1

Model Evaluation and Selection

Jay I. Myung[1], Daniel R. Cavagnaro[2], and Mark A. Pitt[3]

July 22, 2014
(word count: 13,260)

To appear in William H. Batchelder, Hans Colonius, Ehtibar Dzhafarov and Jay I. Myung (eds.), *New Handbook of Mathematical Psychology, Vol. 1: Measurement and Methodology*. Cambridge, U.K.: Cambridge University Press

[1] Corresponding author, Department of Psychology, Ohio State University, Columbus, OH 43210 (Email: myung.1@osu.edu)
[2] Mihaylo College of Business and Economics, California State University at Fullerton, Fullerton, CA 92834 (Email: dcavagnaro@fullerton.edu)
[3] Department of Psychology, Ohio State University, Columbus, OH 43210 (Email: pitt.2@osu.edu)

Contents

1.1 Introduction

The study of cognition is challenged by the difficulty of inferring representation and processes in such a complex system as the brain. The field of cognitive science has met this challenge by borrowing and developing research tools with which to study the brain. *Tools* are meant broadly to include not just hardware (e.g., computers, eye trackers, imaging equipment) that are used for data collection, but also the quantitative tools used to guide inference, including statistical methods (Frequentist and Bayesian) and cognitive modeling.

Cognitive modeling assists in scientific inference by, among other things, assessing the plausibility of an explanation (e.g., theory, process). It achieves this by instantiating a version of the explanation in some quantitative form (i.e., the mathematical model), and thereby demonstrating its plausibility (Polk and Seifert, 2002; Busemeyer and Diederich, 2010; Lewandowsky and Farrell, 2011; Lee and Wagenmakers, 2014). But as with theories, all quantitative explanations are not equally good or convincing, so what criteria should be used to evaluate models? What makes a model a good explanation from which it is reasonable to draw inferences, and what signs indicate the model is poor? These questions are the focus of this chapter. Like modeling itself, the field is still very much in its infancy. Progress has been made, but many challenges remain. Before reviewing the state-of-the-art, we first provide a broader context in which to situate the enterprise of model evaluation.

Although cognitive modeling has been around since the 1950s, its popularity increased once computers became cheap and fast. Also, user-friendly software has accelerated its adoptions to the point where more and more researchers recognize the value of models and their usefulness for knowledge discovery (Shiffrin and Nobel, 1997; Fum et al., 2007; McClelland, 2009). Theories in much of the field tend to be broad claims about foundational issues in cognition (e.g., representations are distributed rather than local; grammar acquisition is probabilistic rather than rule-based; category learning is Bayesian). By instantiating these claims in a model, the theory becomes more viable, and as a consequence more persuasive, especially when its performance is shown to mimic that of individuals. In addition, it can be difficult to develop a theory with much depth without formalizing it quantitatively in some way. The reason for this stems from the challenge in understanding how the many parts of the theory (e.g., constructs, processes) operate and interact to lead to a specific outcome. The more complex the theory, the more difficult it is to make predictions with certainty (A version

of this problem haunts models as well, which we will discuss at the end of the chapter). Quantification of a theory provides a framework in which these interactions can be explored systematically so that they are understood, and then tested in future experiments.

For example, a theory of memory should include an explanation of forgetting. Initially it might posit only that memory decays as a function of the time between studying a list of words and being asked to recall them. Such a claim is specific enough to evaluate it experimentally, and might then lead to the further assertion that the relationship between recall and test delay is best described by a power function, with recall performance decreasing quickly over the first few time delays and then gradually leveling off over longer delays. At this point a simple model starts to emerge, and multiple roads can be taken. One is to demonstrate that other quantitative relationships (e.g., exponential, hyperbolic) between test delay and recall provide poorer descriptions of the data. A more ambitious goal, and one that is of more interest to the researcher, is to expand the model to specify the memory processes and how they operate to yield retention that decays according to a power function. Although the first path of exploration might constrain solutions to the second, it can be nontrivial to accomplish even the more modest goal of determining the shape of the retention function (e.g., Rubin and Wenzel, 1996; Navarro et al., 2004). The reasons for this are discussed later in the chapter.

The preceding example is intended to draw attention to a few aspects of cognitive modeling and their implications for model evaluation. One is that modeling requires the researcher to specify a mathematical formulation of the theory, however simple or tenuous it might be. This initial implementation then becomes a stand-in for the theory, serving as a surrogate that is evaluated on its own, with the theory lurking somewhere in the distance. A second implication is that no matter how thoughtful or careful a researcher is in developing an implementation, it is instructive to remember that it will always be wrong, exemplified by the famous quote,"All models are wrong, but some are useful" (G.E.P. Box, 1976). For a model to be even in the ballpark of possibilities, one would have to have a vast amount of very informative and precise (noise-free) data. Even then, the task of inferring the true form of the underlying model seems so daunting that one might wonder whether the approach is misguided. The tools in cognitive science are rarely able to provide the strong constraints needed to achieve such a lofty objective.

It is for these reasons that models are most productively viewed as tools for studying cognition. A quantitative framework serves to direct inquiry

toward particular issues, whether testing basic model assumptions, implementational choices, or predictions about variable interactions. If a model serves as a useful explanatory device and has advanced understanding in a field, then it has done its job, even if it is ultimately abandoned in favor of an alternative.

Again, how do we determine if a model is useful? An answers to this question requires careful evaluation of the model itself and can be nontrivial to obtain, but it behooves the researcher to scrutinize the design choices made in creating the model. What is needed for this purpose is yet another set of tools to evaluate the models themselves. More are being developed, but they lag behind modeling itself. Although many of the methods themselves were developed in the context of mathematical models, the fundamental issues in model evaluation are pertinent to all types of models.

Of course, the types of questions about model performance that one asks can depend on the modeling framework adopted, whether it be cognitive architectures, parallel distributed processing (PDP) systems, or Bayesian models. That so many styles of modeling flourish simultaneously in the cognitive sciences underscores the challenge of modeling all of cognition using a single framework. Instead, each style of modeling seems best suited for a particular content domain (McClelland, 2009).

In this chapter, we focus on the evaluation of and selection among mathematical models, which are defined as models that can be expressed in algebraic form and have a likelihood function. They include models of decision making, memory retention, psychophysical models, and some models of categorization. In each content area, models differ in the number of parameters (never more than five) and functional form (how the parameters and input are combined in the model equation). You might think that such simple models should be easy to choose among, but as we discuss in the following pages there are challenges at every turn. Because these models are similar to those developed in fields such as statistics and engineering, model evaluation methods developed in those domains have been imported into cognitive science.

We devote the majority of this chapter to methods that assess a model's suitability in accounting for data collected in an experiment. These methods are most well-known, and include measuring a model's fit to data as well as its flexibility. The final part of the chapter introduces computational methods that improve model selection by optimizing the design of the experiment used to discriminate the models. That is, knowlege of model behavior is used to identify experimental designs (e.g., choices of stimuli) that have the high-

est likelihood of discriminating among models. We begin with a discussion of the fundamental problems in model evaluation and selection.

1.2 Basic Ideas

In order to develop appropriate tools for evaluating models and selecting among them, we must first answer the questions of what makes a model good, and what makes one model better than another. A common misconception about modeling is that the goal is to fit the data as well as possible. This misconception probably stems from the fact that when people are first introduced to mathematical modeling, usually in the form of simple linear regression, the focus is on model fitting (i.e., finding the parameters of the model that best fit the data; see section 1.3). Model fitting (e.g., parameter estimation) is an important aspect of mathematical modeling, but modeling is about much more than just finding a set of best-fitting parameters. A model entails assumptions about the structure of data and the relationships between variables. For example, a simple linear regression model assumes a linear relationship between the independent and dependent variables, and a normal distribution of the dependent variable at each level of the independent variable. If our goal were to fit the data as well as possible, why limit ourselves to a linear relationship with just two parameters (slope and intercept) when a better fit could be obtained by assuming a more complex relationship such as a polynomial with three, for, or even five parameters? Why not fifty parameters?

John Von Neumann famously said, "With four parameters, I can fit an elephant, with five I can make him wiggle his trunk." By this he meant that one should not be impressed when a complex model fits a data set well. With enough parameters, you can fit any data set. A model with a lot of parameters is said to be complex because it can fit complex patterns of data. While data fitting is important, a complex model can fit for the wrong reasons, by fitting noise instead of regularities. Although we aim to precisely control the conditions of our experiments, real data are awash with idiosyncrasies due to individual differences, quirks, and nuance that cannot be controlled. These idiosyncrasies are commonly referred to as noise. All other things being equal, simpler models are more attractive because they are sufficiently constrained to make them easily falsifiable. The issue of model complexity will be addressed in section 1.4.3.

What is desired for model evaluation is a yardstick that measures a model's ability to capture the underlying regularity only, not idiosyncratic

noise. This requires a balance between goodness of fit and complexity, key concepts in model evaluation that are elaborated in the following sections.

1.3 Model Estimation

1.3.1 How Is a Model Specified?

As noted above, models are quantitative stand-ins for theories. A formal model expresses the theorized relationships between latent (not directly observable) processes (e.g., memory, attention) and observable responses using the language of mathematics. A mathematical mode is defined in terms of a mathematical equation that specifies the range of data patterns it predicts by varying the values of model parameters. As a concrete example, consider the following power model of memory retention (e.g., Wixted and Ebbesen, 1991):

$$\textit{Power model}: \quad p(a,b|t) = a(t+1)^{-b} \tag{1.1}$$

where $p(a,b|t)$ denotes the model's prediction of the probability of correct response at retention interval t, and a $(0 < a < 1)$ and b $(b > 0)$ are the model's two parameters. Shown on the left panel of Figure 1.1 are ten different power curves created by varying the value of parameter b with parameter a fixed to 1. The right panel of the same figure shows another ten power curses generated by varying the value of parameter a with parameter b set to 0.5. Note the diversity of memory decay patterns that the power model predicts for different choices of its parameters a and b.

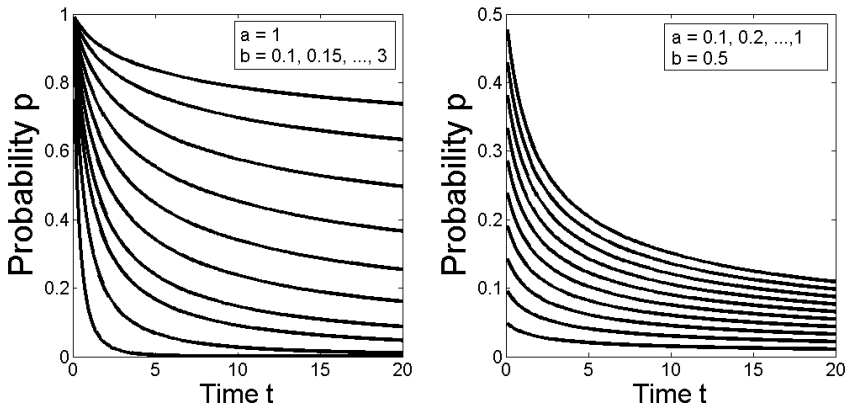

Figure 1.1 Sample power curves generated by varying values of model parameters (a, b) in Eq. (1.1).

Writing down the model equation such as in Eq. (1.1) is an important first step of model specification but is not its end. This is because the *deterministic* equation, as it is, represents an idealistic but unrealistic view of the mental and behavioral processes that the model is trying to capture. Experimental data are inevitably corrupted by random variability, whether it is due to the variability of experimental procedures or due to the imprecision of the measurement instrument. The same experimental stimulus does not necessarily invoke the same behavioral response from participants, often not even from the same participant. It is therefore important for a model to also specify how the random variability in the observed data is accounted for, in addition to the theorized regularity (memory decay rate) underlying the data.

To show how this is done, let us consider a retention memory experiment in which a participant first studies a list of n words in a study phase and then in a test phase is asked to recall the studied words. The time lag between the study and test phase defines the retention interval t. The power function in Eq. (1.1) specifies the probability of correct recall of a word at time t given a parameter vector (a, b). For example, for $t = 1$ and $(a = 0.9, b = 0.15)$, the power model predicts the probability to be equal to $p(a, b|t) = 0.9(1+1)^{-0.15} = 0.731$. Assuming that the participant recalls each word independently of the others, the number of correctly recalled words, denoted by y, out of the n words in the studied list would follow the binomial probability distribution:

$$f(y|n, (a, b)) = \frac{n!}{(n-y)!y!}\, p(a, b|t)^y (1 - p(a, b|t))^{n-y} \tag{1.2}$$

where $y = 0, 1, ..., n$.

Figure 1.2 depicts two binomial probability distributions corresponding to different choices of the parameter vector, $(a = 0.9, b = 0.15)$ (top panel) and $(a = 0.8, b = 1.3)$ (bottom panel), given $n = 10$ and $t = 1$. The top panel shows the distribution of the number of correct responses when the probability of correct recall of one word is equal to 0.731. While there is a relatively high probability of observing seven or eight correct responses out of ten, as one might expect, it is still possible to observe lower or higher numbers, and even much lower (e.g., 3) correct responses. This "random" variability is due to the binomial sampling plan used during data collection in an experiment; that is, only the overall number of correct responses, not the identity of individual words that are correctly recalled, is collected and recorded on each trial of the experiment.

To summarize, a mathematical model can be viewed as made of two sub-

components: (1) a deterministic component that describes the theorized functional relationships between latent and observable variables and; (2) a stochastic noise component that accounts for random variability, often due to sampling error and/or imprecise measurements. In what follows, we formalize this notion in statistical language.

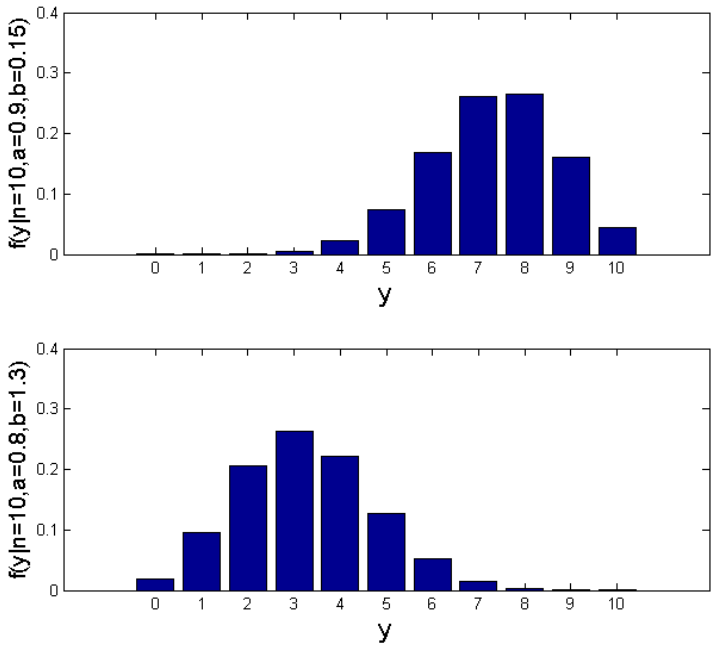

Figure 1.2 Example probability distributions. Both distributions are obtained from the binomial probability distribution in Eq. (1.2) for two different values of the parameter vector (a, b) indicated on the y-axis of each graph.

1.3.2 Formal Definition of a Model

The goal of modeling is to capture the essential features of complex behavior with a simplified mathematical model. The model is usually motivated and developed from behavioral data. From a statistical standpoint, the data consist of a set of observations, denoted by a vector $y = (y_1, ..., y_n)$, as a random sample drawn from an unknown population, or equivalently, the probability distribution that specified the probability of observing a value of y. The underlying probability distribution is to be inferred from an actual observed value of y, often referred to as just "the data."

Suppose that we have a model M with k free parameters denoted by a parameter vector $\theta = (\theta_1, ..., \theta_k)$. Formally, a model consists of a parametric collection of probability distributions indexed by the parameter θ. That is, associated with each value of θ is a unique probability distribution such that as the parameter changes in value, different probability distributions are identified, as illustrated in Figure 1.2. It is assumed that one of the model's probability distributions corresponds to the population underlying the data.

To capture the functional role of the data variable y, the probability distribution associated with each parameter value is called the *probability density function* (PDF), denoted by $f(y|\theta)$, which specifies the probability of observing the particular value of y given a fixed parameter θ. By definition, the total probability must be equal to one, $\sum_y f(y|\theta) = 1$ or $\int f(y|\theta)\, dy = 1$ for all θ, depending upon whether the random variable y is discrete or continuous, respectively. Under the assumption that the n observations are independently distributed, the PDF of the data vector $y = (y_1, ..., y_n)$ can be rewritten as a product of individual PDFs

$$f(y = (y_1, ..., y_n)|\theta) = \prod_{i=1}^{n} g_i(y_i|\theta) \tag{1.3}$$

where each $g_i(y_i|\theta)$ is the PDF of each y_i. The further assumption that the observations are *identically* distributed leads to a simpler expression of $g_i(y_i|\theta) = g(y_i|\theta), i = 1, ..., n$.

As a concrete example of Eq. (1.3), consider a lexical decision experiment in which participants are asked to decide as quickly as possible whether a string of letters is a word or a nonword, and the time taken to correctly classify words is measured as a function of word frequency. Suppose that the results suggest that the higher the word is in frequency, the faster the response time is to that word. Many different forms of mathematical functions can capture this qualitative relationship, including the following exponential model equation:

$$y_i = ae^{-bx_i} + c + e_i \tag{1.4}$$

where y_i is the lexical decision time in seconds for word i, x_i is the word frequency measured in an appropriate unit, a, b and c are positive parameters, and finally, e_i is a normal error term with zero mean and variance σ^2. The PDF for observation y_i is then given by

$$g_i(y_i|\theta) = \frac{1}{\sqrt{2\pi}\sigma}\, e^{-\frac{1}{2\sigma^2}\left(y_i - ae^{-bx_i} - c\right)^2} \tag{1.5}$$

where $\theta = (a, b, c, \sigma)$. Putting these individual PDFs together according to Eq. (1.3), we obtain the overall PDF for the data consisting of n observations as

$$f(y = (y_1, ..., y_n)|\theta) = \left(\frac{1}{\sqrt{2\pi}\sigma}\right)^n e^{-\frac{1}{2\sigma^2}\sum_{i=1}^{n}\left(y_i - ae^{-bx_i} - c\right)^2} \tag{1.6}$$

with the parameter vector $\theta = (a, b, c, \sigma)$.

1.3.3 Parameter Estimation

Once we have collected data and have specified a model for the data, the next step is to assess the model's descriptive adequacy: How well does the model fit the data? A good model should, minimally, replicate the essential characteristics of observed data. Specifically, the goal is to find the model's parameter value that best fits the data in a properly defined sense. This procedure is called *parameter estimation* in statistics (e.g., Myung, 2003; Casella and Berger, 2002).

Statisticians usually employ one of two generally accepted methods of parameter estimation. They are the *least squares estimation* (LSE) and the *maximum likelihood estimation* (MLE). In LSE, the goal is to identify the parameter value that minimizes the difference between observations and model predictions. The goal of MLE is to identify the probability distribution that is most likely to have generated the observed data. We discuss each in turn below.

Formally, LSE seeks the parameter value that minimizes the sum of squares error (SSE) between observations and predictions defined as:

$$SSE(\theta) = \sum_{i=1}^{n} \left(y_i - y_{i,prd}(\theta)\right)^2 \tag{1.7}$$

where $y_{i,prd}(\theta)$ is a model's prediction for observation i given parameter θ. The value of the parameter that minimizes the above SSE is called the *least squares estimate* denoted by θ_{LSE}. Note that LSE does not require a probabilistic specification of a model so the model does not have to be defined in terms of its PDFs, insofar as the model makes predictions that can be compared against observations. LSE results are often summarized in terms of the *root mean square error* (RMSE):

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}\left(y_i - y_{i,prd}(\theta_{LSE})\right)^2}{n}} \tag{1.8}$$

LSE, while intuitive and easy-to-interpret, is primarily a descriptive method

of parameter estimation that is developed to provide a summary of the data at hand, as opposed to making inferences about the regularity behind the data, thus gaining insights into the underlying processes. MLE is designed to serve this latter purpose.

MLE is an inferential method of parameter estimation that requires a probabilistic specification of a model in terms of PDFs. MLE provides a formal basis for many statistical methods, including chi-square goodness-of-fit testing, missing data analysis, and model selection.

Formally, MLE seeks the parameter value that maximizes the *likelihood function* of the model given observed data defined as follows:

$$Likelihood\ function:\ L(\theta) = f(y_{obs}|\theta) \tag{1.9}$$

Note that the likelihood function $L(\theta)$ is a function of model parameter θ and is obtained from the model's PDF by substituting the observed data vector y_{obs} for y. Also note that the variable in the likelihood function is the parameter θ. As such, the likelihood function does not normalize to one, i.e., $\int L(\theta)\, d\theta \neq 1$, and so it is not a probability distribution. The parameter value that maximizes the likelihood function in Eq. (1.9) is called the *maximum likelihood estimate* denoted by θ_{MLE}, whose value of course depends upon the observed data so we often denote it by $\theta_{MLE}(y_{obs})$. The particular PDF associated with the maximum likelihood estimate, that is, $f(y|\theta_{MLE})$, is called the *maximum likelihood distribution* associated with the data. It is this distribution that is most likely to have generated the data in the MLE sense.

To illustrate MLE, let us revisit the power model of memory retention discussed earlier. Eqs. (1.1) and (1.2) describe the model equation and the corresponding PDF, respectively. Suppose that we conducted an experiment with $n = 50$ Bernoulli trials and eight time intervals of $t = (0.5, 1, 2, 4, 8, 12, 16, 18)$, and recorded the number of correct responses out of 50 trials at each time interval. The observed data vector was obtained as $y_{obs} = (44, 34, 27, 26, 19, 17, 20, 11)$, or equivalently, the observed proportion correct as $p_{obs}(= y_{obs}/n) = (0.88, 0.68, 0.54, 0.52, 0.38, 0.34, 0.40, 0.22)$. The proportion data are shown as solid circles in Figure 1.3. The desired likelihood function to be maximized is given by

$$L(\theta = (a, b)) = \prod_{i=1}^{8} \frac{n!}{(n - y_{obs,i})!\, y_{obs,i}!}\, p(\theta|t_i)^{y_{obs,i}} (1 - p(\theta|t_i))^{n - y_{obs,i}} \tag{1.10}$$

where $p(\theta = (a, b)|t_i) = a(t_i + 1)^{-b}$, $n = 50$, and θ is a random variable. The maximum likelihood estimate is obtained as $\theta_{MLE} = (0.985, 0.424)$, and the

resulting best-fit power curve as $p = 0.985(t+1)^{-0.424}$, as shown as the solid line in Figure 1.3. The corresponding maximum likelihood distribution is then obtained as

$$f(y|\theta_{MLE}) = \prod_{i=1}^{8} \frac{n!}{(n-y_i)!\,y_i!}\, p(\theta_{MLE}|t_i)^{y_i}(1-p(\theta_{MLE}|t_i))^{n-y_i} \qquad (1.11)$$

where $\theta_{MLE} = (0.985, 0.424)$, $n = 50$, and y_i $(= 0, 1, ..., n)$ is now a random variable.

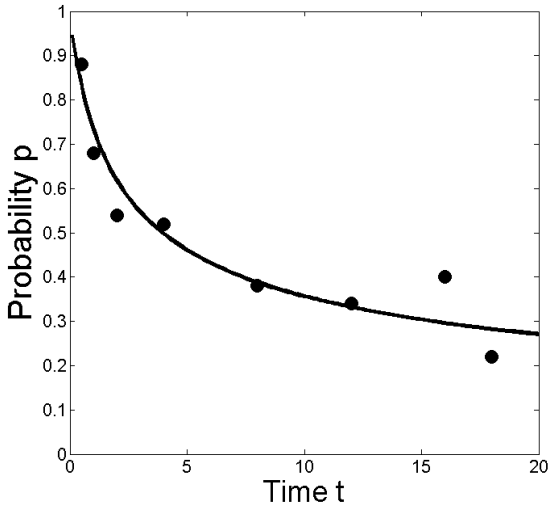

Figure 1.3 Illustration of parameter estimation. The filled circles are eight simulated observations, and the solid curve represents the MLE best-fit power curve, $p = 0.985(t+x)^{-0.424}$, obtained by maximizing the likelihood function in Eq. (1.10).

We close this section with a brief mention of how to find LSE and MLE parameter estimates. It is generally not possible to find these estimates in analytic closed-form expressions when the model is non-linear in its parameters. Instead, the solutions must be sought numerically on computer using optimization search methods, which perform "smart" searches of the parameter space iteratively until a solution is reached. For a discussion of the technical details, the reader is directed to a tutorial article by Myung (2003).

1.4 Model Evaluation

1.4.1 How Should A Model Be Evaluated?

Once a model has been fitted to data and the best-fit parameter values have been found, the questions that may arise naturally to the modeler is, Is the model any good? If so, in what sense? How persuasive is a good fit? These are the questions of *model evaluation*. A model may fit the data well, and even better than its competitor models. It does not, however, necessarily lead to the conclusion that the model successfully captures the underlying process (Roberts and Pashler, 2000) because a good fit is only a necessary, but not sufficient, condition for drawing that conclusion (e.g., Myung, 2003, p. 97). How should we then evaluate a model?

One can think of several criteria with which to evaluate a model (Myung et al., 2005). First of all and minimally, a model must be *falsifiable*, or equivalently, must satisfy the *testability* criterion. By this we mean that there must exist potential data patterns that the model cannot account for. Obviously, there would be no point of testing an unfalsifiable model that is simply a re-description of the data. As a rule of thumb, if the number of a model's parameters is equal to or greater than the number of observations in a data set, so that the model has zero or negative degrees of freedom, then the model would be unfalsifiable. For example, for the retention data in Fig 1.3 consisting of eight observations, the following 8-parameter power model would be unfalsifiable: $p = a(t+b)^{-c} + de^{-et}sin(ft+g) + h$, where $\theta = (a, b, c, d, e, f, g, h)$. It turns out, however, that the aforementioned counting rule does not always work, especially for non-linear models, for which more sophisticated rules must be used to determine model falsifiability (Bamber and van Santen, 1985, 2000).

Explanatory adequacy is another criterion of model evaluation. A model should provide insights that are are generally not possible to gain otherwise. The model should also be *plausible* in that its assumptions make sense and are consistent with established biological and psychological findings. Relatedly, the model should be *interpretable* as well so that each of its parameters permits interpretation in terms of a known psychological process or construct. Further, the model should be *faithful* in that its ability to account for the underlying process derives from the theoretical principles the model substantiates but not from the subsidiary assumptions the model makes in its computational implementation (Myung et al., 1999). While it is important to consider these four criteria (*explanatory adequacy, plausibility, interpretability*, and *faithfulness*) in model evaluation, given the qualitative (as opposed to quantitative) nature of their definitions, the modeler will

have to apply them in a subjective and sensible manner in assessing the viability of the model under consideration.

On the other hand, there exist other criteria that are quantifiable and thus entail quantitative metrics by which the model can be evaluated. They are (1) *goodness-of-fit* (the extent to which a model fits observed data); (2) *model complexity* (a model's inherent flexibility to fit a wide variety of data patterns); and (3) *generalizability* (a model's ability to predict new observations). In what follows, we discuss each of these three quantitative measures in turn and their interrelationships with one another in greater depth.

1.4.2 Goodness-of-fit and the Overfitting Problem

A common but misleading practice of modeling is to evaluate a model solely on the basis of its *goodness-of-fit* (GOF), that is, how well the model fits the data at hand. GOF measures that are commonly used include *root mean square error* (RMSE) in Eq. (1.8), *percent variance accounted for* (PVAF), and *maximized likelihood* (ML). The latter two are defined as

$$\begin{aligned} PVAF &= 100\,(1 - SSE(\theta_{LSE})/SST) \\ ML &= L(\theta_{MLE}) \end{aligned} \tag{1.12}$$

where $SSE(\theta_{LSE})$ is the maximized sum of squares error in Eq. (1.7) and SST is the sum of squares total defined as $SST = \sum_i (y_i - y_{mean})^2$.

Off-hand, GOF measures may seem like good metrics by which to measure the model's ability to describe the underlying process being modeled. This would make sense if observations were free of random error. In practice, however, behavioral measurements are invariably corrupted with noise and artifacts of various kinds that have little to do with the regularity of interest. In other words, conceptually, GOF contains two unrelated components:

$$GOF = Fit\ to\ regularity \quad + \quad Fit\ to\ noise. \tag{1.13}$$

It is the first component we are interested in, but not the second one. It is, however, generally not possible to disentangle the separate effects of the two factors. Consequently, a GOF measure gives the overall value of their sum. What makes matters worse is the fact that there is a property of a model that enables it to fit random noise to an arbitrary extent, independent of its ability to fit the regularity. The case in point is a complex model with many parameters and a highly nonlinear model equation that may absorb noise rather easily, but without actually capturing the underlying regularity. As a result, a complex model often provides a better fit than a simpler model that

has actually generated the data. This is why a good fit can be misleading (Pitt and Myung, 2002).

The problem of a model fitting random noise over and above the underlying relationship is known as *overfitting* in statistics. This is illustrated in Figure 1.4. The solid circles are observed data and the curves represent best-fits by two hypothetical models that differ in the number of parameters. Model A is a simple model with fewer parameters than Model B, and does a good (though not perfect) job of describing the variation in the data including the general trend, but also predicts well a new data point (represented by the star symbol in Figure 1.4). In contrast, Model B, with its extra parameters, fits much better than Model A, but this superior goodness-of-fit is achieved at the cost of failing to predict the new data point. Despite its extra complexity, Model B predicts the new data point no better than, and perhaps worse than, Model A.

In conclusion, assessing the adequacy of a model based solely on goodness-of-fit (GOF) can result in overfitting. The overfitting occurs because GOF can be improved arbitrarily by increasing the complexity of a model. What is model complexity then, and how do we measure it? This is the topic of the next section.

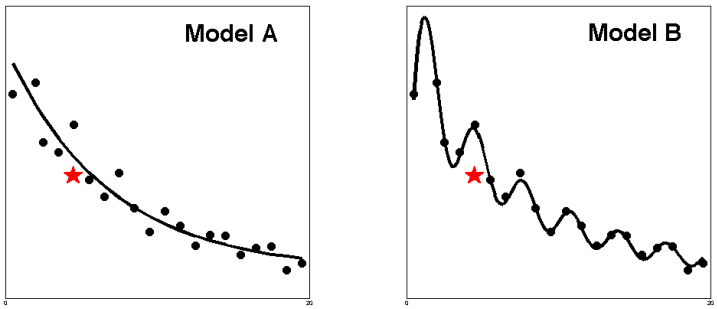

Figure 1.4 Schematic illustration of the trade-off between goodness of fit and model complexity. The complex model (Model B) provides a superior fit to the data (solid circles) than the simple model (Model A) but does a poor job in predicting a new observation (star).

1.4.3 Model Complexity

Intuitively, *model complexity* , or flexibility, refers to the property of a model that enables it to fit a wide range of data patterns, regardless of whether

they represent the regularity of interest or the idiosyncratic random noise. Model complexity comes in at least two dimensions: (1) the number of model parameters; (2) the functional form of model equation. The first dimension of complexity (number of parameters) is well established in statistics: A model with many parameters is more complex than the one with fewer parameters.

On the other hand, the second dimension of complexity is less obvious. This *functional form complexity* refers to the way in which the parameters of a model are combined in its model equation (Myung and Pitt, 1997). As an example, the power and exponential models of retention memory, $p = a(t+1)^{-b}$ and $p = ae^{-b}$, respectively, have the same number of parameters (two) but differ in functional form, and they may therefore have different complexity. Pitt et al. (2002, Figure 4) presents a striking example of variation in model complexity in which three one-parameter models differ widely in their ability to describe a range of data patterns.

In the literature, several approaches to quantifying model complexity in numerical metrics have been proposed. In what follows, we introduce and discuss two such measures.

Perhaps the most sophisticated measure of model complexity (MC) is the one derived from *Normalized Maximum Likelihood* (NML; Rissanen, 2001)[1], which takes the following form:

$$MC_{NML} = \ln \int L\left(\theta_{MLE}(z)\right) dz \tag{1.14}$$

where $L(\theta_{MLE}(z))$ denotes the maximized likelihood (ML) given a data vector z. Note in the above equation that the model complexity is expressed as an integration of the ML over the entire data space. Accordingly, the NML model complexity is defined conceptually as the *sum of all best-fits* the model can provide collectively for all potential (not just actually observed) data that can be realized in an experimental setting. The larger the sum (or integral, in the continuous case) is, the more complex the model is. As such, this complexity measure represents a formalization of what is referred to intuitively and informally as a model's ability to fit a wide range of data patterns.

The NML complexity in Eq. (1.14), though conceptually elegant, does not clearly reveal what constitutes model complexity, that is, what dimensions of complexity are captured in the measure. This is revealed in an asymptotic approximation of NML complexity that is known as the *Fisher Information*

[1] The NML is a method of model selection to which we return later in this chapter

Approximation (FIA) model complexity (Rissanen, 1996; Su et al., 2005)[2],

$$MC_{FIA} = \frac{k}{2} \ln \frac{n}{2\pi} + \ln \int \sqrt{det(I(\theta))}\, d\theta \tag{1.15}$$

where k is the number of parameters, n is the sample size (i.e., number of indepent identically distributed observations), $I(\theta)$ is the Fisher information matrix of sample size 1,[3] *det* denotes the determinant of a matrix, and finally, *ln* is the logarithm of base e. The first term of the right hand side of the equation captures the effects of complexity due to the number of parameters (k), and the second term captures the functional form effects of complexity through $I(\theta)$. Both dimensions of complexity are therefore reflected in MC_{FIA}. We note further that the magnitude of the first term increases logarithmically with the sample size n whereas the second term does not depend upon n. An implication of this observation is that in the limit of large sample sizes, the functional form effects of complexity become negligible relative to the effects due to the number of parameters, thereby effectively making the former the sole contributor to model complexity, as conventionally conceptualized.

To summarize, we have learned so far that goodness-of-fit (GOF) is a necessary but not sufficient criterion of model adequacy. We also learned that the model must be sufficiently complex to capture the regularity in the data but not so complex as to overfit the data by capitalizing on random error. It seems, then, that a good model is one that strikes the "right" balance between GOF and model complexity, both of which can be objectively measured in quantifiable terms. What is the right balance? Answering this question is tied directly to the goal of modeling that in turn leads us to the third quantifiable criterion of model evaluation, *generalizability*.

1.4.4 Generalizability

The goal of modeling is to deduce the model that generated the observed data. In reality, however, this is not possible because of two fundamental limitations: (1) *Finiteness of data*: Observations in a data set may never be sufficient to exactly and uniquely identify the ground truth; and (2) *Complexity of truth*: The truth may be quite complex well beyond anyone's imagination and thus not among the models under consideration. This second limitation is another reminder of G.E.P. Box's quote that all models

[2] The FIA is another method of model selection that is discussed later in this chapter

[3] The Fisher information, defined in terms of the covariances of the second-order partial derivatives of the log likelihood function, $\ln L(\theta)$, with respect to the parameters, in essence measures the amount of information in data about model parameters (e.g., Schervish, 1995).

are wrong. Given these challenges, a more realistic, and perhaps achievable, goal is to identify a model that is deemed the "best possible" approximation to the underlying truth in a defined sense. The current consensus in the field is that the best (and so most useful) approximate truth is the model with highest generalizability.

Generalizability (GN) is defined as a model's ability to fit not only the observed data at hand but also new data from the same process that has generated the current data. Put another way, this criterion, often known as *predictive accuracy*, refers to how accurately the model predicts future observations. GN is indeed the ultimate yardstick and gold standard of model evaluation by which all models with varying degrees of goodness-of-fit and complexity are to be judged for their usefulness. Revisiting the models in Figure 1.4, Model A clearly generalizes better and would therefore be judged as a closer approximation to the underlying process being modeled.

To reiterate, the central creed of model evaluation is good generalizability. This is achieved by striking the right balance between goodness of fit and model complexity by trading off one for the other. In other words, the model should be no more complex than what is necessary to extract the underlying regularity. It is in this sense that the generalizability criterion can be regarded as a formal embodiment of the principle of *Occam's razor*, "Entities should not be multiplied beyond necessity" (William of Occam, 1288-1348).

Generalizability can be made more precise and rigorous in formal terms. In doing so, let us first define the notion of discrepancy between two probability distributions. Specifically, a discrepancy function $D(f, g)$ between two distributions, f and g, is a continuous real-valued positive function that satisfies the condition of $D(f, g) > D(f, f) = 0$ for all $f \neq g$ (e.g., Linhart and Zucchini, 1986). The well-known Kullback-Leibler information divergence is an example of such a *discrepancy function*. The smaller the $D(f, g)$, the more similar the two distributions are to each other, and thus, the better the distribution f approximates the distribution g, and vice versa. As such, $D(f, g)$ is a kind of "distance" measure between two distributions, though the discrepancy itself does not necessarily satisfy the symmetric condition, i.e., $D(f, g) = D(g, f)$. In terms of the discrepancy function, a formal definition of the generalizability for model M is given as (e.g., Su et al., 2005, p. 413)

$$E\left[D(f_T, f_M)\right] = \int D\left(f_T, f_M(\theta_{MLE}(y))\right) f_T(y)\, dy \tag{1.16}$$

where $f_M(\theta_{MLE}(y))$ denotes the model's *maximum likelihood distribution* given a data vector y and $f_T(y)$ is the probability distribution that generates observed data (i.e., ground truth). As shown above, generalizability is defined as a mean discrepancy or "distance" between the true model distribution and the best-fitting distribution of the model family of interest, averaged over all possible data under the true model. It is in this sense that the model with highest generalizability can be regarded as the "best possible" approximation to the truth.

1.4.5 Relationship among Goodness-of-fit, Complexity, and Generalizability

Figure 1.5 illustrates the relationships among the three quantitative criteria discussed so far: goodness of fit, model complexity, and generalizability. There are a few main points to make from the figure. First, as indicated by the upper curve in the figure, goodness of fit can always be improved by increasing model complexity. Second, increasing complexity only improves generalizability up to a certain point, as shown by the lower curve. Third, the most preferred model given observed data is the one with the complexity corresponding to the peak of the generalizability curve. This is the model whose complexity best matches the complexity in the data. The model is sufficiently complex to capture the regularity but not overly complex to start absorbing random noise in the data. Fourth, as indicated in the figure, overfitting manifests itself as the difference between the goodness of fit and generalizability curves above and beyond the optimal point of model complexity.

The three smaller graphs in the bottom of Figure 1.5 provide concrete examples. In the left graph, the model (line) is not complex enough to match the complexity of the data (dots). The model and data are well matched in complexity in the middle graph, which is why this occurs at the peak of the generalizability function. In the right graph, the model is more complex than the data, capturing incidental variation due to random error.

1.5 Model Selection

The preceding section describes the dimension on which models should be evaluated. How are they implemented in practice to guide researchers in choosing one model over the others? In this section, we review methods of model selection that are currently in use. For a more thorough treatment of the topic, the reader is directed to two *Journal of Mathematical Psychology*

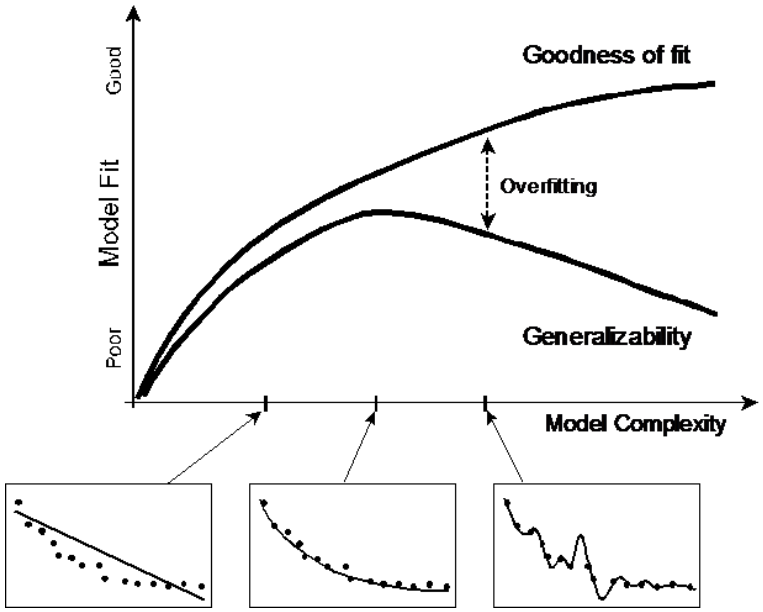

Figure 1.5 An illustration of the relationships among goodness of fit, model complexity, and generalizability as a function of model complexity. The vertical axis represents a model fit index, where a larger value indicates a better fit (e.g., percent variance accounted for). Reprinted from Pitt and Myung (2002).

special issues (Myung et al., 2000; Wagenmakers and Waldorp, 2006), and an excellent tutorial article on the topic (Shiffrin et al., 2008).

The central tenet of model selection, as discussed earlier, is to choose, among a set of candidate models being compared, the one that generalizes best, or equivalently, the one that provides the closest approximation to the truth. One may then use the generalizability measure in Eq. (1.16) for the purpose of identifying the best generalizing model. Unfortunately however, this measure cannot be directly computed as it is defined in terms of the true distribution $f_T(y)$, which is unknown or unknowable. Consequently, the generalizability measure must be *estimated* from observed data. Virtually all methods of model selection including the ones we discuss here can be seen as generalizability estimates of one kind or another.

1.5.1 Penalized-Likelihood Model Selection

What is wanted in model selection is, again, a method that estimates a model's generalizability by taking into account the effects of model complexity on model fit. In other words, model selection is in essence about

achieving a balance between two opposing forces, model complexity on one side and goodness-of-fit on the other. In each of the four methods of model selection we introduce in this section, this is instantiated by penalizing the model under consideration for excessive and unnecessary complexity, that is, the portion of its complexity that is more than what is needed to capture the regularity in the data, thereby putting all the models on an equal footing so to speak.

The four methods are the *Akaike Information Criterion* (AIC; Akaike, 1973), the *Bayesian Information Criterion* (BIC; Schwarz, 1978), the *Fisher Information Approximation* (FIA; Rissanen, 1996; Su et al., 2005), and the *Normalized Maximum Likelihood* (NML; Rissanen, 2001). They are defined as

$$\begin{aligned} AIC &= -2\ln L(\theta_{MLE}(y)) + 2k \\ BIC &= -2\ln L(\theta_{MLE}(y)) + k\ln(n) \\ FIA &= -\ln L(\theta_{MLE}(y)) + \frac{k}{2}\ln\frac{n}{2\pi} + \ln\int\sqrt{det(I(\theta))}\,d\theta \\ NML &= -\ln L(\theta_{MLE}(y)) + \ln\int L\left(\theta_{MLE}(z)\right)\,dz. \end{aligned} \tag{1.17}$$

In the above equation, y is a vector of observed data, $L(\theta_{MLE}(y))$ $(= f(y|\theta_{MLE}))$ is the maximized likelihood of the data, z is a vector variable of potential data, and finally, k, n and $I(\theta)$ are defined earlier in Eqs. (1.14) and (1.15).

Each criterion above consists of the first term representing a lack-of-fit measure and the second- and remaining terms representing a model complexity measure. Combined, they estimates a model's generalizability such that a lower value of the overall criterion indicates better generalizability. Accordingly, the criterion prescribes that among a set of competing models, the one with the lowest criterion value should be selected as the best-generalizing model.

Notice two counteracting forces at work in each criterion: An increase in the second- and remaining terms (i.e., increasing complexity) generally results in a decrease in the first term (i.e., better goodness-of-fit), and vice versa. A logical corollary of this asymmetry is that the criterion implicitly penalizes the model with excess complexity. That is to say, a model may provide a superior goodness of fit over other models being considered, but that alone does not necessarily make it a better generalizing model. This is because that model's complexity may be too large to the extent of causing a net positive increase in the overall criterion value, thereby in essence creating an overfitting situation. In short, the aim is to achieve the "optimal" trade-off

between the two forces so as to avoid overfitting as well as underfitting. The notion of optimality is conceptualized and defined differently for different model section methods, which we discuss below one at a time.

The AIC criterion is historically the first method of model selection that has been introduced for choosing among nonlinear and non-nested models, and is rooted in information theory (Cover and Thomas, 1991). Specifically, AIC is derived as a large sample (i.e., asymptotic) approximation of the generalizability measure defined in Eq. (1.16) in which the discrepancy function $D(f, g)$ is the Kullback-Leibler information divergence between the true probability distribution and the maximum likelihood distribution of the model under consideration. Accordingly, AIC selects one model, among a set of competing models, that has the closest distance to the truth in an information theoretic sense. One shortcoming, however, is that from the AIC standpoint, the number of model parameters (k) is the sole contributor to model complexity, thereby ignoring other relevant and potentially significant factors such as sample size and function form.[4]

The BIC criterion is a Bayesian statistical criterion and is derived as an asymptotic approximation of the Bayesian Model Selection (BMS), which is introduced later in this chapter. The basic idea of BMS (hence also BIC) is to identify the model that is most likely to have generated observed data in the sense of Bayesian probability theory (e.g., Gelman et al., 2013). Notice that the model complexity term of the BIC includes the contribution of the sample size (n) as well as that of the number of free parameters (k). As such, as the sample size, or the number of observations, increases, BIC tends to favor models with fewer parameters, unlike AIC that does not take into account the sample size factor.

Both FIA and NML criteria are methods of model selection derived from the principle of *Minimum Description Length* (MDL; Grünwald et al., 2005; Myung et al., 2006; Grünwald, 2007) in computer science, with FIA being an asymptotic approximation of NML. According to the principle of MDL, the goal of modeling is to compress the data as tightly as possible without loss of information; the model is viewed as a code with which to encode the data, and finally, the best model is the one that provides the shortest description length of the data in bits. To elaborate further, the more the model is able to extract regularities in data structure, the better the model can compress the raw data with the help of the uncovered regularities, thus providing a shorter description length of the data. This in turn leads to

[4] To be fair, the original derivation of AIC (Akaike, 1973) does include higher-order terms that reflect the effects of sample size and functional form, but the terms are subsequently dropped at the later stages of the asymptotic expansion for the sake of simplicity of computation.

better generalization because the model can now use the extracted regularities to predict future data accurately. In short, regularity extraction, data compression, and generalization are three variations of essentially the same idea. Regarding how model complexity is conceptualized and implemented in FIA and NML, as discussed earlier in this chapter, the complexity terms of both criteria are sensitive to the number of parameters, the sample size, and importantly, the functional form. Of particular note is that the NML complexity in Eq. (1.14) is not only intuitively appealing but also represents a "full and complete" view of model complexity. For a technically rigorous treatment of this and related material, the reader is directed to an excellent tutorial article by Grünwald (2005). Example applications of FIA and NML to model selection in cognitive psychology can be found in Wu et al. (2010), Klauer and Kellen (2011), and Singman and Kellen (2013), for example.

1.5.2 Cross-Validation and Accumulative Prediction Error

Each of the four methods of model selection discussed in the preceding section estimates a model's generalizability through an equation that specifies explicitly how goodness-of-fit should be traded off for model complexity. In contract, *cross-validation* (CV; Stone, 1974; Browne, 2000) and the *Accumulative Prediction Error* (APE; Dawid, 1984; Wagenmakers et al., 2006) that we introduce in this section estimate generalizability directly from the observed data by simulating the data-generation and model-prediction steps, but without relying upon a formulaic equation with an explicit measure of model complexity.

Both CV and APE are implemented in the following three-step procedure. First, the data sample is split into two non-overlapping subsamples, the calibration sample denoted by y_{cal} and the validation sample denoted by y_{val}. Second, the model of interest is fit to the calibration sample y_{cal} and the model's MLE, $\theta_{MLE}(y_{cal})$, is obtained. Third, the model is then fit to the validation sample y_{val} directly with its parameter values *fixed to* $\theta_{MLE}(y_{cal})$. The resulting fit, or prediction error, to the validation sample is taken as the model's generalizability estimate. The two methods, CV and APE, differ from each other in how the data are split into calibration and validation samples. Even within CV, there exist different variations of this general scheme depending upon how the calibration-validation split is defined.

In what is known as the *split-half CV*, the observed data are divided randomly into two subsamples of equal size, one half for the calibration and the other half for the validation. This split-half CV method is illustrated in Figure 1.6 for a hypothetical model. As shown in the figure, the data of 24

observations in the top panel are split into a calibration sample consisting of 12 blue (darker) filled circles in the lower left panel and a validation sample consisting of 12 red (lighter) filled circles in the lower right panel. The solid curve in the lower left is the best-fit MLE curve to the calibration sample. How well this solid curve fits the calibration sample defines the model's goodness-of-fit (GOF). The same curve, now denoted by the dotted curve in the lower right, is fitted directly to the validation sample without further parameter tuning. How well this dotted curve fits the validation sample defines the model's generalizability estimate. One drawback of split-half CV is that its generalizability estimate depends upon the particular way the data are divided into two equal halves and further, that there are practically an infinite number of ways to do the splitting. Another CV procedure we discuss next gets around this problem by adopting an unequal splitting rule.

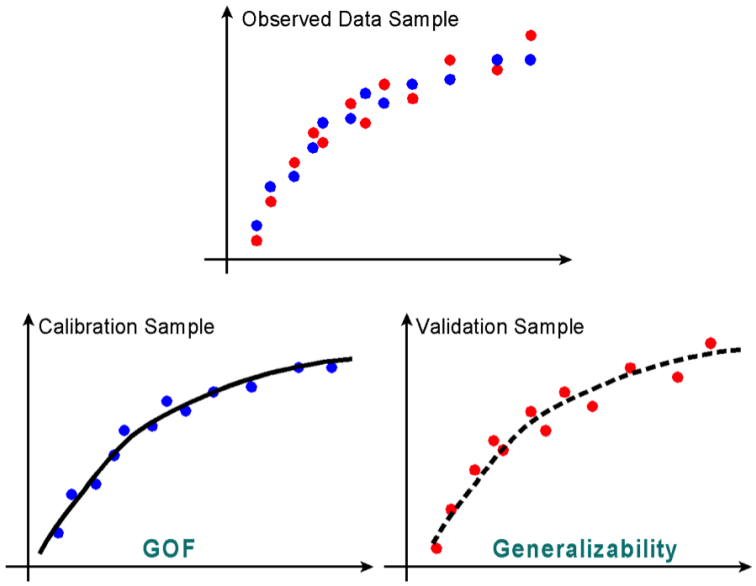

Figure 1.6 Illustrated scheme of split-half cross-validation.

In *leave-one-out-cross-validation* (LOOCV), the data sample of n observations, denoted by a vector $y = (y_1, ..., y_n)$, is split into a calibration sample of $(n-1)$ observations and a validation sample of the remaining one observation, and the model's generalizability is then estimated according the three-step procedure mentioned above. This $(n-1)$-vs-1 split process is repeated for all possible n splits. The model's *final* generalizability is obtained as the arithmetic mean of n individual generalizability estimates, formally

expressed as

$$LOOCV = -\sum_{i=1}^{n} ln\, f\left(y_i|\theta_{MLE(y_{\neq i})}\right). \tag{1.18}$$

APE is similar to LOOCV in spirit, but differs in implementation such that generalizability is estimated in a sequential and accumulative manner, instead of the $(n-1)$-vs-1 split between calibration and validation samples. To elaborate, given the data of n observations and a model with k parameters, the data are split into a calibration sample consisting of the *first* $(k+1)$ observations and a validation sample of the remaining $(n-k-1)$ observation. The model's generalizability with respect to this particular $(k+1)$-vs-$(n-k-1)$ split is estimated following the same three-step procedure. The calibration sample is then increased in size by one observation by taking in the next $(k+2)$-*th* observation, and the model's generalizability with respect to the new $(k+2)$-vs-$(n-k-2)$ split is again estimated. This successive and accumulative process continues until there is only one observation left in the validation sample. The model's *final* generalizability is obtained as the arithmetic mean of $(n-k-1)$ individual generalizability estimates, formally expressed as

$$APE = -\sum_{i=k+2}^{n} ln\, f\left(y_i|\theta_{MLE(y_{1,2,\ldots,i-1})}\right). \tag{1.19}$$

Both cross-validation and the accumulative prediction error prescribe that among a set of competing models, the one with the lowest value of the given criterion should be selected as the best generalizing model. There are several characteristics that make these methods appealing alternatives to the penalized-likelihood methods of model selection in Eq. (1.17). One is their ease of computation. CV and APE can easily be implemented; all that is needed is the calculation of maximum likelihood estimates. This is unlike FIA and NML that involve high dimensional integration, which can be non-trivial to compute numerically. Another appealing feature of CV and APE is that they supposedly (albeit implicitly) take into account the effects of all three dimensions of model complexity, that is, the number of parameters, the sample size, and the functional form. Accordingly, performance of CV and APE should generally be superior to that of either AIC or BIC, which does not consider all of these dimensions of complexity.

1.5.3 Bayesian Model Selection

Bayesian model selection (BMS; Kass and Raftery, 1995; Wasserman, 2000

is the standard, state-of-the-art method of model selection for Bayesian inference and is defined as the minus logarithm of the *marginal likelihood* of the model of interest,

$$BMS = -ln \int L(\theta(y))p(\theta)\, d\theta \tag{1.20}$$

where y is a vector of observed data, $L(\theta(y))$ $(= f(y|\theta))$ is the likelihood function of the data defined in Eq. (1.9), and $p(\theta)$ is the parameter prior distribution. BMS prescribes that the model with the lowest BMS value should be preferred. It is worth noting that BMS is closely related to the Bayes factor, which is defined as the ratio of two marginal likelihoods between a pair of competing models, in such a way that either criterion, BMS or Bayes factor, always leads to the same model choice.

Note in Eq. (1.20) that the marginal likelihood, $\int L(\theta(y))p(\theta)d\theta)$, is nothing but the mean likelihood obtained by averaging the likelihood across all parameter values and weighted by the parameter prior. This Bayesian averaging is exactly how BMS avoids overfitting, that is, by selecting the model with the highest *mean* likelihood value, instead of the one with the highest *maximum* likelihood value. The latter would necessarily result in overfitting. In other words, model complexity is automatically adjusted in BMS through the built-in averaging operation. In so doing, the method considers all three dimensions of complexity; this can be seen more clearly in an asymptotic approximation of BMS (Balasubramanian, 1997), which turns out to be just the same as FIA in Eq. (1.17)! This points to a potentially intriguing connection between minimum description length and Bayesian model selection, despite the fact that they are rooted in divergent theoretical and philosophical foundations. A further approximation of BMS leads to one-half of BIC so that the latter can be considered as a quick and rough version of the former. Finally, the computation of BMS would be in general nontrivial as it involves numerical integration of high dimensions. We noted earlier that similar computational difficulties plague the routine use of FIA and NML.

1.5.4 Illustrated Example

In this section, we present and discuss an illustrated example of the four model selection methods (AIC, BIC, LOOCV, APE).[5]. Two goodness of fit measures, PVAF and ML in Eq. (1.12), are also included for the comparison purpose.

[5] FIA and NML are not included in the example due to the computational challenges to implement them.

Four retention models are compared in terms of their generalizability estimated by each of the four selection criteria. The four models are as follows:

$$\begin{aligned} POW: &\ p = a(t+1)^{-b} \\ POW2: &\ p = a(t+1)^{-b} + c \\ POW3: &\ p = a(t+1)^{-b} + c + d \cdot sin(e \cdot t) \\ EXP: &\ p = ae^{-bt}. \end{aligned} \tag{1.21}$$

Note that POW and EXP have two parameters, POW2 has three parameters, and POW3, the most complex among the four, has five parameters. Each of these models was fitted to the data shown as the solid circles in Figure 1.3 with the binomial likelihood function in Eq. (1.10). The MATLAB source code of this simulation is included in the Appendix.

The fitted data and best-fit curves of the four models are shown in Figure 1.7. The model selection results are reported in Table 1.1. Let us first examine goodness of fit performance of the four models. As expected, the more parameters a model has, the better the model fits the data. Not surprisingly, POW3, the most complex model among the four, provided the best fit, capturing 98.7% of the total variance, which is perhaps an overfitting. The two-parameter exponential model (EXP) fared poorly, only capturing 79% of the variance. Accordingly, it seems this model can be safely ruled out from further consideration. The log maximum likelihood results based on *LogLik* values lead to essentially the same conclusion.

On the other hand, the generalizability results lead us to different conclusions. Both AIC and BIC prefer the simplest power model (POW) as the best-generalizing model among the four under consideration. In other words, according to AIC and BIC, the other power models, POW2 and POW3, apparently overfit the data. Interestingly enough, however, LOOCV and APE results draw a different picture. Both of these methods select POW3, the most complex of the four, as the best-generalizing model! This model choice is, obviously, in direct contradiction with that based on AIC and BIC, and is thus somewhat of a surprise. Since the underlying truth is unknown in this case and also given the relatively small size of the data, it is difficult to discern the possible causes and reasons for these conflicting results so we would have to take them at face value. That is, different methods of model selection can sometimes lead to differing interpretations of the same data.

To summarize, we demonstrated application of the model selection procedure to the problem of choosing among a set of models that differ not only in the number of parameters but also in functional form. The reader should not over-generalize the particular results from this example application, which

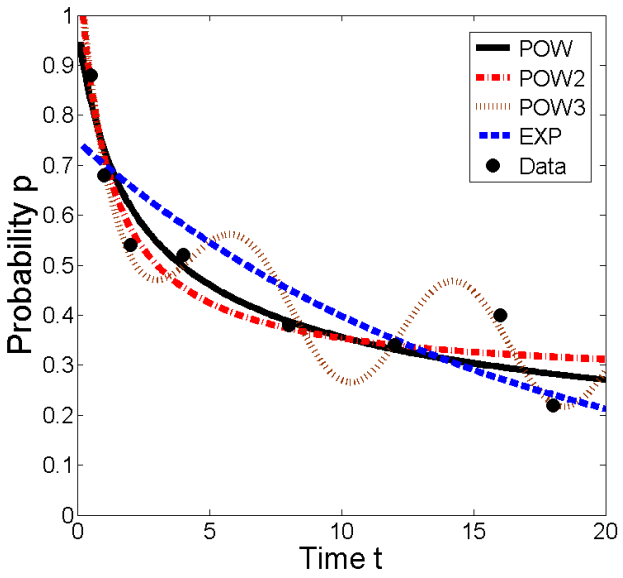

Figure 1.7 Illustration of model selection for four models of memory retention in Eq. (1.21). The filled circles are the data and the curves represent the best-fit model predictions.

Table 1.1 *Model selection results for four retention models defined in Eq. (1.21). The LogLik stands for the log maximum likelihood. The best-fit parameters were sought by maximizing the likelihood function in Eq. (1.10) without the constant, parameter-independent term,* $n!/\left((n-y_i)!\,y_i!\right)$.

Model	POW	POW2	POW3	EXP
Number of params	2	3	5	2
PVAF	91.2	92.6	**98.7**	79.0
LogLik	-247.34	-246.68	**-244.48**	-252.06
AIC	**498.67**	499.35	498.96	508.11
BIC	**502.50**	505.09	508.53	511.94
LOOCV	31.41	31.52	**30.80**	32.53
APE	32.63	31.94	**30.64**	35.57

was simply intended to serve as an illustration of model selection–but no more. We conclude this section with a quote from Myung and Pitt (2004, p. 365) on the importance of viewing model selection as a statistical inference problem:

Model selection is an inference problem. The quality of the inference depends strongly on the characteristics of the data (e.g., sample size, experimental design, type of random error) and the models themselves (e.g., model equation, parameters, nested vs nonnested). For this reason, it is unreasonable to expect a selection method to perform perfectly all the time.

1.5.5 Summary

Computational modeling has become an important tool for advancing the study of mind and brain and has contributed substantially to theorizing and experimentation in cognitive psychology. The success of modeling depends upon the availability of theoretically sound methods for comparing and selecting among computational models. Often, a number of models (or theoretical explanations) can account for a given set of empirical data, and it is not clear how one can best choose among these competing models. At its most basic level, this is a problem of uncertainty of inference from data to the model, and model selection (comparison) methods help reduce this uncertainty by using sound statistical methods.

In this section we reviewed quantitative approaches that guide model selection and thus improve scientific inference. The main take-home messages from this review can be summarized into the following four steps:

- Step 1 (Goodness-of fit): Evaluate each model's fit, among a set of candidate models being evaluated, to observed data to assess its descriptive adequacy, or goodness of fit.
- Step 2 (Model complexity): Consider the model's inherent flexibility to fit other potential data that could be collected.
- Step 3 (Generalizability): Estimate the model's generalizability by properly trading off goodness of fit for model complexity using a given method of model selection.
- Step 4 (Model choice): Choose the model with the best generalizability.

To reiterate, models should be evaluated based on generalizability, not on goodness of fit, as echoed by the following statement: *Thou shall not select the best-fitting model but shall select the best-generalizing model.*

1.6 Design Optimization

1.6.1 Further Improving Model Selection through Design Optimization

Statistical tools for model selection that we have discussed in the preceding sections are developed to assist researchers in making inferences about models given data samples collected in experiments. However, because such tools are applied *after* data have been collected, their potential to yield definitive conclusions is limited by the quality of the empirical data that they have to work with; sometimes the data simply do not provide clear differentiation between models. As mentioned previously, part of the problem stems from the fact that different models can mimic one another. That is, many sets of data can be explained just as well by one model as by another.

For example, consider the hypothetical set of retention data shown below in Figure 1.8, which were generated by computer from the model POW, which is defined in Eq. (1.21) with $a = 0.8$, $b = 0.5$, and 50 Bernoulli trials at each of 8 different retention intervals: 5, 15, 25, 35, 45, 55, 65, and 75 seconds. Suppose, for the purposes of illustration, that one did not know the true data-generating model and wished to determine whether these data were more likely to have been generated by POW or a competing model, EXP also as defined in Eq. (1.21). The BIC, introduced in the preceding section, is an appropriate criterion for this task, as the model with the lower BIC is more likely to have generated the observed data in the sense of Bayesian probability theory. However, for these data, the BICs for POW and EXP are quite similar: 49.1 for EXP and 48.3 for POW. This difference between these two values, ΔBIC=0.8, means that POW is slightly more likely to have generated the data, but the margin is razor thin so the result is inconclusive.

Now consider a different set of hypothetical retention data, shown in Figure 1.9. These data were generated from the same model as in the preceding example, with the same number of Bernoulli trials at each retention interval, but at a different set of 8 retention intervals: 1, 3, 5, 10, 15, 20, 40, and 80 seconds. If we perform the same model selection analysis on these data, we get a BIC of 57.8 for EXP and 49.0 for POW. This result, ΔBIC=8.8, means that POW is much more likely to have generated the data than EXP. In other words, these data strongly identify POW as the generating model.

The preceding examples illustrate that our ability to successfully discriminate between models can sometimes hinge on decisions that are made *before* data are collected. For instance, in collecting retention data, one must first choose retention intervals at which to test memory. In the first example,

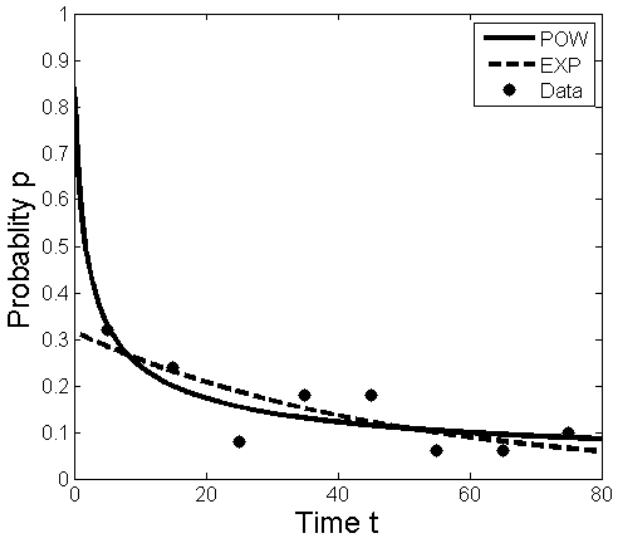

Figure 1.8 Example of a poor experimental design for discriminating between power (POW) and exponential (EXP) models of retention. The design consists of eight equally spaced time intervals $\{5, 15, 25, ..., 75\}$. Both models provide equally good descriptions of the data obtained from this design ($BIC_{POW} = 48.3$ vs. $BIC_{EXP} = 49.1$) and thus the are not discriminated.

memory was tested at retention intervals that were roughly evenly spaced between zero and 80 seconds, while in the second example, testing was concentrated at intervals near zero seconds, with far fewer observations in the range of 20 to 80 seconds. It turned out that the latter choice of retention intervals improved the informativeness of the data for discriminating between the models.

In general, many such decisions about the design of an experiment must be made before data collection can begin, including the number of treatment groups, the sample size in each treatment group, and the timing and order of stimuli, among others. The precise set of relevant design variables will vary from experiment to experiment. The settings of these variables are referred to as the experimental design, and the experimenter's choices regarding the experimental design affect not only the potential statistical value of the results, but also the cost of the experiment in terms of time, money, and participants. Therefore, an optimal experimental design can be regarded as one that maximizes the informativeness of the experiment while being cost effective for the experimenter.

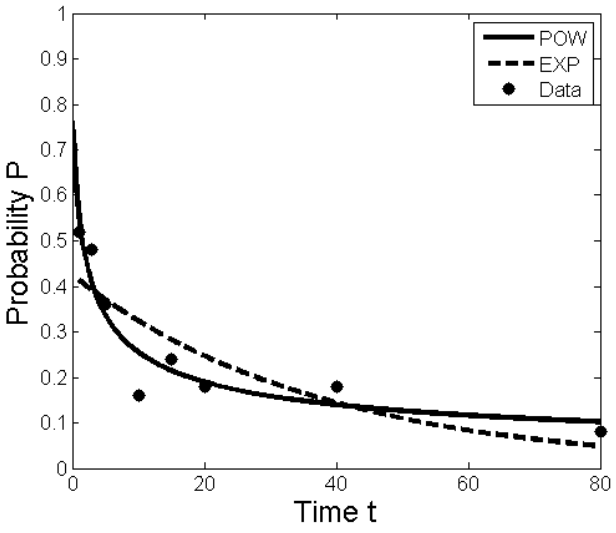

Figure 1.9 Example of a good experimental design with a different set of eight time intervals {1,3, 5, 10, ...,80}, than the one shown in Figure 1.8. Note that in terms of BIC, model POW provides a much better description of the data obtained from this design than model EXP: $BIC_{POW} = 49.0$ vs. $BIC_{EXP} = 57.8$.

1.6.2 Design Optimization

To optimize design decisions and maximize the chances of discriminating between models, *Design Optimization* (DO) applies statistical inference to evaluate design decisions at the front-end of an experiment (i.e., before data have been collected), in order to make model evaluation easier at the back end of the experiment (i.e., after data have been collected). There is a rich literature in statistics on the problem of design optimization dating back to the 1950s (e.g., Kiefer, 1959; Atkinson and Federov, 1975; Atkinson and Donev, 1992; Chaloner and Verdinelli, 1995). Here we introduce the reader to the conceptual framework of DO as it relates to the optimal design of psychological experiments. For in-depth details of the theoretical and computational aspects of DO, the reader is directed to other publications from our lab (Myung and Pitt, 2009; Cavagnaro et al., 2010; Myung et al., 2013).

The implementation of the DO approach in practice requires that one must first define the design space, which consists of the set of all possible values of design variables that are controlled by the experimenter. The problem of design optimization is then to search that design space and identify the design that has the greatest potential to successfully discriminate among the models under consideration.

But how does one measure the potential of each design? Bayesian decision theory offers a principled approach to this problem. In Bayesian design optimization, each potential design is treated as a gamble whose payoff is determined by the outcome of an experiment carried out with that design. The idea is to estimate the "utilities" of hypothetical experiments carried out with a given design, so that an expected utility of that design can be computed. This is done by considering every possible observation that could be obtained from an experiment with a given design and then evaluating the relative likelihoods and statistical values of these observations. The design with the highest expected utility (i.e., the one that yields the most informative data, on average) is then chosen as the optimal design.

For example, in a retention experiment, the set of retention intervals at which to test memory is a design variable that could be optimized. In the example above, we considered two different sets of eight retention intervals, but many other sets are possible. To calculate the expected utility of a given set of retention intervals, simulated experiments must be conducted on computer (i.e., Bernoulli trials with the probabilities specified by a hypothesized generating model and parameters). The utility of those simulated data depend on how conclusively they identify the generating model and parameters (according to some model selection statistic such as the Bayes factor). The expected utility is obtained by repeating this process across numerous iterations with different generating models and parameters, and taking an average of resulting utilities. Thus, the expected utility measures how conclusively the data are expected to identify the generating model, whatever that model may be.

In quantitative terms, the utility function to be optimized, denoted by $U(d)$ as a function of design d, is expressed in the following form (e.g., Chaloner and Verdinelli, 1995):

$$U(d) = \sum_m p(m) \int \int u(d, \theta_m, y_m)\, p(y_m|\theta_m, d)\, p(\theta_m)\, dy_m\, d\theta_m. \qquad (1.22)$$

In the above equation, y_m denotes the data outcome under a hypothesized generating model m in a simulated experiment, θ_m denotes the model parameter, $p(y_m|\theta_m, d)$ is the model's probability distribution (often called the likelihood function), $p(m)$ is the model's prior probability, and finally, $p(\theta_m)$ is the parameter's prior distribution. The function, $u(d, \theta_m, y_m)$, measures the "local" utility of design d given the parameter value θ_m and the data outcome y_m. Note that "global" utility function $U(d)$ is defined as an average of the local utility $u(d, \theta_m, y_m)$ over the models under consideration, model

parameters, and data outcomes, with respect to the model prior $p(m)$, the parameter prior $p(\theta_m)$, and the likelihood function $p(y_m|\theta_m, d)$, respectively.

The choice of the local utility function used is derived from the specific goal of the experiment, whether the goal is parameter estimation, i.e., the estimation of a model's parameters, or alternatively, whether the goal is model discrimination, i.e., the identification of the underlying data-generating model among a set of competing models. Just to give an example, for parameter estimation, one may consider $u(d, \theta_m, y_m) = \log \frac{p(\theta_m|y_m,d)}{p(\theta_m)}$, which has an information theoretic interpretation (see, e.g., Myung et al., 2013, p. 58).

The design optimization problem entails identifying an optimal design d^* that maximizes the utility function $U(d)$ in Eq. (1.22). In practice, however, solving the problem presents computational challenges that make standard solution methods impractical or impossible. First, evaluating the expected utility of a given design entails high-dimensional numerical integration (over possible models, parameters, and observed data patterns), which requires Monte Carlo simulation. Moreover, spaces are often high-dimensional, requiring an intelligent search algorithm to ensure convergence to the global optimum. For example, if the design space to be searched consisted of all possible sets of 8 retention intervals, even a very sparse grid search would entail evaluating millions, or perhaps billions of expected utilities.

Recent breakthroughs in stochastic optimization have made this problem tractable (Müller et al., 2004; Amzal et al., 2006), but they are beyond the scope of this chapter. Alternatively, the problem of searching a high-dimensional design space can be mitigated by turning to a related method called *Adaptive Design Optimization* (ADO). In ADO, which we discuss in the following section, the full optimization problem is broken into pieces that can be evaluated sequentially, as the experiment progresses.

1.6.3 Adaptive Design Optimization

ADO is an adaptive search algorithm that combines design optimization with real-time Bayesian updating of parameter estimates and model probabilities. The idea of *Adaptive Design Optimization* (ADO) is to treat the full experiment as a sequence of mini-experiments, with the design of the mini-experiments optimized on the fly as the experiment progresses. *Adaptive* in Adaptive Design Optimization refers to the fact that the design of each mini-experiment is adapted based on the results of the preceding mini-experiments. By using all available information about the models and how the participant has responded, ADO collects data intelligently, making it well suited for evaluating computational models.

ADO has two main advantages over fixed (non-adaptive) DO. The first advantage stems from the fact that ADO optimizes the designs of the mini-experiments sequentially, as the experiment progresses, rather than optimizing them jointly before experimentation begins. This reduces the dimensionality of the search space, thereby greatly reducing the overall computational load. For example, consider a retention experiment in which memory is to be tested at eight different retention intervals. This experiment could be partitioned into a sequence of eight mini-experiments in which memory is to be tested at one retention interval. Finding an optimal design for the entire experiment would entail searching an eight-dimensional Euclidean space, whereas finding an optimal design for each mini-experiment would only entail searching a 1-dimensional Euclidean space.

The second advantage pertains to flexibility and data quality. The adaptive nature of ADO, by construction, controls for individual differences and thus makes it well suited for studying the most common and often largest source of variance in experiments. When participants are tested individually using ADO, the algorithm adjusts (i.e., optimizes) the design of the experiment to the performance of that participant, thereby maximizing the informativeness of the data at an individual participant level. Response strategies and group differences can be readily identified.

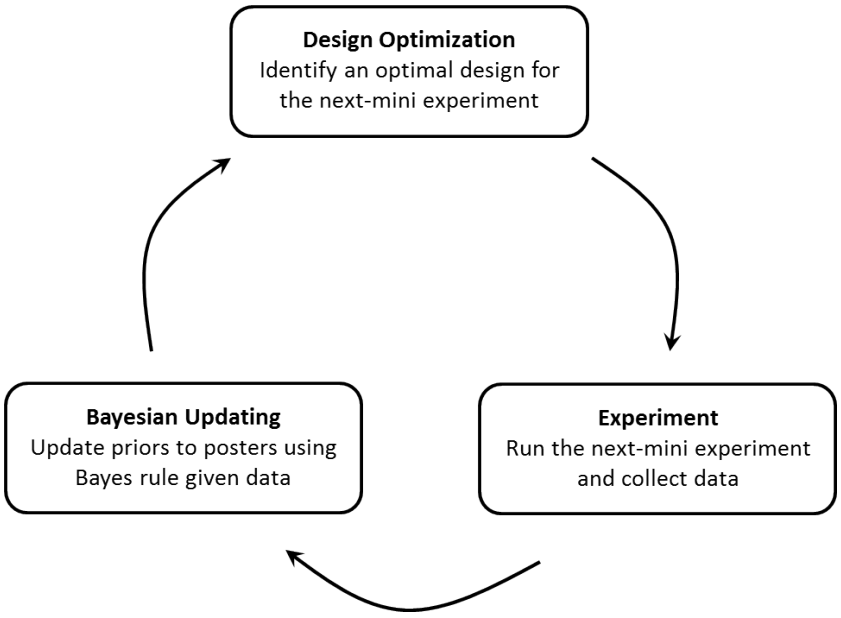

Figure 1.10 Schematic diagram illustrating the three-step procedure of Adaptive Design Optimization (ADO).

Specifically, an ADO experiment cycles through three basic steps: (1) design optimization (DO); (2) experiment; and (3) Bayesian updating. The relationship between these steps is depicted in Figure 1.10. The process begins with a design optimization step, in which the optimal design for the first mini-experiment is sought (top box in Figure 1.10). This step amounts to solving Eq. (1.22) with prior information about the models, i.e., the parameter prior $p(\theta_m)$ and the model prior $p(m)$. Once an optimal design d^* is identified, the first mini-experiment is carried out with that design (experiment step, lower-right in Figure 1.10). After data have been collected in the mini-experiment, they are used to update the parameter and model priors of each model using Bayes rule to the corresponding posteriors (lower-left in Figure 1.10). Model evaluation and comparison statistics like the MLE and BIC of each model can also be computed in this step. The updated parameter estimates and model probabilities then become the priors for the next design optimization step, in which the design for the second mini-experiment is identified, and the full cycle repeats. This adaptive and sequential process continues until the all of the mini-experiments have been completed.

1.6.4 Illustrative Example

In this section we provide an example application of the adaptive design optimization (ADO) methodology using a simulated experiment. We illustrate its application to demonstrate that ADO can successfully identify not only the data-generating model underlying simulated data between two competing models, but also the true parameter values of the model.

Recent findings from developmental studies on how children represent numbers such as the location of integers on number lines and amount of money, suggested that their numerical estimates can be highly inaccurate and warped (e.g., Opfer and Siegler, 2007). To give an example, children often perceive the difference between number 100 and number 1 as being greater than the difference between 1000 and 901, thereby indicating a compressed, logarithm-like scale representation, instead of the correct, linear scale representation.

Suppose that you as a developmental psychologist wish to identify the exact form of a child's numerical representation, in particular, whether the representation is linear or logarithmic, with a number-line experimental task. In the task, the child is presented on computer with an integer between 1 and 1000 and asked to indicate how large that number is by placing a vertical hatch with a mouse on a horizontal number line labelled with 1 on the left most end and 1000 on the right most end (Opfer and Siegler, 2007). The

two competing models to be discriminated in this experiment are defined as

$$\begin{aligned} LIN: &\ y = ax + b + e \\ LOG: &\ y = a\ln(x) + b + e, \end{aligned} \tag{1.23}$$

where x is the stimulus value between 0 and 1000, y is the observed response given x, and e is a normal random error with zero mean and standard deviation s. So each model has three parameters, $\theta = (a, b, s)$.

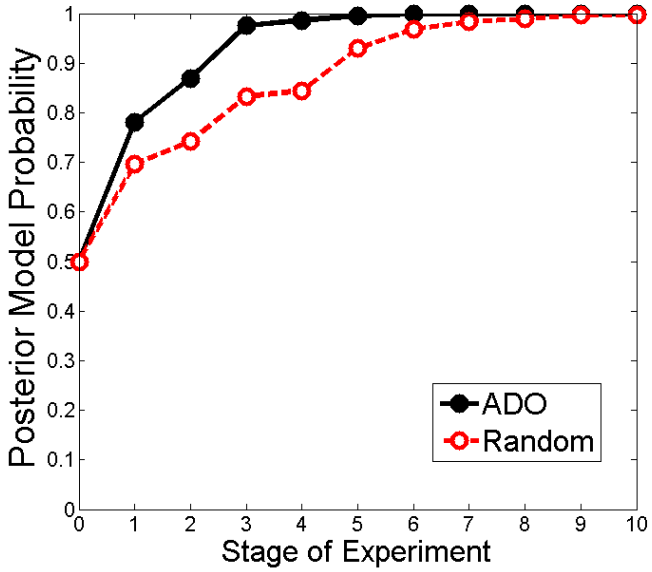

Figure 1.11 Posterior model probability curves of model LIN as a function of stages of the experiment under ADO (solid filled) and random (open broken) conditions. Each curve represents the mean of ten independent replications. The simulated data were generated from model LIN.

To illustrate how ADO works, we conducted a simulated number-line experiment using ADO to select the optimal deign (value on the number line), i.e., $d = x$, on each stage of a mini-experiment. The optimal design is the one that "best" discriminates the two models, LIN and LOG in some defined sense. Simulated responses were generated from model LIN with its parameter values of $\theta = (1.5, -0.5, 0.1)$. The three-step procedure of ADO shown in Figure 1.10 was repeated for ten stages of the experiment, each stage consisting of one trial of the experiment. In seeking an optimal design d^* that maximizes the global utility function $U(d)$ in Eq. (1.22), we employed a local utility function $u(d, \theta_m, y_m)$ defined as the ratio of two marginal like-

lihoods in Eq. (1.20) of the two competing models, which is known as the Bayes factor.

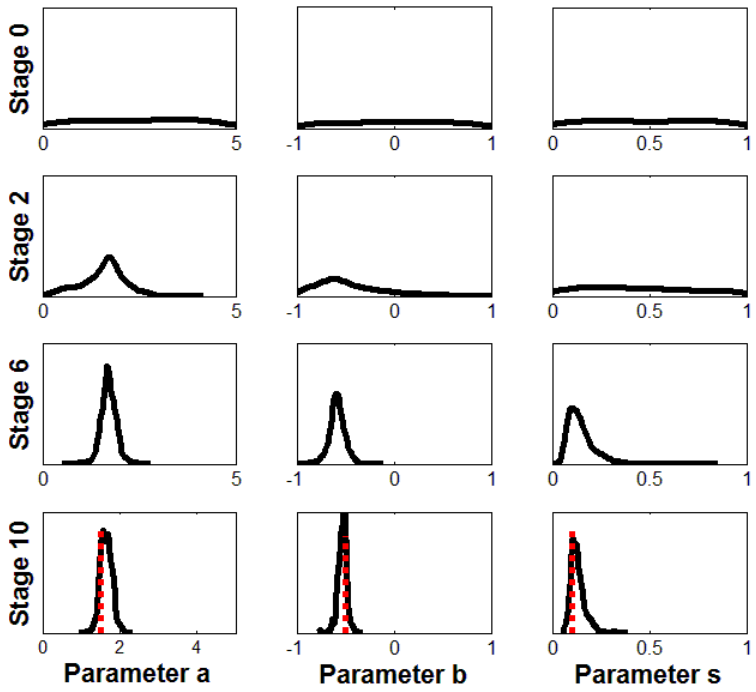

Figure 1.12 Posterior distributions of the LIN model's parameters, shown for four selected stages of experiment. The distributions are approximated using kernel smoothing densities with the normal kernel function. The vertical dotted red (lighter) lines in the bottom row indicate the parameter values with which simulated responses were generated by model LIN, i.e., $\theta = (a, b, s) = (1.5, -0.5, 0.1)$.

A summary of the results from the ADO simulation is presented in Figure 1.11. As shown in this figure, ADO clearly outperformed non-ADO, random selection in which the stimulus value was selected randomly and uniformly between 0 and 1000 on each stage, independent of either the observed response or the parameter and model priors. Note that ADO needed just three stages to identify the data-generating model (LIN) with over 0.95 probability, whereas non-ADO random selection required six stages, twice as many, to reach the same level of performance. Figure 1.12 is another summary of the simulation that shows the posterior distributions of parameters of the data-generating model at four selected stages under the ADO condition. Shown in the top row are uniform distributions (priors) used for

all three parameters at stage 0. Note that as the stage of experiment progresses, all three posterior distributions (priors) becomes more peaked and narrower around the true parameter values. In short, these results clearly demonstrate the superior efficiency of the ADO procedure in identifying the underlying data-generating model as well as its parameter values compared to non-ADO design selection.

For additional application examples of ADO in cognitive psychology, including ADO-based experiments with human participants, the reader is referred to Cavagnaro et al. (2010), Zhang and Lee (2010), Cavagnaro et al. (2011), Cavagnaro et al. (2013a), and Cavagnaro et al. (2013b).

1.6.5 Limitations

DO and ADO can increase the efficiency of data collection for the purposes of evaluating and comparing mathematical models, but it is important to be aware of the limitations and assumptions of the methodologies. First, not all design variables in an experiment can be optimized computationally. The variables must be quantifiable in such a way that the likelihood function depends explicitly on the values of the design variables being optimized. Consequently, neither DO nor ADO is applicable to nominal variables (e.g., task modality: words vs. pictures).

Another important limitation is the assumption that one of the models under consideration is the true, data-generating model. This assumption, obviously, is likely to be violated in practice, given that our models are merely approximations of the cognitive process under study. Ideally, one would like to optimize an experiment for an infinite array of models representing a whole spectrum of realities. However, no implementable methodology currently exists that can handle a problem of this scope.

1.7 General Discussion

Cognitive models are useful to the extent that we understand how they work. The model evaluation tools described in this chapter are meant to provide this understanding, elucidating the performance characteristics of a model and how it performs relative to other models. The pairing of optimal experimental design with post-experiment model selection methods form a potent combination of tools that can maximize inference about the structure and function of the cognitive process under study. Having devoted the preceding pages to a description of these tools, we close this chapter by placing them

in the context of broader issues in model evaluation and comparison so that readers can understand the strengths and weaknesses of the approach.

It should always be remembered that we have presented a purely quantitative approach to model evaluation. Qualitative criteria, such as model plausibility or interpretability, are not considered, yet they are extremely important to assess a model, so much so that they should be satisfied to a reasonable degree before applying quantitative criteria. If the assumptions underlying the formulation of a model are not sensible or are incompatible with one another, there is no reason to pursue it further, regardless of how well it might perform on quantitative measures.

The majority of quantitative criteria involve a consideration of a model's fit to the data. As fundamental as it may seem, it has its drawbacks. On the one hand, data are our only link to the cognitive process being studied, so the model must describe the data to some level of accuracy. As discussed above (section 1.4.3), model complexity must be considered to properly interpret the quality of a fit. However, the perils of overfitting can be mitigated by relaxing the objective and making goodness-of-fit (GOF) a more qualitative-like criterion, relying instead on a more subjective evaluation as to whether model behavior resembles human behavior.

Most behavioral data do not exhibit complex patterns, which is one reason why there can be such a crowded field of competing models. Too many functions can fit simple data. Of primary importance is whether the model captures the main pattern in the data, whether it be a linear trend of some sort (e.g., Figure 1.5) or an ordinal relation across nominal conditions. If the model mimics human performance sufficiently well, when is it overkill to split hairs regarding whether a competing model absorbs slightly more or less variance in the data? At some point, data fitting can become an unproductive and distracting exercise that no longer advances knowledge.

We are not suggesting GOF be abandoned, but rather that it be used thoughtfully. For example, it is an easy means of weeding out models that fail even to describe the salient trends in the data. It is equally important to recognize when it is ineffective in discriminating among the remaining contenders, especially if repeated experiments will likely fail to identify a clear winner. In fields where there are many contenders (e.g., decision making), a saturation point can be reached at which the models are no longer discriminable, at least given current technology. Because behavioral data are never noise-free, a point will be reached at which competitors mimic each other closely, and are thus similarly good. When confronted with this situation, it could be more productive to compare models on other criteria instead of continuing to design (even optimal) experiments with the hope of discrim-

inating them. In short, avoid obsessing over GOF. It is a mistake to focus on any single criterion.

A situation in which model evaluation tools can be particularly valuable is in deciding whether and how to expand a model. This is a thorny problem. A mathematical model is usually introduced in a very narrow topic area to explain a particular phenomenon (e.g., recognition memory). To the extent that the model proves valuable, a natural next step is to expand its scope and extend it to related phenomena (e.g., recall). There is little guidance on how to do this, so it is left to the researcher's ingenuity. Model evaluation tools can be instrumental in informing the development process. In particular, they can aid in justifying how the model is revised. What are the consequences of adding more parameters, which can be necessary when additional psychological constructs must be incorporated into the model? How should this new parameter be added, as a multiplicative or exponential term? Extensive simulations are necessary to answer these question. Even then, it can be extremely difficult to decide on the proper decision because of the many constraints that must be satisfied. It is not just that the new model must fit a new type of data or a broader range of data, but that these modifications must also provide explanatory value. That is, they must provide new insight into the operation of the cognitive process. For example, if a common memory mechanism is postulated to be responsible for recognition and recall, its operation in both tasks must be described at a level of detail to understand how the new model differs from and is an improvement over its predecessor. Without this additional yet crucial step, the process of model expansion turns into an exercise in statistical modeling (i.e., data fitting), not cognitive modeling.

A more common and similar endeavor is model revision, whereby an existing model is altered in light of new data that show its performance to be inferior to competing models in some way, usually a poorer fit. The same issues of justifying the change on theoretical grounds as well as on performance apply here. Although revision is a natural part of the scientific process, an over-emphasis on quantitative performance such as GOF can eventually yield models that are indistinguishable. They mimic each other closely because they have, over time, all been tuned to fit the same collection of data generated from years of experiments. The plus side of this scenario is that the models are together converging on the underlying cognitive model of interest, it is just being expressed in different forms. Although perhaps wishful thinking, some creative inquiry could identify an overarching (superordinate) model that includes each model as a special case. Short

of this, the field is left with competitors that could well prove very difficult to distinguish given that they all grew to be so similar.

A few final words are in order about the evidence justifying the modification or expansion of a model. Model development generally proceeds from the simple to the complex. The higher the bar for justifying the more complex model, the more likely simple models (explanations) will prevail. In this regard, it is unclear how wise it is to abide by Occams razor, because it is not always clear whether one has truly multiplied entities beyond necessity. A considerable amount of new data collection, not just a single data-fitting exercise or model simulation, is required to determine this.

The bias to favor simple models might also reflect a limitation of cognitive modeling itself. As a model is expanded to account for ever more phenomena, its behavior can become intractable. When too many parameters combine in complex ways, an understanding of behavior becomes elusive, even if the model mimics human performance impressively well (*Bonini's paradox*). It is for this reason that many of the tools described in this chapter are most productively applied to relatively simple models (< 8 parameters). The exception is cross-validation, which is probably the simplest and most versatile model selection tool.

In conclusion, science is driven more by technological advances than theoretical ones, even though technology works in the service of theory. The model evaluation tools discussed in this chapter provide a means of advancing model development. They are not full-proof, nor are they alone sufficient, but when used with other criteria they can assist the researcher in making informed decisions about model design and model choice.

1.8 Acknowledgments

This research is supported in part by National Institute of Health Grant R01-MH093838 to JIM and MAP. The sections of this chapter on model evaluation and model selection draw upon the work of Myung et al. (2009).

1.9 Appendix: Matlab Code

```
%++++++++++++++++++++++++++++++++++++++
% ModelSelection.m
% Main Program
% MATLAB Code for Model Selection Simulation
% Author: Jay Myung, Ohio State University (July 2014)
% Distribution: Public & Unlimited
```

```
%
%--- Initialization
clear;
global n;
opts=optimset('DerivativeCheck','off','Display','off','TolX',...
  1e-7,'TolFun',1e-7,'Diagnostics','off','MaxIter',500,...
  'LargeScale','on');

n=50;% sample size, i.e., number of binomial trials
t=[0.5 1 2 4 8 12 16 18];t=t';% time intervals
ycnt=[44 34 27 26 19 17 20 11];% observed correct responses
y=ycnt/n;y=y';
x=n*y;

%--- MLE, AIC & BIC
[am1,loglik1]=fmincon('power_mle',rand(2,1),[],[],[],[],...
   zeros(2,1),1*ones(2,1),[],opts,t,x);
[am2,loglik2]=fmincon('power2_mle',rand(3,1),[],[],[],[],...
   zeros(3,1),1*ones(3,1),[],opts,t,x);
yprd1=am1(1,1)*(t+1).^(-am1(2,1));
r2(1,1)=1-sum((yprd1-y).^2)/sum((y-mean(y)).^2);
yprd2=am2(1,1)*(t+1).^(-am2(2,1))+am2(3,1);
r2(2,1)=1-sum((yprd2-y).^2)/sum((y-mean(y)).^2);
%
pinit=[rand(5,1)];
pinit=[.906 .836 .263 -.094 .769]';% initial seed values for MLE
plower=[0 0 0 -.3 0];
pupper=[1 1 1 1 10]';
[am3,loglik3]=fmincon('power3_mle',pinit,[],[],[],[],plower,...
    pupper,[],opts,t,x);
yprd3=am3(1,1)*(t+1).^(-am3(2,1))+am3(3,1)+am3(4,1)*sin(am3(5,1)*t);
r2(3,1)=1-sum((yprd3-y).^2)/sum((y-mean(y)).^2);

[am4,loglik4]=fmincon('expo_mle',rand(2,1),[],[],[],[],...
     zeros(2,1),1*ones(2,1),[],opts,t,x);
yprd4=am4(1,1)*exp(-am4(2,1)*t);
r2(4,1)=1-sum((yprd4-y).^2)/sum((y-mean(y)).^2);

%-----------
logml=[loglik1 loglik2 loglik3 loglik4]';logml=(-1)*logml;
```

```
   % Log ML
aic=[2*loglik1+2*2 2*loglik2+2*3 2*loglik3+2*5 2*loglik4+2*2]';
   % AIC
bic=[2*loglik1+2*log(n) 2*loglik2+3*log(n) 2*loglik3+5*log(n) ...
    2*loglik4+2*log(n)]'; % BIC

disp('--R2  LogML  AIC    BIC ----------');
disp(num2str([r2 logml aic bic], '% 10.3f'));
disp('-- MLE estimates ----------');
disp(num2str([am1'],'% 10.3f'));
disp(num2str([am2'],'% 10.3f'));
disp(num2str([am3'],'% 10.3f'));
disp(num2str([am4'],'% 10.3f'));

%--- Plot the results
tt=(0.1:.1:20)';
ypow=am1(1,1)*(tt+1).^(-am1(2,1));
ypow2=am2(1,1)*(tt+1).^(-am2(2,1))+am2(3,1);
ypow3=am3(1,1)*(tt+1).^(-am3(2,1))+am3(3,1)+am3(4,1)*sin(am3(5,1)*tt);
yexp=am4(1,1)*exp(-am4(2,1)*tt);

clf;
plot(tt,ypow,'k-',tt,yexp,'b--',tt,ypow2,'r-',tt,ypow3,'g-',...
     'LineWidth',3);hold on;grid on;
xlim([0 20]);ylim([0 1]);xlabel('Time t', 'FontSize', 20);
ylabel('Probability p','FontSize', 20);
plot(t,y,'ko','MarkerFaceColor','k','MarkerSize',10);

%--- LOOCV
bm1=am1;bm2=am2;bm3=am3;bm4=am4;

tcv=zeros(7,1);xcv=zeros(7,1);
loocv=zeros(8,4);
for jj=1:8
if jj==1; tcv=t(2:8,:);xcv=x(2:8,:);
elseif jj==8;tcv=t(1:7,:);xcv=x(1:7,:);
else tcv=[t(1:jj-1,:);t(jj+1:8,:)];xcv=[x(1:jj-1,:);x(jj+1:8,:)];
end;

[am1]=fmincon('power_mle',bm1,[],[],[],[],zeros(2,1),...
```

```
     1*ones(2,1),[],opts,tcv,xcv);
[am2]=fmincon('power2_mle',bm2,[],[],[],[],zeros(3,1),...
     1*ones(3,1),[],opts,tcv,xcv);
pinit=[.906 .836 .263 -.094 .769]';% PVAF 98.8
plower=[0 0 0 -.3 0];
pupper=[1 1 1 1 10]';
[am3]=fmincon('power3_mle',bm3,[],[],[],[],plower,pupper,...
      [],opts,tcv,xcv);
[am4]=fmincon('expo_mle',bm4,[],[],[],[],zeros(2,1),...
     1*ones(2,1),[],opts, tcv,xcv);

loglik1=power_mle(am1,t(jj,1),x(jj,1));
loglik2=power2_mle(am2,t(jj,1),x(jj,1));
loglik3=power3_mle(am3,t(jj,1),x(jj,1));
loglik4=expo_mle(am4,t(jj,1),x(jj,1));
loocv(jj,:)=[loglik1 loglik2 loglik3 loglik4];
end;% jj
disp('--- LOOCV ---------');
disp(num2str([mean(loocv)'], '% 10.3f'));

%--- APE
bm1=am1;bm2=am2;bm3=am3;bm4=am4;

apepow=zeros(5,1);apeexp=zeros(5,1);
for jj=1:5;
 tape=t(1:2+jj,:);xape=x(1:2+jj,:);
 [am1]=fmincon('power_mle',bm1,[],[],[],[],zeros(2,1),...
      1*ones(2,1),[],opts,tape,xape);
 [am4]=fmincon('expo_mle',bm4,[],[],[],[],zeros(2,1),...
      1*ones(2,1),[],opts,tape,xape);
loglik1=power_mle(am1,t(jj+3,1),x(jj+3,1));
loglik4=expo_mle(am4,t(jj+3,1),x(jj+3,1));
apepow(jj,1)=loglik1;
apeexp(jj,1)=loglik4;
end;% jj

apepow2=zeros(4,1);
for jj=1:4;
 tape=t(1:3+jj,:);xape=x(1:3+jj,:);
 [am2]=fmincon('power2_mle',bm2,[],[],[],[],zeros(3,1),...
```

```
      1*ones(3,1),[],opts,tape,xape);
loglik2=power2_mle(am2,t(jj+4,1),x(jj+4,1));
apepow2(jj,1)=loglik2;
end;% jj

apepow3=zeros(2,1);
for jj=1:2;
 tape=t(1:5+jj,:);xape=x(1:5+jj,:);
pinit=[.906 .836 .263 -.094 .769]';% PVAF 98.8
plower=[0 0 0 -.3 0];
pupper=[1 1 1 1 10]';
[am3]=fmincon('power3_mle',bm3,[],[],[],[],plower,pupper,...
      [],opts,tape,xape);
loglik3=power3_mle(am3,t(jj+6,1),x(jj+6,1));
apepow3(jj,1)=loglik3;
end;% jj

disp('--- APE ---------');
disp(num2str([mean(apepow) mean(apepow2) mean(apepow3) ...
     mean(apeexp)]','% 10.3f'));
%--- END of ModelSelection.m

%++++++++++++++++++++++++++++++++++++++
% Function-call Programs
%
function loglik = power_mle(a,t,x)
global n
   [mc,mr]=size(x);
   p=a(1,1)*(t+1).^(-a(2,1));
   p=(p < ones(mc,1)).*p+(p >= ones(mc,1))*.999999;
   loglik=(-1)*(x.*log(p)+(n-x).*log(1-p));
   loglik=sum(loglik);

function loglik = power2_mle(a,t,x)
global n
   [mc,mr]=size(x);
   p=a(1,1)*(t+1).^(-a(2,1))+a(3,1);
   p=(p < ones(mc,1)).*p+(p >= ones(mc,1))*.999999;
   loglik=(-1)*(x.*log(p)+(n-x).*log(1-p));
   loglik=sum(loglik);
```

```
function loglik = power3_mle(a,t,x)
global n
   [mc,mr]=size(x);
   p=a(1,1)*(t+1).^(-a(2,1))+a(3,1)+a(4,1)*sin(a(5,1)*t);
   p=(p < ones(mc,1)).*p+(p >= ones(mc,1))*.999999;
   loglik=(-1)*(x.*log(p)+(n-x).*log(1-p));
   loglik=sum(loglik);

function loglik = expo_mle(a,t,x)
global n
   [mc,mr]=size(x);
   p=a(1,1)*exp(-a(2,1)*t);
   p=(p < ones(mc,1)).*p+(p >= ones(mc,1))*.999999;
   loglik=(-1)*(x.*log(p)+(n-x).*log(1-p));
   loglik=sum(loglik);
```

References

Akaike, H. 1973. Information theory and an extension of the maximum likelihood principle. Pages 267–281 of: Petrov, B. N., and Caski, F. (eds), *Proceedings of the Second International Symposium on Information Theory*. Budapest: Akademiai Kiado.

Amzal, B., Bois, F. Y., Parent, E., and Robert, C. P. 2006. Bayesian-Optimal Design via Interacting Particle Systems. *Journal of the American Statistical Association*, **101**(474), 773–785.

Atkinson, A.C., and Donev, A.N. 1992. *Optimum Experimental Designs*. Oxford University Press.

Atkinson, A.C., and Federov, V.V. 1975. Optimal design: Experiments for discriminating between several models. *Biometrika*, **62**(2), 289.

Balasubramanian, V. 1997. Statistical inference, Occam's razor and statistical mechanics on the space of probability distributions. *Neural Computation*, **9**, 349–368.

Bamber, D., and van Santen, J.P.H. 1985. How many parameters can a model have and still be testable. *Journal of Mathematical Psychology*, **29**, 443–473.

Bamber, D., and van Santen, J.P.H. 2000. How to assess a model's testability and identifiability. *Journal of Mathematical Psychology*, **44**, 20–40.

Browne, M. W. 2000. Cross-validation methods. *Journal of Mathematical Psychology*, **44**, 108–132.

Busemeyer, J. R., and Diederich, A. 2010. *Cognitive Modeling*. Thousand Oaks, CA: Sage Publications.

Casella, G., and Berger, R. L. 2002. *Statistical Inference (2nd edition)*. Duxbury.

Cavagnaro, D. R., Myung, J. I., Pitt, M. A., and Kujala, J. V. 2010. Adaptive design optimization: A mutual information based approach to model discrimination in cognitive science. *Neural Computation*, **22**(4), 887–905.

Cavagnaro, D. R., Pitt, M. A., and Myung, J. I. 2011. Model discrimination through adaptive experimentation. *Psychonomic Bulletin & Review*, **18**(1), 204–210.

Cavagnaro, D. R., Pitt, M. A., Gonzalez, R., and Myung, J. I. 2013a. Discriminating among probability weighting functions using adaptive design optimization. *Journal of Risk and Uncertainty*, **47**, 255–289.

Cavagnaro, D. R., Gonzalez, R., Myung, J. I., and Pitt, M. A. 2013b. Optimal decision stimuli for risky choice experiments: An adaptive approach. *Management Science*, **59**(2), 358–375.

Chaloner, K., and Verdinelli, I. 1995. Bayesian experimental design: A review. *Statistical Science*, **10**(3), 273–304.

Cover, T. M., and Thomas, J. A. 1991. *Elements of Information Theory*. Hoboken, New Jersey: John Wiley & Sons, Inc.

Dawid, A. P. 1984. Statistical theory: The prequential approach. *Journal of the Royal Statistical Society, Series A*, **147**, 278–292.

Fum, D., Del Missier, F., and Stocco, A. 2007. The cognitive modeling of human behavior: Why a model is (sometimes) better than 10,1000 words. *Cognitive Systems Research*, **8**, 135–142.

Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., and Rubin, D. B. 2013. *Bayesian Data Analysis (3rd edition)*. Boca Raton, Florida: Chapman & Hall/CRC.

Grünwald, P. D. 2005. A tutorial introduction to the minimum description length principle. In: Grünwald, P., Myung, I. J., and Pitt, M. A. (eds), *Advances in Minimum Description Length: Theory and Applications*. The M.I.T. Press.

Grünwald, P. D. 2007. *The Minimum Description Length Principle*. M.I.T. Press.

Grünwald, P. D., Myung, I. J., and Pitt, M. A. (eds). 2005. *Advances in Minimum Description Length: Theory and Applications*. M.I.T. Press.

Kass, R. E., and Raftery, A. E. 1995. Bayes Factors. *Journal of the American Statistical Association*, **90**, 773–795.

Kiefer, J. 1959. Optimum experimental designs. *Journal of the Royal Statistical Society: Statistical Methodology B*, **21**, 272–319.

Klauer, K. C., and Kellen, D. 2011. The flexibility of models of recognition memory: An analysis by the minimum-description length principle. *Journal of Mathematical Psychology*, **55**, 430–450.

Lee, M. D., and Wagenmakers, E-J. 2014. *Bayesian Cognitive Modeling: A Practical Course*. Cambridge, U.K.: Cambridge University Press.

Lewandowsky, S., and Farrell, S. 2011. *Computational Modeling in Cognition: Principles and Practice*. Thousand Oaks, CA: Sage Publications.

Linhart, H., and Zucchini, W. 1986. *Model Selection*. New York, NY: John Wiley & Sons.

McClelland, J. L. 2009. The place of modeling in cognitive science. *Topics in Cognitive Science*, **1**, 11–38.

Müller, P., Sanso, B., and De Iorio, M. 2004. Optimal Bayesian design by inhomogeneous Markov chain simulation. *Journal of the American Statistical Association*, **99**(467), 788–798.

Myung, I. J. 2003. Tutorial on maximum likelihood estimation. *Journal of Mathematical Psychology*, **47**, 90–100.

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

Myung, I. J., Brunsman, A. E., and Pitt, M. A. 1999. True to thyself: Assessing whether computational models of cognition remain faithful to their theoretical principles. Pages 462–467 of: Hahn, M., and Stoness, S.C. (eds), *Proceedings of the 21st Annual Conference of the Cognitive Science Society*. Mahwha, N.J.: Lawrence Erlbaum Associates.

Myung, I. J., Forster, M., and Browne, M. W. 2000. Special issue on model selection. *Journal of Mathematical Psychology*, **44**, 1–2.

Myung, I. J., Pitt, M. A., and Kim, W. 2005. Model evaluation, testing and selection. Pages 422–436 of: Lambert, K., and Goldstone, R. (eds), *The Handbook of Cognition*. Sage Publications.

Myung, I. J., Navarro, D. J., and Pitt, M. A. 2006. Model selection by normalized maximum likelihood. *Journal of Mathematical Psychology*, **50**, 167–179.

Myung, J. I., and Pitt, M. A. 2004. Model comparison methods. *Methods in Enzymology*, **383**, 351–366.

Myung, J. I., and Pitt, M. A. 2009. Optimal experimental design for model discrimination. *Psychological Review*, **58**, 499–518.

Myung, J. I., Tang, Y., and Pitt, M. A. 2009. Evaluation and comparison of computational models. *Methods in Enzymology*, **454**, 287–304.

Myung, J. I., Cavagnaro, D. R., and Pitt, M. A. 2013. A tutorial on adaptive design optimization. *Journal of Mathematical Psychology*, **57**, 53–67.

Navarro, D. J., Pitt, M. A., and Myung, I. J. 2004. Assessing the distinguishability of models and the informativeness of data. *Cognitive Psychology*, **49**, 47–84.

Opfer, J., and Siegler, R. 2007. Representational change and chidren's numerical estimation. *Cognitive Psychology*, **55**, 165–195.

Pitt, M. A., and Myung, I. J. 2002. When a good fit can be bad. *Trends in Cognitive Sciences*, **6**(10), 421–425.

Pitt, M. A., Myung, I. J., and Zhang, S. 2002. Toward a method of selecting among computational models of cognition. *Psychological Review*, **190**(3), 472–491.

Polk, T. A., and Seifert, C. M. (eds). 2002. *Cognitive Modeling.* Cambridge, MA: MIT Press.

Rissanen, J. 1996. Fisher information and stochastic complexity. *IEEE Transactions on Information Theory*, **42**, 40–70.

Rissanen, J. 2001. Strong optimality of the normalized ML models as universal code and information in data. *IEEE Transactions on Information Theory*, **42**, 1712–1717.

Roberts, S., and Pashler, H. 2000. How pursuasive is a good fit? A comment on theory testing. *Psychological Review*, **107**, 358–367.

Rubin, D.C., and Wenzel, A.E. 1996. One Hundred Years of Forgetting: A Quantitative Description of Retention. *Psychological Review*, **103**(4), 734–760.

Schervish, M. J. 1995. *The Theory of Statistics.* New York, NY: Springer-Verlag.

Schwarz, G. 1978. Estiamting the dimension of a model. *Annals of Statistics*, **6**, 461–464.

Shiffrin, R. M., and Nobel, P. A. 1997. The art of model development and testing. *Behavior Research Methods, Instruments, & Computers*, **29**, 6–14.

Shiffrin, R. M., Lee, M. D., Kim, W., and Wagenmakers, E-J. 2008. A survey of model evaluation approaches with a tutorial on hierarchical Bayesian methods. *Cognitive Science*, **32**, 1248–1284.

Singman, H., and Kellen, D. 2013. MPTinR: analysis of multinomial processing tree models in R. *Behavioral Research Methods*, **45**, 560–575.

Stone, M. 1974. Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society, Series B*, **36**, 111–147.

Su, Y., Myung, I. J., Pitt, M. A., and Kim, W. 2005. Minimum description length and cognitive modeling. Pages 411–433 of: Grünwald, P. D., Myung, I. J., and Pitt, M. A. (eds), *Advances in Minimum Description Length: Theory and Applications.* MIT press.

Wagenmakers, E-J., and Waldorp, L. 2006. Editors' introduction. *Journal of Mathematical Psychology*, **50**, 99–100.

Wagenmakers, E-J., Grünwald, P. D., and Steyvers, M. 2006. Accumulative prediction error and the selection of time series models. *Journal of Mathematical Psychology*, **50**, 149–166.

Wasserman, L. 2000. Bayesian model selection and model averaging. *Journal of Mathematical Psychology*, **44**, 92–107.

Wixted, J.T., and Ebbesen, E.B. 1991. On the Form of Forgetting. *Psychological Science*, **2**(6), 409–415.

Wu, H., Myung, J. I., and Batchelder, W. H. 2010. Minimum description length model selection of multinomial processing tree models. *Psychonomic Bulletin & Review*, **17**, 276–286.

Zhang, S., and Lee, M. D. 2010. Optimal experimental design for a class of bandit problems. *Journal of Mathematical Psychology*, **54**, 499–508.

Index