enough flexibility to match the probability of success $p_f(P_1^A|\{P_0^A, P_1^A\}, \theta)$ with the observed value of 0.27 in $\mathbf{y}^B$, using fewer assumptions about the underlying process than Model B. Even when latency is recorded, Model A can exactly match the sample averages $(0.27, 0.3501)$ from $\mathbf{y}^B$; for example, by setting $C_0 = C_1 = 0.3501$ and $\tau_u = 1.4945$, and could still be selected as the more plausible generating process for $\mathbf{y}^B$ by parsimony. The Bayes factor penalizes Model A

so severely on $\mathbf{y}^B$ because no matter how much flexibility it is granted in θ, Model A cannot account for the bimodal distribution of failure latency observed in $\mathbf{y}^B$. This structure makes Model A too simple for $\mathbf{y}^B$, and so Model B is overwhelmingly preferred.

4.4. A note on scalability and experimental design

The scalability of Bayesian methods in ACT-R relies on the ability to perform the two steps of the MCMC algorithm efficiently. The number of states in complex ACT-R models can be very large; even when this number is manageable, sampling temporal information conditional on observed latency is still a challenge. Multiple modules, asynchronous actions, and evolving memory structures (e.g., production and chunk merging and chunk encoding) also add complexity to $\boldsymbol{\eta}$. Incorporating sub-symbolic learning mechanisms requires a likelihood structure where θ evolves over time, depending on feedback from $\mathbf{y}$. These details are approachable within the likelihood framework, but the question remains, What is practical in applications?

In lieu of a comprehensive answer, I will propose a scenario. Suppose we relax the zero-parameter fit constraint to apply only within-subjects: that is, we assume that each individual i has a sub-symbolic "profile," θ_i, which is transferrable among several cognitive tasks. Suppose we wish to account for this source of variation in a very complex model, C. One approach is to gather data $\mathbf{y}_i^s$ from each individual i in a simpler modeling task, S, and to perform the full-scale Bayesian inference to obtain posterior distributions $p(\theta_i|\mathbf{y}_i^s)$. Then, we perform the following simulation (repeated $b = 1, \ldots .B$ times):

- Sample θ_{ib} from $p(\theta_i|\mathbf{y}_i^s)$.
- Set sub-symbolic parameters in C to θ_{ib}, and run C to obtain $\boldsymbol{\eta}_{ib}^{*c}$.

This is a "data fixed, model variable" adaptation of current simulation methods that uses all individual data instead of summary statistics. The values $\boldsymbol{\eta}_{ib}^{*c}$, $b = 1, \ldots, B$ provide a predictive distribution of participant i's performance in the complex task, accounting for uncertainty about θ_i. When i performs the task, we can determine likely path structures for the observation $\mathbf{y}_i^c$ using latency and timestamp–action traces from the $\boldsymbol{\eta}_{ib}^{*c}$. An unlikely $\mathbf{y}_i^c$ for a single participant may indicate an individual who does not follow C's assumptions. But systematic errors in C's predictions across individuals may pinpoint structural flaws in C itself.

5. Conclusion

In statistics, the most famous quote about modeling comes from George Box (1979): "All models are wrong, some are useful." To those faced with the task of cognitive model validation and theory testing, this may seem like a pessimistic view of the world. But, perhaps appropriately, the statement itself is a generalized approximation of the truth, and we may still derive some use from it. In reality, models have many different purposes, and the statistical and philosophical viewpoints ask many of the same questions, if framed in different contexts.

I propose likelihood-based methods for evaluating and adapting ACT-R models on both substantive and predictive criteria. At the substantive level, the likelihood function makes

explicit the relationship between parameters, data generating processes, and observable quantities in an experiment, offering a detailed explanation of why a model performs as it does. At the predictive level, a computational cognitive model is an approximation of an individual; in the simulation setting, it is designed to think, act, and learn the way a human participant would. To evaluate such a model based on aggregated summary statistics introduces an unnecessary level of obfuscation that can not only hide large inaccuracies, but also mask subtle successes. In experimental settings with multiple sources of variation, the first step is to put the model of individual cognition on equal footing with the human participant, to learn how to use the gathered data to its fullest extent.

Notes

1. Functional magnetic resonance imaging facilitates measurement of some other observable experimental data, but most ACT–R models focus on time and accuracy.
2. Non-learning models can be interpreted as having reached expert status, evaluated using expert data (*terminal* models—see Salvucci and Anderson, 1998), or as models pinpointing a particular stage of the learning process.
3. Seen as a continuous dynamic system, ACT–R is considered a *semi-Markov* model, as it does not have exponential waiting times. I bypass the need for a continuous system by including the temporal state variables in η_k.

References

Anderson, J. R. (1983). A spreading activation theory of memory. *Journal of Verbal Learning and Verbal Behavior, 22*, 261–295.

Anderson, J. R., Bothell, D., Lebiere, C., & Matessa, M. (1998). An integrated theory of list memory. *Journal of Memory and Language, 38*, 341–380.

Anderson, J. R., & Lebiere, C. (1998). *The atomic components of thought*. Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Anderson, J. R., & Matessa, M. P. (1997). A production system theory of serial memory. *Psychological Review, 104*, 728–748.

Baker, R. S., Corbett, A. T., & Koedinger, K. R. (2001). Toward a model of learning data representations. In J. D. Moore & K. Stenning (Eds.), *Proceedings of the 23rd annual conference of the Cognitive Science Society* (pp. 45–50). Mahwah, NJ: Lawerence Erlbaum Association, Inc.

Baker, R. S., Corbett, A. T., & Koedinger, K. R. (2003). Statistical techniques for comparing ACT-R models of cognitive performance. In J. R. Anderson et al. (Eds.), *Proceedings of the 10th annual ACT-R workshop* (pp. 129–134).

Ball, J. T., Gluck, K. A., Krusmark, M. A., & Rodgers, S. M. (2003). Comparing three variants of a computational process model of basic aircraft maneuvering. In *Proceedings of the 12th conference on Behavior Representation in Modeling and Simulation* (pp. 87–89). Orlando, FL: Institute for Simulation and Training.

Barron, A., Rissanen, J., & Yu, B. (1998). The minimum description length principle in coding and modeling. *IEEE Transactions on Information Theory, 4*, 2743–2760.

Bevington, P., & Robinson, K. (1992). *Data reduction and error analysis for the physical sciences* (2nd ed.). New York: McGraw-Hill.

Box, G. (1979). Robustness in the strategy of scientific model building. In R. Launer & G. Wilkinson (Eds.), *Robustness in statistics* (pp. 201–236). New York: Academic.

Casella, G., & Berger, R. (2001). *Statistical inference* (2nd ed.). New York: Duxbury.
Cox, D. (1990). Role of models in statistical analysis. *Statistical Science, 4*, 169–174.
Cutting, J. (2000). Accuracy, scope and flexibility of models. *Journal of Mathematical Psychology, 4*, 3–19.
Daily, L. Z., Lovett, M. C., & Reder, L. M. (2001). Modeling individual differences in working memory performance: A source activation account in ACT-R. *Cognitive Science, 25*, 315–353.
Fleetwood, M. D., & Byrne, M. D. (2001). Modeling icon search in ACT-R/PM. In E. Altmann, A. Cleermans, C. Schunn, & W. Gray (Eds.), *Proceedings of the 4th international conference on Cognitive Modeling* (pp. 17–23). Mahwah, NJ: Lawrence Erlbaum Associates, Inc.
Gilks, W., Richardson, S., & Spiegelhalter, D. (1996). *Markov chain Monte Carlo in practice*. Boca Raton, FL: Chapman & Hall/CRC.
Hodges, D., & Sargent, J. (2001). Counting degrees of freedom in hierarchical and other richly parameterized models. *Biometrika, 88*, 367–379.
Holland, P. (1990). On the sampling theory foundations of item response theory models. *Psychometrika, 55*, 577–601.
Kass, R., & Raftery, A. (1995). Bayes factors. *Journal of the American Statistical Association, 90*, 773–795.
Kieras, D., & Meyer, D. (1997). An overview of the EPIC architecture for cognition and performance with application to human–computer interaction. *Human–Computer Interaction, 12*, 391–438.
MacLaren, B., & Koedinger, K. (2002). *When and why does mastery learning work: Instructional experiments with ACT-R "Sim Students."* 6th International Conference, ITS 2002, Biarritz, France and San Sebastian, Spain.
Laird, J., Newell, A., & Rosenblum, P. (1987). Soar: An architecture for general intelligence. *Artificial Intelligence, 33*, 1–64.
Lehmann, E. (1990). Model specification: The views of Fisher and Neyman, and later developments. *Statistical Science, 5*, 160–168.
MacCallum, R., Wegener, D., Uchino, B., & Fabrigar, L. (1993). The problem of equivalent models in applications of covariance structure analysis. *Psychological Bulletin, 144*, 185–199.
Patz, R., & Junker, B. (1999). A straightforward approach to Markov chain Monte Carlo methods for item response models. *Journal of Educational and Behavioral Statistics, 24*, 146–178.
Patz, R., Junker, B., Johnson, M., & Mariano, L. (2002). The hierarchical rater model for rated test items and its application to large-scale educational assessment data. *Journal of Educational and Behavioral Statistics, 27*, 341–384.
Pavlik, P., & Anderson, J. (2004). An ACT-R model of memory applied to finding the optimal schedule of practice. In M. Lovett, C. Schunn, C. Leviere, & P. Munro (Eds.), *Proceedings of the 6th international conference on Cognitive Modeling* (pp. 376–377). Mahwah, NJ: Erlbaum.
Pitt, M., Myung, I., & Zhang, S. (2002). Toward a method of selecting among competing models of cognition. *Psychological Review, 109*, 472–491.
Roberts, S., & Pashler, H. (2000). How persuasive is a good fit? A comment on theory testing. *Psychological Review, 107*, 358–367.
Roweis, S., & Gharamani, Z. (1999). A unifying review of linear Gaussian models. *Neural Computation, 11*, 305–345.
Salvucci, D., & Anderson, J. (1998). Analogy. In J. Anderson & C. Lebiere (Eds.), *The atomic components of thought* (pp. 343–384). Mahwah, NJ: Lawrence Erlbaum Associates, Inc.
Schunn, C., & Wallach, D. (2001). *Evaluating goodness-of-fit in comparison of models to data*. Retrieved March 31, 2004, from http://www.lrdc.pitt.edu/schunn/gof/index.html
Schwarz, G. (1978). Estimating the dimension of a model. *Annals of Statistics, 6*, 461–464.
Tanner, M., & Wong, W. (1987). The calculation of posterior distributions by data augmentation (with discussion). *Journal of the American Statistical Association, 82*, 528–550.
van Rijn, H., Someren, M., & van der Maas, H. (2000). Modeling developmental transitions in ACT-R. Simulating balance scale behavior by symbolic and subsymbolic learning. In N. Taatgen & J. Aasman (Eds.), *Proceedings of the 3rd international conference on Cognitive Modeling* (pp. 226–233). Veenendahl, The Netherlands: Universal Press.
Wexler, K. (1978). A review of John R. Anderson's "Language, Memory, and Thought." *Cognition, 6*, 327–351.