Title

Hierarchical Bayesian inference in the brain: psychological models and neural implementation

Author

Shi, Lei

Publication Date

2009

Hierarchical Bayesian inference in the brain: Psychological models and neural implementation

by

Lei Shi

B.ENG. (Tongji University, Shanghai, China) 2001
M.S. in Communications Engineering (Technical University of Munich, Germany) 2003
M.S. in Biomathematics (Technical University of Munich, Germany) 2004
M.A. in Statistics (University of California, Berkeley) 2008

A dissertation submitted in partial satisfaction
of the requirements for the degree of

Doctor of Philosophy

in

Neuroscience

in the

GRADUATE DIVISION

of the

UNIVERSITY OF CALIFORNIA, BERKELEY

Committee in charge:

Professor Thomas Griffiths, Chair
Professor Bruno Olshausen
Professor Jose Carmena
Professor Gerald Westheimer

Fall 2009

The dissertation of Lei Shi is approved:

Professor Thomas Griffiths, Chair Date

Professor Bruno Olshausen Date

Professor Jose Carmena Date

Professor Gerald Westheimer Date

University of California, Berkeley

Fall 2009

Hierarchical Bayesian inference in the brain:
Psychological models and neural implementation

by

Lei Shi

Abstract

Hierarchical Bayesian inference in the brain:
Psychological models and neural implementation
by
Lei Shi
Doctor of Philosophy in Neuroscience
University of California, Berkeley
Professor Thomas Griffiths, Chair

The human brain effortlessly solves problems that still pose a challenge for modern computers, such as recognizing patterns in natural images. Many of these problems can be formulated in terms of Bayesian inference, including planning motor movements, combining cues from different modalities, and making predictions. Recent work in psychology and neuroscience suggests that human behavior is often consistent with Bayesian inference. However, most research using probabilistic models has focused on formulating the abstract problems behind cognitive tasks and their optimal solutions, rather than considering mechanisms that could implement these solutions. Therefore, it is critical to understand the psychological models and neural implementations that carry out these notoriously challenging computations.

Exemplar models are a successful class of psychological process models that use an inventory of stored examples to solve problems such as identification, categorization, and function learning. We show that exemplar models can be used to perform a sophisticated form of Monte Carlo approximation known as importance sampling, and thus provide a way to perform approximate Bayesian inference. Simulations of Bayesian inference in speech perception, generalization along a single dimension,

making predictions about everyday events, concept learning, and reconstruction from memory show that exemplar models can often account for human performance with only a few exemplars, for both simple and relatively complex prior distributions. These results suggest that exemplar models provide a possible mechanism for implementing at least some forms of Bayesian inference.

The goal of perception is to infer the hidden states in the hierarchical process by which sensory data are generated, a problem that can be solved optimally using Bayesian inference. Here we propose a simple mechanism for Bayesian inference which involves averaging over a few feature detection neurons which fire at a rate determined by their similarity to a sensory stimulus. This mechanism is again based on importance sampling. Moreover, many cognitive and perceptual tasks involve multiple levels of abstraction, which results in "hierarchical" models. We show that a simple extension to recursive importance sampling can be used to perform hierarchical Bayesian inference. We identify a scheme for implementing importance sampling with spiking neurons, and show that this scheme can account for human behavior in sensorimotor integration, cue combination, and orientation perception.

Another important function of nervous system is to process temporal information in the dynamical environment, such as motion coordination where the system's state is estimated sequentially based on the constant perceptual feedback. Our study suggests that a neural network structure similar to recursive importance sampling can solve the sequential estimation problem by approximating the posterior updates. This algorithm performs as well as the state-of-the-art sequential Monte Carlo methods know as particle filtering and fulfills many constraints of the biological system. Studying the detailed neural implementation of this algorithm finds an interesting resemblance to neural circuits in cerebellum.

To my parents,

Yan Sun & Kegang Shi

Acknowledgements

I owe my deepest gratitude to my thesis advisor, Prof. Thomas Griffiths, whose encouragement, guidance and support lead me into this fascinating research field and all the way to the dissertation. It has been my honor!

This thesis would not have been possible without the guidance of other thesis committee members, Prof. Gerald Westheimer, Prof. Bruno Olshausen and Prof. Jose Carmena. I have learned tremendously from their broad expertise. Their support is one of the most important driving forces keep me on track of the adventure. I would like to thank my collaborators Naomi Feldman and Adam Sanborn for their contributions to Chapter 3. I am grateful to Mrs. Kati Markowitz, our graduate program manager, and Prof. John Ngai, our graduate program head, for making Neuroscience an excellent graduate program for interdisciplinary studies. Special thank to Prof. Yang Dan for introducing me into electrophysiology experiment and Prof. Lu Chen for guiding me through the molecular neurobiology side project for my qualification exam. It is a pleasure to thank those who made this thesis possible, including everybody in the Computational Cognitive Science Lab and Helen Wills Neuroscience Institute.

My parents are the most enthusiastic supporters behind every decision I made. My gratitude to them is beyond the description of any words. I am also indebted to all my friends, who make the stay in San Francisco Bay Area an absolutely wonderful experience.

Lastly, I offer my regards and blessings to all of those who supported me in any respect during the doctoral study at Berkeley.

Lei Shi
Berkeley, CA 2009

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 The brain as an inference machine

Much of cognition and perception involves inference under uncertainty, using limited data from the world to evaluate under determined hypotheses. Probabilistic models provide a way to characterize rational solutions to these problems, with probability distributions encoding the beliefs of agents and Bayesian inference updating those distributions as data become available. As a consequence, probabilistic models are becoming increasingly widespread in both cognitive science and neuroscience, providing explanations of behavior in domains as diverse as motor control [Körding and Wolpert, 2004], reasoning [Oaksford and Chater, 1994], memory [Anderson and Milson, 1989], and perception [Yuille and Kersten, 2006]. However, these explanations are typically presented at Marr's [1982] computational level, focusing on the abstract problem being solved and the logic of that solution. Unlike many other formal approaches to cognition, probabilistic models are usually not intended to provide an account of the mechanisms underlying behavior – how people actually produce responses consistent with optimal statistical inference.

Table 1.1: Marr's three-level analysis in understanding Bayesian inference in the brain

Computational level	Bayesian inference
Algorithmic level	*Psychological process models*
Mechanistic level	*Neural implementations*

Understanding the mechanisms that could support Bayesian inference is particularly important since probabilistic computations can be extremely challenging. Representing and updating distributions over large numbers of hypotheses is computationally expensive, a fact that is often viewed as a limitation of rational models (e.g.,[Kahneman and Tversky, 1972; Gigerenzer and Todd, 1999]). The question of how people could perform Bayesian inference can be answered at at least two levels (as suggested by Marr, [1982], see Table 1.1). One kind of answer is at the level of psychological processes – showing that the Bayesian inference can be performed using mechanisms that are no more complex than those used in psychological process models. The language of such answers is representations, similarity, activation, and so forth, and some preliminary work has been done in this direction [Kruschke, 2006; Sanborn *et al.*, 2006]. A second kind of answer focuses on the neural level, exploring ways in which systems of neurons could perform probabilistic computations. The language of such answers is that of neurons, tuning curves, firing rates, and so forth, and several recent papers have explored ways in which systems of neurons could perform probabilistic computations (e.g., [Ma *et al.*, 2006; Zemel *et al.*, 1998]).

1.2 Psychological models for Bayesian inference

The focus of the first part of the thesis is on a familiar class of psychological process models known as exemplar models. These models assume that people store many instances ("exemplars") of events in memory, and evaluate new events by activating stored exemplars that are similar to those events [Medin and Schaffer, 1978; Nosofsky, 1986]. It is well known that exemplar models of categorization can be analyzed in terms of nonparametric density estimation, and implement a Bayesian solution to this problem [Ashby and Alfonso-Reese, 1995]. Here we show that exemplar models can be used to solve problems of Bayesian inference more generally, providing a way to approximate expectations of functions over posterior distributions. Our key result is that exemplar models can be interpreted as a sophisticated form of Monte Carlo approximation known as *importance sampling.* This result illustrates how at least some cases of Bayesian inference can be performed using a simple mechanism that is a common part of psychological process models.

Our analysis of Bayesian inference using exemplar models is also an instance of a more general strategy for exploring possible psychological mechanisms for implementing rational models. Importance sampling is one of a variety of methods used for approximating probabilistic computations in computer science and statistics. These methods are used because they provide efficient approximate solutions to problems that might be intractable to solve exactly. If we extend the principle of optimality underlying rational models of cognition to incorporate constraints on processing, we might expect to see similarities between the approximation schemes used by computer scientists and statisticians and the mechanisms by which probabilistic computations are implemented in the human mind. In some cases, as for importance sampling and exemplar models, the resulting "rational process models" provide a way to connect the abstract level of analysis used in many probabilistic models of cognition with existing ideas about psychological processes.

Establishing a stronger connection between rational models of cognition and psychological mechanisms has been a goal of cognitive scientists at least since Simon [1957] introduced the notion of "bounded rationality." Several different strategies for taking into account the effects of information-processing constraints have been considered, including incorporating those constraints into the optimization process involved in rational analysis (e.g.,[Anderson, 1990]), handicapping rational models to produce behavior closer to that of human participants (e.g.,[Steyvers *et al.*, 2003]), and rejecting the principle of optimization altogether in favor of finding simple but effective heuristics (e.g.,[Gigerenzer and Todd, 1999]). The idea of developing rational process models shares characteristics with all of these strategies, with its focus being on finding psychologically plausible processes that can be justified as approximations to rational statistical inference. Such processes ideally generalize beyond the solutions to specific optimization problems, or schemes for handicapping specific models, and provide a new way to look at the mechanistic or heuristic accounts that psychologists have developed in order to explain aspects of human behavior.

1.3 Neural implementation of Bayesian inference

Living creatures occupy an environment full of uncertainty due to noisy sensory inputs, incomplete information, and unobserved variables. One of the goals of the nervous system is to infer the states of the world given these limited data and make decisions accordingly. This task involves combining prior knowledge with current data [Körding and Wolpert, 2004], and integrating cues from multiple sensory modalities [Ernst and Banks, 2002]. Studies of human psychophysics and animal behavior suggest that the brain is capable of solving these problems in a way that is consistent with optimal Bayesian statistical inference [Körding and Wolpert, 2004; Ernst and Banks, 2002; Stocker and Simoncelli, 2008; Blaisdell *et al.*, 2006]. Moreover, complex brain functions such as visual information processing involve multiple brain areas [Van Essen *et al.*, 1992], and require making inferences at multiple levels of abstraction. Hierarchical Bayesian inference has been proposed as a computational framework for modeling such processes [Lee and Mumford, 2003]. Identifying neural mechanisms that could support hierarchical Bayesian inference is important, since probabilistic computations can be extremely challenging. Just representing and updating distributions over large numbers of hypotheses is computationally expensive.

The Bayesian perspective on cortical computation has received much attention in the computational neuroscience community in recent years [Doya, 2007]. Much effort has recently been devoted towards proposing possible mechanisms for implementing Bayesian inference based on known neuronal properties. One prominent approach to explaining how the brain uses the activation of a population of neurons for probabilistic computations employs a "Bayesian decoding" framework [Zemel *et al.*, 1998]. In this framework, it is assumed that the firing rate of a population of neurons, r, can be converted to a probability distribution over stimuli, $p(s|r)$, by applying Bayesian inference, where the likelihood $p(r|s)$ reflects the probability of that firing pattern given the stimulus s. A firing pattern thus encodes a distribution over stimuli, which can be recovered through Bayesian decoding. The problem of performing probabilistic computations then reduces to identifying a set of operations on firing rates r that result in probabilistically correct operations on the resulting distributions $p(s|r)$. For example, [Ma *et al.*, 2006] showed that when the likelihood $p(r|s)$ is an exponential family distribution with linear sufficient statistics, adding two sets of firing rates is equivalent to multiplying probability distributions.

In this work, we take a different approach, allowing a population of neurons to encode a probability distribution directly. Each neuron acts as a feature detector whose expected firing rate is proportional to the probability that the feature is presented. This approach simplifies coding-decoding operations and provides a straightforward solution to the problem of hierarchical Bayesian inference. Since the composition of a neural population and the pattern of spikes produced by neurons both introduce elements of stochasticity, the result is a Monte Carlo approximation, capturing the properties of a probability distribution through a set of samples from that distribution. This perspective also provides a natural way to understand the variability observed in neural responses [Hoyer and Hyvärinen, 2002]. These topics are discussed in detail in Chapter 4.

The brain is constantly making sequential estimates based on real-time sensory inputs. For example, in reaching for a tennis ball, visual information about the position and velocity of the hand and ball is fed back to coordinate the muscle movement. This is a probabilistic computation due to the internal noise of the nervous system. To achieve optimal solutions, the brain needs to maintain a posterior density and keep updating upon receiving new sensory inputs. This problem can be formulated as sequential Bayesian inferences. Particle filtering is a Monte Carlo method that provides an approximated general solution to this problem. However, its neural implementation is not straightforward because of biological constraints. In Chapter 5, we study the neural implementation of sequential Bayesian inference based on importance sampling and propose the cerebellum as the neural substrate to execute the computation.

This thesis is organized in the following way. Chapter 2 lays out the theoretical foundations of Bayesian inference, Monte Carlo methods and importance sampling. Chapter 3 shows that exemplar models are equivalent to a special case of importance sampling and, therefore, perform approximate Bayesian inference. Chapter 4 looks at the neural implementation of importance sampling and extends the solution to hierarchical Bayesian inference. Chapter 5 study how nervous system perform sequential Bayesian inference and the neural substrates for such computation. Chapter 6 concludes the thesis.

Chapter 2

Theoretical Background

Bayes' rule connects prior knowledge and observations in assessing the probability of certain hypotheses given observations. Using Bayes' rule, beliefs of hypotheses can be updated upon new observations, a process known as Bayesian inference. Exact inference is often hard to obtain because it often requires integration of irregular functions or sums over high dimensional spaces. In this chapter, we first introduce Bayesian inference and then discuss algorithms that approximate exact inference.

2.1 Bayesian inference

Many cognitive problems can be formulated as evaluating a set of hypotheses about processes that could have produced observed data. For example, perceiving speech sounds requires considering what sounds might be consistent with an auditory stimulus [Feldman *et al.*, 2009], generalizing a property from one object to another involves considering the set of objects likely to possess that property [Shepard, 1987], predicting the duration of an ongoing event necessitates reasoning from its current duration to a hypothetical future endpoint [Griffiths and Tenenbaum, 2007], and learning a concept from examples means evaluating a space of possible concepts [Tenenbaum and Griffiths, 2001]. Even reconstructing information from memory can be analyzed as an inference about the nature of that information from the data provided by a noisy memory trace [Huttenlocher *et al.*, 2000].

Bayesian inference provides a solution to problems of this kind. Letting h denote a hypothesis and d the data, assume a learner encodes his or her degrees of belief regarding the hypotheses before seeing d using a probability distribution, $p(h)$, known as the *prior* distribution. Then, the degrees of belief after seeing d are given by the *posterior* distribution, $p(h|d)$, obtained from Bayes' rule

$$p(h|d) = \frac{p(d|h)p(h)}{\int_{\mathcal{H}} p(d|h)p(h)\,dh}, \tag{2.1}$$

where $\mathcal{H}$ is the set of hypotheses under consideration (the *hypothesis space*), and $p(d|h)$ is a distribution indicating the probability of seeing d if h were true, known as the *likelihood.*

While our analysis applies to Bayesian inference in the general case, we introduce it using a specific example that is consistent with several of the psychological tasks we consider later in Chapter 3. We return to the general case after working through this specific example. Assume we observe a stimulus x, which we believe to be corrupted by noise and potentially missing associated information, such as a category label. Let x^* denote the uncorrupted stimulus, and z denote the missing data. Often, our goal is simply to reconstruct x, finding the x^* to which it corresponds. In this case, z can be empty. Otherwise, we seek to infer both x^* and the value of z which corresponds to x. We can perform both tasks using Bayesian inference.

The application of Bayes' rule is easier to illustrate in the case where z is empty, where we simply wish to infer the true stimulus x^* from noisy x. We use the probability distribution $p(x|x^*)$ to characterize the noise process, indicating the probability with which the stimulus x^* is corrupted to x, and the probability distribution $p(x^*)$

to encode our a priori beliefs about the probability of seeing a given stimulus. We can then use Bayes' rule to compute the posterior distribution over the value of the uncorrupted stimulus, x^*, which might have generated the observation x, obtaining

$$p(x^*|x) = \frac{p(x|x^*)p(x^*)}{\int p(x|x^*)p(x^*)\, dx^*}, \tag{2.2}$$

where $p(x|x^*)$ is the likelihood and $p(x^*)$ is the prior.

This analysis is straightforward to generalize to the case where z contains missing data, such as the label of the category from which x was generated. In this case, we need to define our prior as a distribution over both x^* and z, $p(x^*, z)$. We can then use Bayes' rule to compute the posterior distribution over the uncorrupted stimulus, x^*, and missing data, z, which might have generated the observation x, obtaining

$$p(x^*, z|x) = \frac{p(x|x^*)p(x^*, z)}{\int\int p(x|x^*)p(x^*, z)\, dx^*\, dz}, \tag{2.3}$$

where we also assume that the probability of observing x is independent of z given x^*, so $p(x|x^*, z) = p(x|x^*)$.

2.2 Evaluating expectations by Monte Carlo

Posterior distributions on hypotheses given data can be used to answer a variety of questions. To return to the example above, a posterior distribution on x^* and z can be used to evaluate the properties of x^* and z given x. A standard way to do this is to use the expectation of a function over the posterior distribution. For any function $f(x^*, z)$, the posterior *expectation* of that function given x is

$$E\left[f(x^*, z)|x\right] = \int\int f(x^*, z)p(x^*, z|x)\, dx^*\, dz, \tag{2.4}$$

being the average of $f(x^*, z)$ over the posterior distribution. Since $f(x^*, z)$ can pick out any property of x^* and z that might be of interest, many problems of reasoning under uncertainty can be expressed in terms of expectations. For example, we could compute the posterior mean of x^* by taking $f(x^*, z) = x^*$, or calculate the posterior probability that z takes a particular value by taking $f(x^*, z)$ to be 1 when z has that value, and 0 otherwise.

Evaluating expectations over the posterior distribution can be challenging: it requires computing a posterior distribution, which is a hard problem in itself, because the integrals in Eq. 2.4 can range over many values for x^* and z. Consequently, Monte Carlo methods are often used to approximate expectations. Monte Carlo methods approximate the expectation of a function with respect to a probability distribution with the average of that function at points drawn from the distribution. Assume we want to evaluate the expectation of a function $g(y)$ over the distribution $p(y)$, $E_p\left[g(y)\right]$ (where we use y as a generic random variable, instead of x^* and z). Let μ denote the value of this expectation. The law of large numbers justifies

$$\mu = E_p\left[g(y)\right] = \int g(y)p(y)\,dy \approx \frac{1}{m}\sum_{j=1}^{m} g(y_j), \qquad (2.5)$$

where the y_j are all drawn from the distribution $p(y)$.

This simple Monte Carlo method requires that we are able to generate samples from the distribution $p(y)$. However, this is often not the case: it is quite common to encounter problems where $p(y)$ is known at all points y but hard to sample from. If a *surrogate distribution* $q(y)$ is close to $p(y)$ but easy to sample from, a form of Monte Carlo called *importance sampling* can be applied (see [Neal, 1993] for a detailed introduction, and [Robert and Casella, 1999] for a mathematical treatment).

Manipulating the expression for the expectation of g gives

$$\int g(y)p(y)\,dy = \frac{\int g(y)p(y)\;dy}{\int p(y)\,dy} = \frac{\int g(y)\frac{p(y)}{q(y)}q(y)\,dy}{\int \frac{p(y)}{q(y)}q(y)\,dy}. \tag{2.6}$$

The numerator and denominator of this expression are each expectations with respect to $q(y)$. Applying simple Monte Carlo (with the same set of samples from $q(y)$) to both,

$$\mu = E_p\left[g(y)\right] \approx \frac{\sum_{j=1}^{m} g(y_j)\frac{p(y_j)}{q(y_j)}}{\sum_{j=1}^{m} \frac{p(y_j)}{q(y_j)}}, \tag{2.7}$$

where each y_j is drawn from $q(y)$. The ratios $\frac{p(y_j)}{q(y_j)}$ are "importance weights" on the samples y_j, correcting for having sampled from $q(y)$ rather than $p(y)$. Intuitively, these weights capture how important each sampled value should be to calculating the expectation, and give importance sampling its name. If the y_j are sampled directly from $p(y)$, they are given equal weight, each having an importance weight of 1. However, when the y_j are sampled from surrogate distribution $q(y)$, they bear nonuniform importance weights due to the difference between $p(y)$ and $q(y)$. Samples with higher probability under $p(y)$ than $q(y)$ occur less often than they would if we were sampling from $p(y)$, but receive greater weight, counter-balancing the lower sampling frequency, with the opposite applying to samples with higher probability under $q(y)$ than $p(y)$.

Importance sampling is a useful method for approximating expectations when simple Monte Carlo cannot be applied because generating samples from the target distribution is difficult. However, using an importance sampler can make sense even in cases where simple Monte Carlo can also be applied. First, it allows a single set of samples to be used to evaluate expectations with respect to a range of distributions, through the use of different weights for each distribution. Second, the estimate of μ produced by the importance sampler can have lower variance than the estimate produced by simple Monte Carlo, if the surrogate distribution is chosen to place high probability on values of y where both $p(y)$ and $g(y)$ are large. [1]

Both simple Monte Carlo and importance sampling can be applied to the problem of evaluating the expectation of a function $f(x^*, z)$ over a posterior distribution on x^* and z with which we began this section. Simple Monte Carlo would draw values of x^* and z from the posterior distribution $p(x^*, z|x)$ directly. Importance sampling would generate from surrogate distribution, $q(x^*, z)$, and then re-weight those samples. One simple choice of $q(x^*, z)$ is the prior, $p(x^*, z)$. If we sample from the prior, the weight assigned to each sample is the ratio of the posterior to the prior

$$\frac{p(x^*, z|x)}{p(x^*, z)} = \frac{p(x|x^*)}{\int\int p(x|x^*)p(x^*, z)\, dx^*\, dz}, \tag{2.9}$$

where we use the assumption that $p(x|x^*, z) = p(x|x^*)$. Substituting these weights into Eq. 2.7 and canceling constants, we obtain

$$E\left[f(x^*, z)|x\right] \approx \frac{\sum_{j=1}^{m} f(x_j^*, z_j)p(x|x_j^*)}{\sum_{j=1}^{m} p(x|x_j^*)}, \tag{2.10}$$

where we assume that x_j^* and z_j are drawn from $p(x^*, z)$. Because the weights on the samples are based on the likelihood, this approach is sometimes known as *likelihood weighting.*

Fig. 2.1 provides a visual illustration of the approximation of Bayesian inference using importance sampling. Here, the goal is to recover the true value of a noisy observation x, which is done by computing the posterior expectation $E[x^*|x]$. This can be

[1] If the function $g(y)$ takes on its largest values in regions where $p(y)$ is small, the variance of the simple Monte Carlo estimate can be large. An importance sampler can have lower variance than simple Monte Carlo if $q(y)$ is chosen to be complementary to $g(y)$. In particular, the asymptotic variance of the sampler is minimized by specifying $q(y)$ as

$$q(y) \propto |g(y) - E_p[g(y)]|\, p(y). \tag{2.8}$$

This is not a practical procedure, since finding this distribution requires computing $E_p[g(y)]$, but the fact that the minimum variance sampler need not be $p(y)$ means that importance sampling can provide a better estimate of an expectation than simple Monte Carlo.

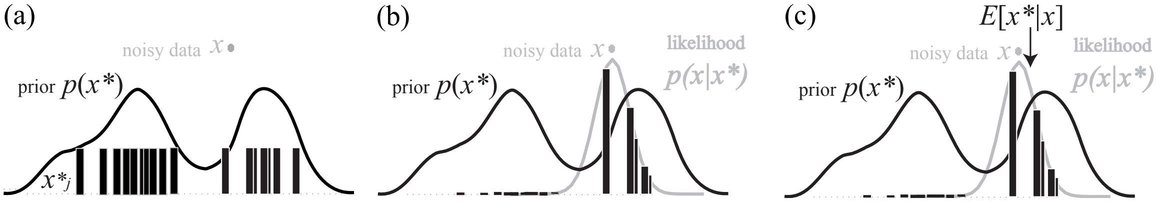


Figure 2.1: Approximating Bayesian inference by importance sampling using the prior $p(x^*)$ as the surrogate distribution. The true value of a stimulus x^* is recovered from a noisy observation x (represented by the gray dot). (a) Exemplars x^*_j are sampled from the prior $p(x^*)$. (b) The x^*_j are weighted by a Gaussian likelihood function $p(x|x^*_j)$. Weights decrease quickly as exemplars move away from x. (c) The expectation is the weighted average of the x^*_j. Compared with x, the estimate $E[x^*|x]$ is shifted towards a region that has higher probability under the prior.

done applying Eq. 2.10 with $f(x^*, z) = x^*$. First, exemplars x^*_j are drawn from prior distribution $p(x^*)$ (Fig. 2.1a). Then, these exemplars are given weights proportional to the likelihood $p(x|x^*)$ (Fig. 2.1b). Finally, $E[x^*|x]$ is estimated by the weighted sum $\sum_j x^*_j p(x|x^*_j)$ normalized by $\sum_j p(x|x^*_j)$. The posterior expectation moves the estimate of x^* closer to the nearest mode of the prior distribution (Fig. 2.1c), appropriately combining prior knowledge with the noisy observation. This computation is straightforward despite the complicated shape of the prior distribution.

The success of this importance sampling scheme for approximating posterior expectations depends on how much probability mass the prior and posterior distribution share. This can be understood by considering how the variance of the importance weights depends on the relationship between the surrogate and target distributions. The variance of the importance weights determines the stability of the estimate produced by importance sampling: If only a few samples have high weights, then the estimate of the expectation is based only on those samples. Fig. 2.2 provides some intuitions for this phenomenon. If the prior largely overlaps with the posterior, as in Fig. 2.2a, the importance weights have little variance and the estimate produced by the sampler is fairly stable. If the prior does not overlap with the posterior, as in

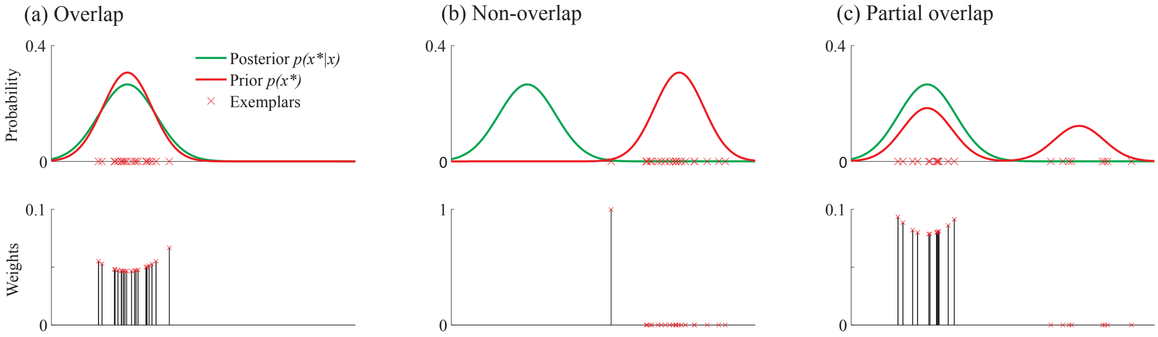


Figure 2.2: The variance of the importance weights in approximating posterior expectations depends on how much probability mass is shared between prior and posterior. Different patterns are observed if posterior and prior distributions are (a) strongly overlapping, (b) non-overlapping or (c) partially overlapping. In these figures, the importance weights have been normalized to make it clear what proportion of the expectation depends on each sample. Greater overlap between prior and posterior results in lower variance in the importance weights, use of a larger set of samples, and consequently a better approximation.

Fig. 2.2b, few samples from the prior fall in the region with higher posterior probability, and these samples are given all the weight. The estimate is then solely dependent on these samples and is highly unstable. In intermediate cases, such as that shown in Fig. 2.2c where the prior is a multi-modal distribution and the posterior is one of the modes, stable results are obtained if enough samples are drawn from each of the modes. In cases where there is not a close match between prior and posterior, a reasonably large number of samples needs to be drawn from the prior to ensure a good approximation.

Chapter 3

Exemplar models as a mechanism for performing Bayesian inference

The previous chapter provides the mathematical formulation of Bayesian inference, and how it can be approximated, focusing on Monte Carlo methods. In this chapter, we first introduce a class of psychological process models known as exemplar models and show its connection to importance sampling. Then we explore the capacity of exemplar models to perform Bayesian inference in various tasks. These include a range of cognitive tasks from perception, generalization, prediction and concept learning. We also use simulations of performance on these tasks to investigate the effects of different kinds of capacity limitations and ongoing storage of exemplars in memory.

3.1 Exemplar models

Human knowledge is formed by observing examples. When we learned the concept "dog," we were not taught to remember the physiological and anatomical characteristics of dogs, but instead, saw examples of various dogs. Based on the large inventory of examples of dogs we have seen, we are able to reason about the properties of dogs, and make decisions about whether new objects we encounter are likely to be dogs. Exemplar models provide a simple explanation for how we do this, suggesting that we do not form abstract generalizations from experience, but rather store examples in memory and use those stored examples as the basis for future judgments (e.g.,[Medin and Schaffer, 1978; Nosofsky, 1986]).

An exemplar model consists of stored exemplars $X^* = \{x_1^*, x_2^*, \cdots, x_n^*\}$, and a similarity function $s(x, x^*)$, measuring how closely a new observation x is related to x^*.[1] On observing x, all exemplars are activated in proportion to $s(x, x^*)$. The use of the exemplars depends on the task [Nosofsky, 1986]. In an identification task, where the goal is to identify the x^* of which x is an instance, the probability of selecting x_i^* is

$$p_r(x_i^*|x) = \frac{s(x, x_i^*)}{\sum_{j=1}^n s(x, x_j^*)}, \tag{3.1}$$

where $p_r(\cdot)$ denotes the response distribution resulting from the exemplar model, and we assume that participants use the Luce-Shepard rule [Luce, 1959; Shepard, 1962] in selecting a response, with no biases towards particular exemplars. In a categorization task, where each exemplar x_j^* is associated with a category c_j, the probability that the new object x is assigned to category c is given by

$$p_r(c|x) = \frac{\sum_{j|c_j=c} s(x, x_j^*)}{\sum_{j=1}^n s(x, x_j^*)}, \tag{3.2}$$

where again we assume a Luce-Shepard rule without biases towards particular categories.

While exemplar models have been most prominent in the literature on categorization, the same basic principles have been used to define models of function learning

[1] Our analysis requires that this similarity measure has a finite integral, with $\int s(x, x^*)dx$ equal to a fixed constant for all x^*. This assumption is satisfied by similarity functions such as the exponential or Gaussian that are typically used in exemplar models.

[DeLosh *et al.*, 1997], probabilistic reasoning [Juslin and Persson, 2002], and social judgment [Smith and Zarate, 1992]. These models pursue a similar approach to models of categorization, but associate each exemplar with a quantity other than a category label. For example, in function learning each exemplar is associated with the value of a continuous variable rather than a discrete category index. The procedure for generating responses remains the same as that used in Eq. 3.1 and 3.2: the associated information is averaged over exemplars, weighted by their similarity to the stimulus. Thus, the predicted value of some associated information f for a new stimulus x is

$$\hat{f} = \frac{\sum_{j=1}^{n} f_j s(x, x_j^*)}{\sum_{j=1}^{n} s(x, x_j^*)}, \tag{3.3}$$

where f_j denotes the information associated with the jth exemplar. The identification and categorization models can be viewed as special cases, corresponding to different ways of specifying f_j. Taking $f_j = 1$ for $j = i$ and 0 otherwise yields Eq. 3.1, while taking $f_j = 1$ if $c_j = c$ and 0 otherwise yields Eq. 3.2. Eq. 3.3 thus provides the general formulation of an exemplar model that we will analyze.

3.2 Exemplar models as importance samplers

Inspection of Eq. 3.3 and 2.10 yields our main result: Exemplar models can be viewed as implementing a form of importance sampling. More formally, assume X^* is a set of m exemplars x^* and associated information z drawn from the probability distribution $p(x^*, z)$, and $f_j = f(x_j^*, z_j)$ for some function $f(x^*, z)$. Then the output of Eq. 3.3 for an exemplar model with exemplars X^* and similarity function $s(x, x^*)$ is an importance sampling approximation to the expectation of $f(x^*, z)$ over the posterior distribution on x^* and z, as given in Eq. 2.3, if two conditions are fulfilled: the x_j^* and z_j making up X^* are sampled from the prior $p(x^*, z)$ and the similarity function

$s(x, x^*)$ is proportional to the likelihood $p(x|x^*)$. Returning to Fig. 2.1, the x_i^* are now exemplars, and the importance weights reflect the amount of activation of those exemplars based on similarity to the observed data x.

The two conditions identified in the previous paragraph are crucial in establishing the connection between exemplar models and importance sampling. They are also reasonably natural assumptions, if we assume that exemplars are stored in memory as the result of experience, and that similarity functions are flexible and can vary from task to task. For most perceptual tasks of the kind we have been considering here, the prior $p(x^*, z)$ represents the distribution over the states of the environment that an agent lives in. Thus, sampling x_j^* and z_j from the prior is equivalent to storing randomly generated events in memory. The second condition states that the similarity between x and x^* corresponds to the likelihood function, subject to a ratio constant. This is straightforward when the stimulus x exists in the same space as x^*, as when x is a noisy observation of x^*. In this case, similarity functions are typically assumed to be monotonically decreasing functions in space, such as exponentials or Gaussians, which map naturally to likelihood functions [Nosofsky, 1986; Ashby and Alfonso-Reese, 1995].

This connection between exemplar models and importance sampling provides an alternative rational justification for exemplar models of categorization, as well as a more general motivation for these models. The justification for exemplar models in terms of nonparametric density estimation [Ashby and Alfonso-Reese, 1995] provides a clear account of their relevance to categorization, but does not explain why they are appropriate in other contexts, such as identification (Eq. 3.1) or the general response rule given in Eq. 3.3. In contrast, we can use importance sampling to provide a single explanation for many uses of exemplar models, such as categorization, identification and function learning, viewing each as the result of approximating an expectation of a particular function $f(x^*, z)$ over the posterior distribution $p(x^*, z|x)$. For categorization, z is the category label and the quantity of interest is $p(z = c|x)$, the posterior probability that x belongs to category c. Hence, $f(x^*, z) = 1$ for all $z = c$ and 0 otherwise. For identification, the question is whether the observed x corresponds to a specific x^*, so $f(x^*, z) = 1$ for that x^* and 0 otherwise, regardless of z. For function learning, z contains the value of the continuous variable associated with x^*, and $f(x^*, z) = z$. Similar analyses apply in other cases, with exemplar models providing a rational method for answering questions expressed as an expectation of a function of x^* and z.

3.3 A general scheme for approximating Bayes

The equivalence between exemplar models and importance sampling established in the previous section focuses on the specific problem of interpreting a noisy stimulus. However, the idea that importance sampling constitutes a psychologically plausible mechanism for approximating Bayesian inference generalizes beyond this specific problem. In the general case an agent seeks to evaluate a hypothesis h in light of data d, and does so by computing the posterior distribution $p(h|d)$ as specified by Eq. 2.1. An expectation of a function $f(h)$ over the posterior distribution can be approximated by sampling hypotheses from the prior, $p(h)$, and weighting the samples by the likelihood, $p(d|h)$. Formally, we have

$$E[f(h)|d] = \int f(h)p(h|d)\,dh \approx \frac{\sum_{j=1}^{m} f(h_j)p(d|h_j)}{\sum_{j=1}^{m} p(d|h_j)}, \tag{3.4}$$

where h_j is drawn from the prior $p(h)$.

Approximating Bayesian inference by importance sampling in this general case can also be interpreted as a kind of exemplar model, but here the stored “exemplars” correspond to hypotheses rather than stimuli. As in a standard exemplar model, these hypotheses can be stored in memory as the consequence of previous learning events. Each hypothesis needs to be weighted by its likelihood, which no longer has a natural interpretation in terms of similarity, but represents the degree to which a hypothesis is "activated" as a result of observing the data. Thus, all that is required for an agent to be able to approximate Bayesian inference in this way is to store hypotheses in memory as they are encountered, and to activate those hypotheses in such a way that the hypotheses that best account for the data receive the most activation.

The theoretical properties of importance sampling suggest that exemplar models of the kind considered in this and the preceding section may provide a way to approximate Bayesian inference in at least some cases. Specifically, we expect that importance sampling with a relatively small number of samples drawn from the prior should produce an accurate approximation to Bayesian inference in cases where prior and posterior share a reasonable amount of probability mass. This can occur in cases where the data are relatively uninformative, either as a result of small samples or high levels of noise. Despite this constraint, we anticipate that there will be a variety of applications in which exemplar models provide a good enough approximation to Bayesian inference to account for existing behavioral data.

In the remainder of the chapter we present a series of simulations evaluating exemplar models as a scheme for approximating Bayesian inference in five tasks. These tasks are selected to illustrate the breadth of this approach, and to allow us to explore the effect of number of exemplars on performance, as well as the consequences of other variants on the basic importance sampling scheme intended to reflect possible psychological or biological constraints. In general, we use the notation from the original papers in describing these simulations. However, in each case we formulate the underlying problem to be solved by Bayesian inference, and relate it back to either the specific or general problems of Bayesian inference we have considered in establishing the connection to exemplar models, identifying the correspondence between the relevant variables.

3.4 Simulation 1: The perceptual magnet effect

Categorical perception of speech sounds was first demonstrated by Liberman *et al.*[1957], who showed that listeners' discrimination of stop consonants was little better than would be predicted on the basis of categorization performance, with sharp discrimination peaks at category boundaries. Evidence has also been found in vowels for a *perceptual magnet effect*, a language-specific shrinking of perceptual space specifically near category prototypes, presumably due to a perceptual bias toward category centers [Kuhl *et al.*, 1992]. However, perception of vowels differs from that of stop consonants in that it is continuous rather than strictly categorical, with listeners showing high levels of within-category discrimination [Fry *et al.*, 1962]. Because of the high level of within-category discriminability in vowels, the perceptual magnet effect has been difficult to capture through traditional labeling accounts of categorical perception.

Feldman *et al.*[2009] argued that the perceptual magnet effect arises because listeners are trying to recover the phonetic detail (e.g., formant values) of a speaker's target production from a noisy speech signal. Under this account, listeners perform a Bayesian de-noising process, recovering the intended formant values of the noisy speech sounds they hear. Speech sounds are assumed to belong to phonetic categories in the native language, and listeners can use their knowledge of these categories to guide their inferences of the speaker's target production. Because this account assumes that listeners are trying to recover phonetic detail, it predicts a baseline level of within-category discrimination while still allowing categories to influence listeners' perception.

The Bayesian model introduced by Feldman *et al.*[2009] assumes that speakers, in producing a speech sound, sample a phonetic value for their target production T from a Gaussian phonetic category c with category mean μ_c and category variance σ_c^2. Listeners hear a speech sound S, which has been perturbed by articulatory, acoustic, and perceptual noise. This noisy speech sound S is normally distributed around the target production T with noise variance σ_S^2. The prior on target productions is therefore a mixture of Gaussians representing the phonetic categories of the language,

$$p(T) = \sum_c p(T|c)p(c) = \sum_c N(T|\mu_c, \sigma_c^2)p(c), \tag{3.5}$$

where $N(T|\mu_c, \sigma_c^2)$ is the probability density at T given a Gaussian distribution with mean μ_c and variance σ_c^2. The likelihood function represents the noise process that corrupts a target production T into a speech sound S, and is given by the Gaussian function representing speech signal noise,

$$p(S|T) = N(S|T, \sigma_S^2). \tag{3.6}$$

Listeners hear the speech sound S and use Bayes' rule to compute the posterior mean (ie. the expectation $E[T|S]$) and optimally recover the phonetic detail of a speaker's target production, marginalizing over all possible category labels.

The problem of inferring T from S is directly analogous to the problem of inferring a true stimulus x^* from a noisy stimulus x that we considered when introducing importance sampling. To complete the analogy, the category c corresponds to the missing information z, and the expectation $E[T|S]$ corresponds to $E[x^*|x]$. This expectation can thus be approximated by an importance sampler of the form given in Eq. 2.10, with $f(x^*, z) = x^*$. By the equivalence between importance sampling and exemplar models, this means that we can approximate the Bayesian solution to the problem of inferring T from S using an exemplar model.

An exemplar model derived through importance sampling provides a psychologically plausible implementation of the model introduced by Feldman et al. [2009], allowing listeners to optimally recover speakers' target productions using unlabeled exemplars. This implementation has two specific advantages over the original Bayesian formulation. First, there is evidence that infants as young as six months show a language-specific perceptual magnet effect even though they are still forming phonetic categories [Kuhl *et al.*, 1992], and importance sampling allows them to perform this computation without any explicit category knowledge. Category labels are not required, and the distribution of exemplars need not follow any parametric distribution. Second, importance sampling directly parallels the neural network model of the perceptual magnet effect proposed by Guenther and Gjaja[1996], allowing the Bayesian model and the neural network model to be interpreted as convergent descriptions of the same perceptual process.

To calculate the expected target production T using importance sampling, listeners need to store their percepts of previously encountered speech sounds, giving them a sample from $p(T)$, the prior on target productions (Eq. 3.5).[2] Upon hearing a new speech sound, they weight each stored exemplar by its likelihood $p(S|T)$ (Eq. 3.6) and take the weighted average of these exemplars to approximate the posterior mean as

$$E[T|S] \approx \frac{\sum_{j=1}^{m} T_j p(S|T_j)}{\sum_{j=1}^{m} p(S|T_j)}, \tag{3.7}$$

where T_j denotes the formant value of a stored target production.

We compared the performance of this exemplar model to multidimensional scaling data from Iverson and Kuhl[1995] on adult English speakers' discrimination of 13

[2] Because listeners only hear noisy speech sounds S, they may not have direct access to a sample from T. Storing samples from S instead of T produces the same qualitative effect, though the computation is no longer optimal. Alternatively, listeners may be able to bootstrap a sample from T by using multiple cues to reduce the amount of noise and by using subsequent percepts to update stored values. We return to the problem of recruiting exemplars during inference in Simulation 5.

equally-spaced stimuli in the /i/ and /e/ categories. The discrimination data were obtained through an AX task in which subjects heard pairs of stimuli and pressed a button to indicate whether the stimuli were identical. Responses and reaction times were used in a multidimensional scaling analysis to create a one-dimensional map of perceptual space, shown in Fig. 3.1. The data show a non-linear mapping between acoustic space and perceptual space, with portions that are more nearly horizontal corresponding to areas in which perceptual space is shrunk relative to acoustic space. Sounds near phonetic category centers are closer together in perceptual space than sounds near category boundaries, despite being separated by equal psychophysical distances. We simulated the performance of exemplar models with ten and fifty exemplars drawn from the prior, examining both the performance of individual simulated participants and the results of aggregating across participants. The results of this simulation, shown together with the multidimensional scaling data in Fig. 3.1, suggest that a relatively small number of exemplars suffices to capture human performance in this perceptual task. Model performance using ten exemplars already demonstrates the desired effect, and with fifty exemplars, the model gives a precise approximation that closely mirrors the combined performance of the 18 subjects in Iverson and Kuhl's multidimensional scaling experiment.

In addition to giving a simple psychological mechanism for approximating Bayesian inference in this task, importance sampling provides a link between the Bayesian model and a previous account of the perceptual magnet effect. The exemplar model considered in this section is isomorphic to a neural mechanism proposed by Guenther and Gjaja[1996] to create a bias toward category centers. In Guenther and Gjaja's neural map, the firing preferences of a population of neurons come to mirror the distribution of speech sounds in the input. Upon hearing a speech sound, listeners recover a percept of that speech sound by taking a weighted average of firing preferences in the neural map. The weights, or neural activations, are determined by

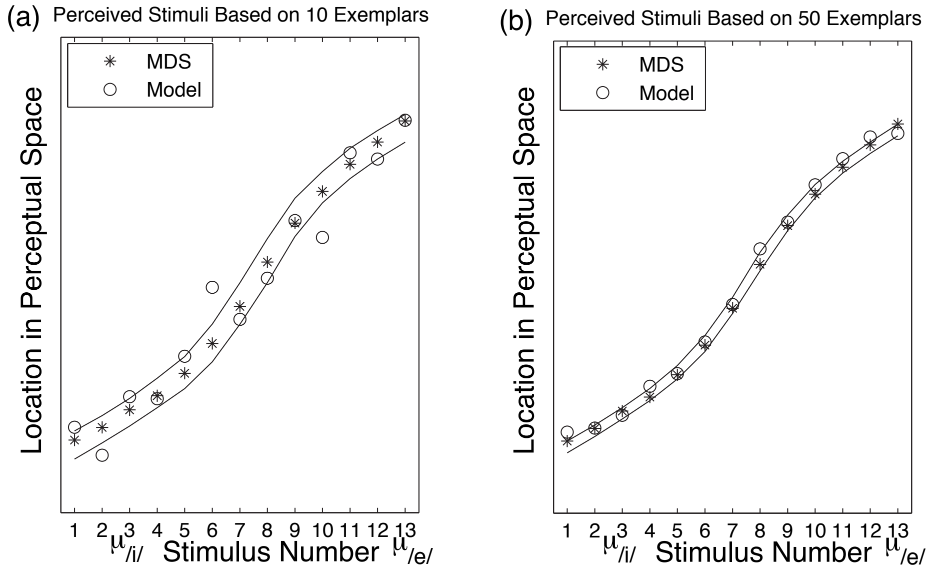


Figure 3.1: Locations of stimuli in perceptual space from Iverson and Kuhl's [1995] multidimensional scaling data and from a single hypothetical subject (open circles) and the middle 50% of hypothetical subjects (solid lines) using an exemplar model in which perception is based on (a) ten and (b) fifty exemplars. The labels $\mu_{/i/}$ and $\mu_{/e/}$ show the locations of category means in the model. Parameter values were those used by Feldman, Griffiths, and Morgan [2009].

the similarity between a neuron's firing preference and the speech sound heard. This perceptual mechanism implements an importance sampler: Firing preferences of individual neurons constitute samples from the prior, and the activation function plays the role of the likelihood. The activation function in the neural map differs from the Gaussian function assumed in the Bayesian model, but both implement the idea that exemplars with similar acoustic values should be weighted most highly. The correspondence between these two models suggests that Monte Carlo methods such as importance sampling may provide connections not just to psychological processes, but to the neural mechanisms that might support probabilistic computations. We return to this possibility in the General Discussion.

3.5 Simulation 2: The universal law of generalization

In a celebrated paper, Shepard [1987] showed that generalization gradients decrease exponentially with psychological distance across many experimental situations. He then gave a probabilistic explanation for this phenomenon that was later formulated in a Bayesian framework [Myung and Shepard, 1996; Tenenbaum and Griffiths, 2001]. Here, we use the notation originally introduced by Shepard. Assume that we observe a stimulus $\mathbf{0}$ that has a certain property (or "consequence"). What is the probability that a test stimulus $\mathbf{x}$ has the same property? Shepard analyzed this problem by assuming that $\mathbf{0}$ and $\mathbf{x}$ were points in a psychological space, and the set of stimuli sharing a property defined a consequential region in the space. We know that the original stimulus $\mathbf{0}$ belongs to this region, and we want to evaluate whether the test stimulus $\mathbf{x}$ does. We thus want to compute the probability that the $\mathbf{x}$ falls into an unknown consequential region containing $\mathbf{0}$.

The first question we can answer is which consequential regions $\mathbf{0}$ could have come from. This is a problem of Bayesian inference, where consequential regions are hypotheses and observing that $\mathbf{0}$ belongs to the region constitutes data. In the case of one-dimensional generalization, we might take consequential regions to be intervals along that dimension, parameterized by their center c and size s. We then want to compute the posterior distribution on intervals (c, s) given the information that $\mathbf{0} \in (c, s)$. This can be done by defining a prior $p(c, s)$ and likelihood $p(\mathbf{0}|c, s)$. Shepard [1987] assumed that all locations of consequential regions are equally probable, so the distribution of c is uniform and the prior distribution $p(c, s)$ is specified purely in terms of a distribution on sizes, $p(s)$. The likelihood is obtained by assuming that $\mathbf{0}$ is sampled uniformly at random from the interval given by (c, s), resulting in $p(\mathbf{0}|c, s) = 1/m(s)$ for all intervals (c, s) containing $\mathbf{0}$, where $m(s)$ is a measure of the volume of a region of size s (in one dimension, the length of the interval), and $p(\mathbf{0}|c, s) = 0$ for all other intervals. Prior and likelihood can then be combined as in Eq. 2.1 to yield a posterior distribution over consequential regions.

With a posterior distribution over consequential regions in hand, the probability that $\mathbf{x}$ belongs to one of the consequential regions containing $\mathbf{0}$ is obtained by summing the posterior probabilities of the regions containing $\mathbf{x}$. This can be expressed as the integral

$$p(\mathbf{x}|\mathbf{0}) = \int_{s,c} \mathbf{1}(\mathbf{x} \in (c, s)) p(c, s|\mathbf{0}) \; ds \; dc, \tag{3.8}$$

where $\mathbf{1}(\mathbf{x} \in (c, s))$ is an indicator function that equals 1 if $\mathbf{x}$ is in the region parameterized by (c, s) and 0 otherwise. This integral can also be viewed as an expectation of the indicator function $\mathbf{1}(\mathbf{x} \in (c, s))$ over the posterior distribution $p(c, s|\mathbf{0})$.

By viewing Eq. 3.8 as an expectation, it becomes clear that it can be approximated by importance sampling, and thus by an exemplar model. Identifying a consequential region does not match the form of the simple stimulus de-noising problem that we used in demonstrating equivalence between importance sampling and exemplar models, requiring us to use the more general idea that Bayesian inference can be approximated by storing hypotheses sampled from the prior and activating them based on consistency with data. In this case, the hypotheses h are consequential regions, the data d consist of the observation that $\mathbf{0}$ is contained in some consequential region, and the function $f(h)$ that we want the expectation of is the indicator function that takes the value 1 if $\mathbf{x}$ is in the consequential region and 0 otherwise. The approximation to this expectation is then given by Eq. 3.4.

The importance sampling approximation to Eq. 3.8 is thus obtained by assuming that a set of hypotheses parameterized by centers and sizes (c_j, s_j) are sampled from the prior and activated by the likelihood $\mathbf{1}(\mathbf{0} \in (c_j, s_j))\, 1/m(s_j)$, to give

$$p(\mathbf{x}|\mathbf{0}) \;\approx\; \frac{\sum_{j=1}^{m} \mathbf{1}(\mathbf{x}, \mathbf{0} \in (c_j, s_j)) \frac{1}{m(s_j)}}{\sum_{j=1}^{m} \mathbf{1}(\mathbf{0} \in (c_j, s_j)) \frac{1}{m(s_j)}}, \tag{3.9}$$

where the numerator combines the indicator function that we want the expectation of, $\mathbf{1}(\mathbf{x} \in (c_j, s_j))$, with that in the likelihood. Since c and s are independent under the prior, we can also draw m samples of each and then take the sum over all m^2 pairs of c and s values, reducing the number of samples that need to be taken from the prior. The results of using this approximation are shown in Fig. 3.2. Relatively small numbers of sampled hypotheses (20 and 100) are sufficient to produce reasonable approximations to the generalization gradients associated with all of the prior distributions considered by Shepard [1987].

3.6 Simulation 3: Predicting the future

Remembering past events, like the local temperature in March in previous years, or the duration of red traffic lights, can help us make good predictions in everyday life. Griffiths and Tenenbaum[2006] studied people's predictions about a variety of everyday events, including the grosses of movies and the time to bake a cake, and

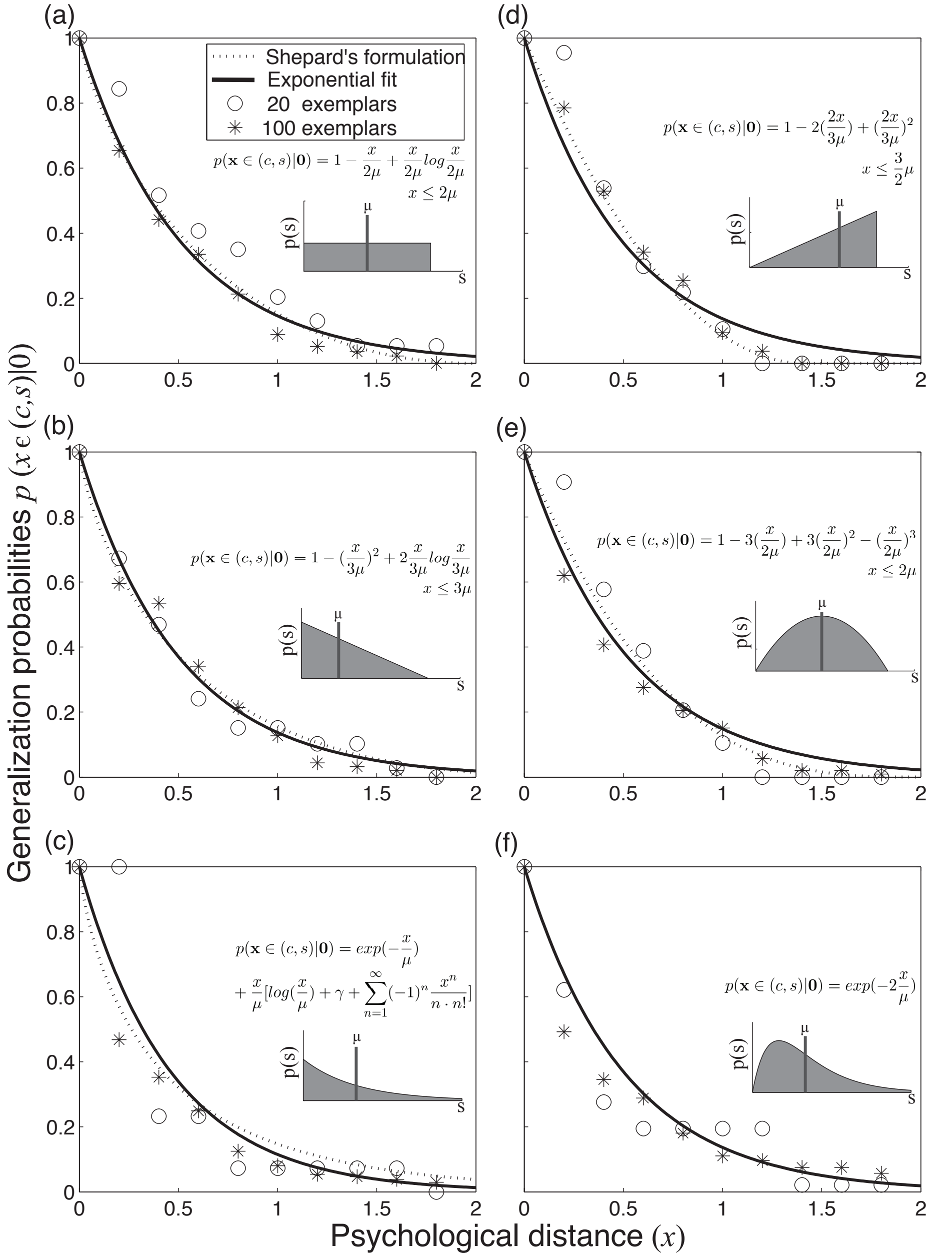


Figure 3.2: Exemplar models approximate six generalizations functions for different prior distributions on the size of consequential regions. The prior distributions are shown as inset shaded curves, reproducing Figure 3 of Shepard [1987]. Analytical results for the form of the generalization function are provided on the top of each inset prior, and are plotted in the dotted curve. An approximating exponential generalization function is plotted as a smooth curve. Exemplar models using 20 and 100 hypotheses sampled from the prior (corresponding to circles and asterisks respectively) provide a good approximation to these theoretical predictions.

found that these predictions corresponded strikingly well with the actual distributions of these quantities. Predicting the future in this way can be analyzed as Bayesian inference, and approximated using an exemplar model.

As formulated in Griffiths and Tenenbaum[2006], the statistical problem that people solved is inferring the total duration or extent of a quantity, t_{total}, from its current duration or extent, t. The goal is to compute the posterior median of t_{total} given t. Unlike the mean, the median gives a robust estimate of t_{total} when the posterior distribution is skewed, which is the case for many of these everyday quantities. The posterior median t^* is defined to be the value such that $p(t_{total} > t^*|t) = 0.5$, where the posterior distribution is obtained by applying Bayes' rule with an appropriate prior and likelihood. The prior $p(t_{total})$ depends on the distribution of the everyday quantity in question, with temperatures and traffic lights being associated with different distributions. As in the previous example, the likelihood is obtained by assuming that the phenomenon is encountered at a random point drawn uniformly from the interval between 0 and t_{total}, with $p(t|t_{total}) = 1/t_{total}$ for all $t_{total} > t$.

Making correct predictions about everyday events requires knowing the prior distributions of the relevant quantities – the grosses of movies, the time taken to bake a cake, and so forth. While it is unlikely that we store these distributions explicitly in memory, the posterior median can be approximated using stored exemplars that are sampled from the prior $p(t_{total})$ using Eq. 2.10. The posterior probability that a value of t_{total} is greater than t^* can be formulated as an expectation,

$$p(t_{total} > t^*|t) = E[\mathbf{1}(t_{total} > t^*)|t], \tag{3.10}$$

where $\mathbf{1}(t_{total} > t^*)$ is an indicator function taking the value 1 when its argument is true, and 0 otherwise, as in the previous example. This problem fits the schema for the general approximation to Bayesian inference given by Eq. 3.4, with the hypotheses

h being values of t_{total}, the data d being the observation t, and the function of interest $f(h)$ being the indicator function $\mathbf{1}(t_{total} > t^*)$. Consequently, the expectation given in Eq. 3.10 can be approximated using an exemplar model in which exemplars $t_{total,j}$ are sampled from the prior $p(t_{total})$ and activated by the likelihood $1/t_{total}$ if they are greater than t. This gives the approximation

$$p(t_{total} > t^*) \quad \approx \quad \frac{\sum_j \mathbf{1}(t_{total,j} > t^*, t_{total,j} > t)\frac{1}{t_{total,j}}}{\sum_j \mathbf{1}(t_{total,j} > t)\frac{1}{t_{total,j}}}. \tag{3.11}$$

The approximate median of the posterior distribution is the exemplar $t_{total,j}$ that has $p(t_{total} > t_{total,j}|t)$ closest to 0.5.

Considering limitations in memory capacity and computational power, we conducted two sets of simulations. In predicting the future, only values of t_{total} that are greater than the observed value of t are plausible, with all other values having a likelihood of 0. Consequently, sampling directly from the prior can be inefficient, with many samples being discarded. We can thus break the approximation process into two steps, with the first being generating a set of values of t_{total} from memory, and the second being assigning those values of t_{total} greater than t a likelihood of $1/t_{total}$ and normalizing. Our simulations considered limitations that could apply to either of these steps. In the *memory-limited* case, the number of exemplars generated from memory is fixed. In the *computation-limited* case, the bottleneck is the number of exemplars that can be processed simultaneously, placing a constraint on the number of exemplars such that $t_{total} > t$. In this case, we assume that exemplars are generated from memory until they reach this upper limit.

Fig. 3.3 shows the results of applying these different approximation schemes to the predicting the future task, varying the number of exemplars. We examined performance across seven prior distributions, corresponding to the baking time of cakes, human life spans, the grosses of movies, the duration of the reigns of pharaohs, the length of poems, the number of terms in the United States House of Representatives, and the runtime of movies, and for 5, 10, and 15 exemplars. In each case, we simulated the performance of 50 participants using the appropriate number of exemplars sampled directly from the prior (for the memory-limited case) or sampled from the prior but constrained to be consistent with the observed value of t (for the computation-limited case). In the memory-limited case, if none of the exemplars is larger than the observation, the observed value t is taken as the only exemplar which results in $t^* = t$. The figure also shows the quality of the approximation produced by directly sampling exemplars from the posterior distribution, rather than generating from the prior. For each approximation scheme, 50 simulated participants' responses were generated. The plot markers indicate the median and the 68% confidence interval on the median (ie. the 16th and 84th percentiles of the sampling distribution), computed using a bootstrap with 1000 samples drawn from the responses of these participants.

For a quantitative measure of the success of the approximation, we computed the sum of the absolute value of the deviations for each of the median results shown in Fig. 3.3 ($t^*_{ML}, t^*_{CL}, t^*_{SA}$ for memory-limited, computation-limited, and sampling respectively) to both the true function (t^*_{Bayes}) and to the median human responses (t^*_{human}). These error scores were then normalized by the difference in t^*_{Bayes} for the lowest and highest values of t for each prior, in order to compensate for the different scales of these quantities, and then summed across priors to produce the scores shown in Table 3.1. This quantitative analysis confirmed the trends evident from the figure. Approximation performance improved with more exemplars, but was already fairly good with only five exemplars. The memory-limited case tended to perform worse than the other approximations for a given number of exemplars, since some of the exemplars generated from the prior would not enter into the approximation for the reasons detailed above.

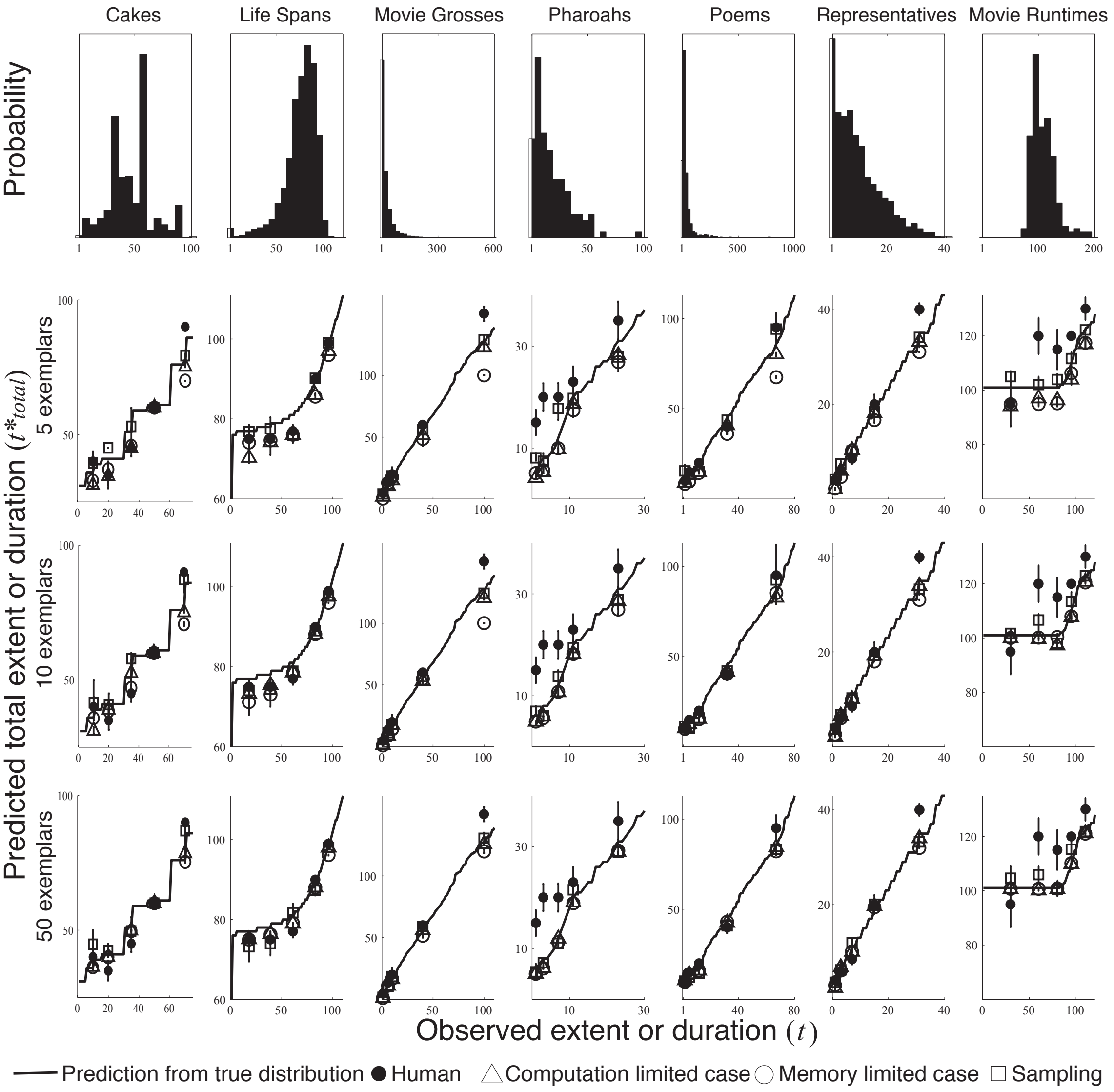


Figure 3.3: Simulations of prediction on everyday cognition, data from Griffiths and Tenenbaum [2006]. The first row is the prior distribution of each dataset. The second to fourth rows are simulations with 5, 10 and 50 exemplars for memory-limited and computation-limited exemplar models, as well as sampling from the posterior. The solid line shows the optimal responses given the prior distribution, and the black dots are the responses of human participants. For both simulations and human data, the plot markers indicate the median response across a population of 50 simulated participants. Error bars show a 68% confidence interval computed by 1000 sample bootstrap.

Table 3.1: Comparison of Approximation Schemes with Exact Bayes and Human Data

	5 exemplars	10 exemplars	50 exemplars
$\sum \lvert t^*_{ML} - t^*_{Bayes} \rvert$	4.2003	2.3333	1.2366
$\sum \lvert t^*_{ML} - t^*_{human} \rvert$	8.3023	7.0858	6.6757
$\sum \lvert t^*_{CL} - t^*_{Bayes} \rvert$	3.5601	1.8620	1.0798
$\sum \lvert t^*_{CL} - t^*_{human} \rvert$	7.8566	6.8283	6.1023
$\sum \lvert t^*_{SA} - t^*_{Bayes} \rvert$	1.4706	1.7449	2.3050
$\sum \lvert t^*_{SA} - t^*_{human} \rvert$	6.8043	6.0633	6.5741
$\sum \lvert t^*_{human} - t^*_{Bayes} \rvert$	6.2626		

Note: Subscripts correspond to memory limited (ML), computation limited (CL), sampling from the posterior (SA), and true Bayesian and human estimates of t^*. Error scores were summed across values of t for each prior, normalized as described in the text, and then summed across priors.

The question of whether approximations based on a small number of exemplars might account for the results of Griffiths and Tenenbaum[2006] was independently raised by Mozer *et al.*[2008], who argued that a close correspondence to the posterior median could be produced by aggregating responses across a large number of participants who each had only limited knowledge of the appropriate prior, such as a handful of samples from that distribution. The original model considered by Mozer et al. [2008], which estimates t^* as the minimum of the set of exemplars greater than t, does not have an interpretation as importance sampling, and degenerates as an approximation as the number of exemplars increases, rather than improving. However, one of the variants on this model, called GTkGuess in their paper, is equivalent to our memory-limited importance sampling approximation provided at least one sampled exemplar is greater than t. Consistent with the results presented here, Mozer et al. [2008] demonstrated that this model produced a good correspondence with the results of Griffiths and Tenenbaum[2006] with only a small number of exemplars, considering both aggregate performance and the amount of variability produced by different approximation schemes.

One important difference between the analysis we present here and that of Mozer et al. [2008] is that we do not necessarily view using an exemplar model to approximate Bayesian inference as being related to having limited prior knowledge. For Mozer et al. [2008], the exemplars used in approximating Bayesian inference were taken to represent all that a given individual knew about a phenomenon. Since each participant in Griffiths and Tenenbaum[2006] made only a single judgment about each phenomenon, it was possible to accurately model the aggregate judgments by making this assumption. However, another possibility that is equally consistent with the data is that each individual has a large pool of exemplars available, and only samples a small number in making a given prediction. In this case, a small number of exemplars are used in order to make the Bayesian computation efficient, not because they represent the complete knowledge of the learner. These two possibilities can be differentiated by conducting an experiment in which individuals make multiple judgments about a given phenomenon. If participants only have access to a small number of exemplars, they produce very similar responses for a range of values of t, while if they are sampling different sets of exemplars on different trials, their responses should increase as a function of t in a way that is consistent with applying Bayesian inference. Lewandowsky *et al.*[in press] conducted such an experiment, and found support for the latter hypothesis.

3.7 Simulation 4: Concept learning

The simulations we have presented so far correspond to cases where Bayesian inference is performed with a hypothesis space that contains only hypotheses that correspond to continuous quantities (formant values, the size of consequential regions, the extent or duration of everyday phenomena). However, Bayesian inference is also carried out with hypothesis spaces in which each hypothesis is discrete, and qualitatively different from other hypotheses. The "number game" of Tenenbaum [1999]; [Tenenbaum and Griffiths, 2001] is a good example. This game is formulated as follows: Given natural numbers from 1 to 100, if a number or set of numbers x belongs to an unknown set C, what is the probability that another number y also belongs to the same set? For example, if the numbers $\{59, 60, 61, 62\}$ all belong to an unknown set, what is the probability that 64 belongs to that set? What about 16?

The problem of determining whether y belongs to the same set as x is another instance of the problem of generalization, and can be answered using a similar Bayesian inference. Our data are the knowledge that x belongs to the set C, and our hypotheses concern the nature of C. Since C is unknown, we should sum over all possible hypotheses h in the hypothesis space $\mathcal{H}$ when evaluating whether y belongs to C,

$$p(y \in C|x) = \sum_{h \in \mathcal{H}} p(y \in C|h)p(h|x) = \sum_{h \in \mathcal{H}} \mathbf{1}(y \in h)p(h|x), \tag{3.12}$$

where $\mathbf{1}(y \in h)$ is the indicator function of the statement $y \in h$, taking value 1 if this is true and 0 otherwise. In the analysis presented by Tenenbaum [1999]; [Tenenbaum and Griffiths, 2001], the likelihood $p(x|h)$ is proportional to the inverse of the size of h (the "size principle") being $1/|h|$ if $x \in h$ and 0 otherwise. This corresponds to the uniform sampling assumption made in the previous two examples. A hypothesis space $\mathcal{H}$ containing a total of 6,412 hypotheses was used, including intervals of numbers spanning a certain range, even numbers, odd numbers, primes, and cubes.

The number game is challenging because any given number (say $x = 8$) is consistent with many hypotheses (not only intervals containing 8, but also hypotheses such as even numbers, cubic numbers, number with final digit 8, etc.). Interestingly, the responses of human participants can be captured quite accurately with this Bayesian model (Fig. 3.4a). However, this involves instantiating all 6,412 hypotheses, calculating the likelihood for each rule and integrating over the product of the prior and likelihood. Such computations are challenging, so a mechanism that approximates the exact solution is desirable. Fortunately, the probability computed in Eq. 3.12 is an expectation, and can be approximated by importance sampling and thus by an exemplar model.

The number game is another instance of a problem that requires the more general approximation scheme summarized in Eq. 3.4. The hypotheses h are candidates for the identity of the concept C, the data d are the observation that x belongs to C, and the function $f(h)$ that we want to evaluate the expectation of is the indicator function $\mathbf{1}(y \in h)p(h|x)$. We can approximate this expectation by sampling hypotheses h_j from the prior $p(h)$, and re-weighting those hypotheses by the likelihood $p(x|h)$, with

$$p(y \in C|x) \approx \frac{\sum_j \mathbf{1}(y \in h_j, x \in h_j)1/|h_j|}{\sum_j \mathbf{1}(x \in h_j)1/|h_j|}, \tag{3.13}$$

meaning that $p(y \in C|x)$ is just the ratio of the summed likelihoods of the hypotheses stored in memory that generate y to the summed likelihoods of all hypotheses stored in memory.

Fig. 3.4b and c show generalization responses for different sets of numbers, x, for a single simulated participant. As in Simulation 3, we conducted simulations for both memory- and computation-limited approximations, with the latter case corresponding to generating sample hypotheses h from the prior until a fixed number consistent with x had been generated. The simulations used the same parameters as those in the full Bayesian model of Tenenbaum and Griffiths [2001], except the likelihood function assigns a small non-zero probability to all natural numbers from 1 to 100 for every hypothesis to ensure numerical stability. The results suggest that a small number of exemplars (20 and 50 for computation-limited and memory-limited respectively) is sufficient to account for human performance. The memory-limited case needs more exemplars because not all exemplars are qualified hypotheses. Therefore, the effective

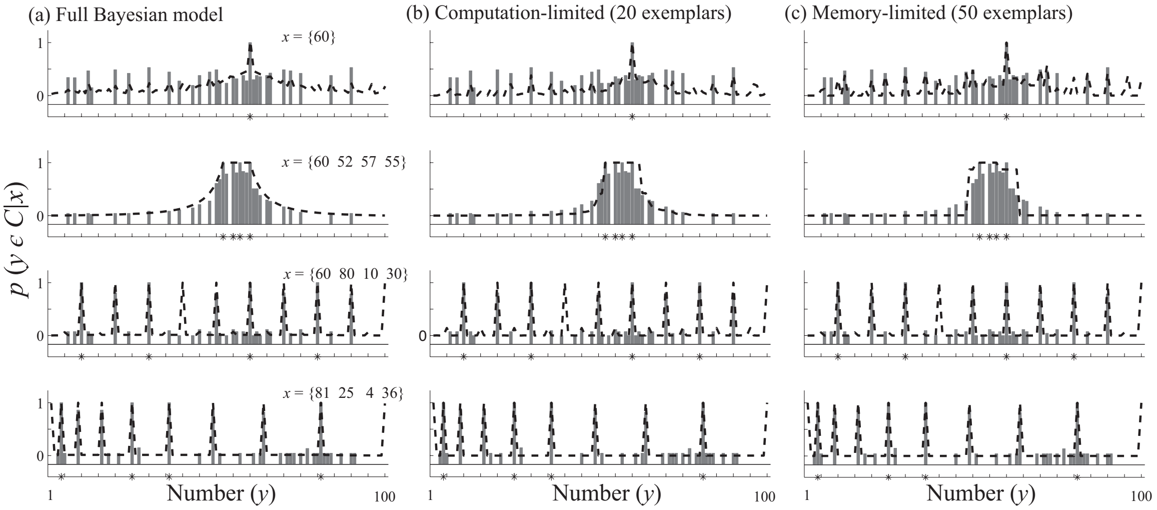


Figure 3.4: Simulations (dashed line) and behavioral data from Tenenbaum [1999] (gray bars) for the number game. The full Bayesian model uses 6,412 hypotheses. Results of computation-limited (20 exemplars) and memory-limited (50 exemplars) exemplar models are based on a single simulated participant with a set of hypotheses (exemplars) sampled from the prior. Models are tested under conditions suggesting single point generalization $x = 60$, a consecutive interval $x = \{60, 52, 57, 55\}$, multiples of 10 $x = \{60, 80, 10, 30\}$ and squares $x = \{81, 25, 4, 36\}$.

number of exemplars, which determines the computational load, is small. The consistency of these results with the human judgments indicates that exemplar models provide a plausible mechanism that relies on reasonable memory and computational resources and can be used with highly structured hypothesis spaces.

To further evaluate the model, we compared the variance of the predictions produced by importance sampling with the variability among individuals on this task. Since the model predictions rely on a sample from the prior, there can be variability between simulated participants which we can compare with the variability among human participants. Moreover, we should expect to see specific simulated participants who produced behavior similar to that of specific human participants. Fig. 3.5a shows the variability among the eight participants analyzed by Tenenbaum [1999], together with the variability among 100 simulated participants (using the memory-limited case). Both human and simulated participants exhibit significant variability in their responses, particularly for the stimulus $x = \{60\}$. The patterns of responses also share some key features. For $x = \{60, 52, 57, 55\}$, since there is no specific numeric rule describing the set, most plausible hypotheses are intervals containing x. Therefore, we expect higher variability near the boundary of the set (ie. below 52 or greater than 60) and lower variability within the set. For $x = \{60, 80, 10, 30\}$, high variability in generalization to multiples of five and ten is observed in both human and simulated participants.

The variability seen in the human and simulated participants disagree in two respects. First, there is significant baseline variability in the human responses that is not captured by the model, especially for $x = \{60, 52, 57, 55\}$ and $x = \{60, 80, 10, 30\}$. After looking in detail at individual trials, we found that high baseline variability is partly due to inconsistent use of the rating scale (which ranged from 1-7) to express "low probability." For example, for $x = \{60, 52, 57, 55\}$, two out of eight participants gave minimum responses of 2 out of 7, while the other six used the full range and had minimum responses of 1. A second point of difference between the human and simulated responses is in the use of the "square numbers" hypotheses with $x = \{81, 25, 4, 36\}$. The model displays greater variability than seen among human participants when generalizing to other squares from this set. This is due to the fact that the memory-limited exemplar model is not guaranteed to sample the "square numbers" rule in every trial, while the educated participants used by Tenenbaum [1999] consistently recognized this mathematical rule.

For a closer look at the way that variability manifests in the model, we examined whether it was possible to find patterns of predictions that matched the behavior of individual participants. Fig. 3.5b shows some close correspondences between human and simulated participants. Each row shows the responses of a different human participant, together with the closest-matching responses chosen from the 100 simulated participants used in our analysis of variability. In each case the correlation between human and simulated participants was greater than $r = 0.95$, and many of the details of the responses are in correspondence. For example, in the case of $x = \{60\}$, this individual evaluated multiple hypotheses such as intervals, multiple of 10 and multiples of 6, and a similar pattern appears in the model predictions.

3.8 Simulation 5: Category effects on reconstruction from memory

Retrieving or reconstructing items from memory can also be formulated as a problem of statistical inference, with Bayes' rule being used to evaluate which item in memory might correspond to a particular cue [Anderson and Milson, 1989; Shiffrin and Steyvers, 1997; Hemmer and Steyvers, 2009; Huttenlocher *et al.*, 2000]. Examining how this kind of probabilistic inference can be approximated using an exemplar model

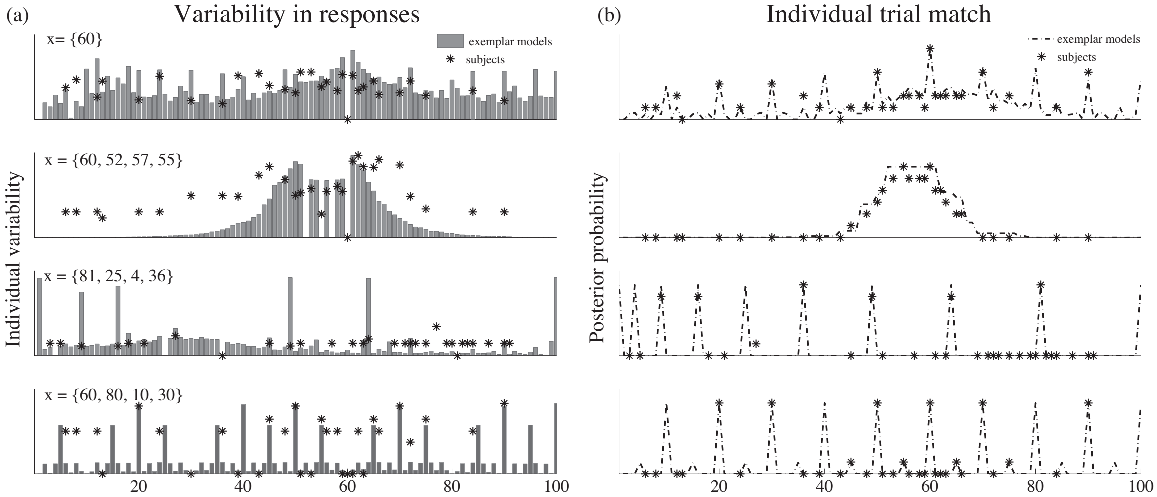


Figure 3.5: Variability across human and simulated participants in the number game. (a) The standard deviation of the ratings produced by eight human participants in the number game (denoted with asterisks) is compared with the standard deviation of the posterior probabilities produced by 100 simulated subjects. (b) Responses from four human participants, compared with the closest matching simulated participants from the pool of 100 used in evaluating variability.

has the potential to be particularly informative, since exemplar models themselves are based on memory. This creates an opportunity to consider how exemplars come to be stored in memory, and what role statistical inference plays in this process.

We will focus on the problem of reconstructing items from memory, and in particular on a study by Huttenlocher et al. [2000, Experiment 1] examining how the relative frequencies of items within a category can be used to improve accuracy in reproducing stimuli. In this study participants learned the distribution associated with a novel one-dimensional stimulus (the width of a schematic fish). The form of this distribution varied across participants. Some participants learned a single category, which was associated with either a uniform or a Gaussian distribution on fish width. Other participants learned two categories, each of which was associated with one half of the uniform distribution used in the one category case (the categories thus corresponded to "slender" and "fat" fish). During training, participants were briefly shown a stimulus, and then asked to reproduce that stimulus from memory (having been provided with its category label). Reconstructions were produced by adjusting the size of a schematic fish until participants felt that they had matched the size of the original stimulus.

Reconstructing a stimulus from memory can be analyzed as a Bayesian inference. Returning to the very first example of Bayesian inference we considered in this chapter, we might assume that the observed stimulus x is taken as a noisily perceived instance of some true stimulus x^*, with the noise process described by the distribution $p(x|x^*)$. The prior distribution on x^* is provided by the category c, which is associated with a distribution $p(x^*|c)$. The best reconstruction of x^*, in the sense of minimizing the squared error between the reconstruction and the true value, is the posterior expectation of x^* given x and c,

$$E[x^*|x,c] \quad = \quad \int x^* p(x^*|x,c) dx^*, \tag{3.14}$$

where the posterior distribution $p(x^*|x,c)$ is calculated using Bayes' rule. Huttenlocher et al. [2000] explicitly tested this model of reconstruction from memory, arguing that using category information to guide reconstruction should increase accuracy.

The problem of reconstruction from memory is of exactly the same form as the stimulus de-noising problem we used to demonstrate the equivalence between importance sampling and exemplar models. The expectation in Eq. 3.14 can be approximated by storing a set of exemplars x_j^* in memory, sampled from the prior $p(x^*|c)$, and then activating those exemplars in proportion to the likelihood $p(x|x^*)$. Huttenlocher et al. [2000] assumed that the likelihood was a Gaussian distribution with a mean at x^*, and explored several different prior distributions $p(x^*|c)$. In each case, the Bayesian inference required to reconstruct a stimulus from memory can be approximated using an exemplar model of the form specified in Eq. 2.10.

Although this analysis of reconstruction from memory is similar to that for the perceptual magnet effect, there are two important differences. First, category labels are given explicitly in the case of reconstruction, but are unknown in the perceptual magnet effect. Second, and perhaps more importantly, the experiments conducted to explore these phenomena differ in how the relevant priors were acquired. The prior distribution on speech sounds was established before the experiment exploring the perceptual magnet effect, as a result of learning the distributions associated with these sounds in English. In contrast, the prior being used to reconstruct the stimuli in the experiment conducted by Huttenlocher et al. [2000] is learned on the fly, through the process of forming the reconstructions. The reconstruction produced on one trial might thus play the role of a stored exemplar on a later trial.

To explore the effects of incrementally building a set of exemplars over time, we conducted a series of simulations of this study in which we used a variant on the standard exemplar model. The reconstruction of the first stimulus seen by each simulated participant was taken to be exactly equal to that stimulus. Each subsequent stimulus was reconstructed using an exemplar model with the previous n stimuli as exemplars (or all stimuli, if fewer than n have been observed), including the observed value of the current stimulus. Following Huttenlocher et al. [2000], the likelihood $p(x|x^*)$ was taken to be a normal distribution with mean x^* and variance σ^2. The resulting model has two parameters: the noise level σ^2, and the memory capacity n. Our simulations varied these two parameters, with $n = \{1, 2, 5, 10, \infty\}$ and $\sigma = \{1, \ldots, 10\}$ pixels.[3]

Fig. 3.6 shows the results of these simulations. In each case, we plot the bias in reconstruction for stimuli of different widths, defined to be the difference between the width of the reconstruction and the width of the stimulus. In general, stimuli that are

[3] We also conducted simulations in cases where perceptual noise was considered and reconstructed stimuli, instead of original stimuli, were taken as exemplars. All of these variations produced similar results.

smaller than the mean of a category show a positive bias and stimuli that are larger show a negative bias, consistent with reconstructions moving towards the mean of each category. This effect comes out in all of our models, being the basic prediction resulting from a Bayesian analysis of this problem. However, the results also show how the exemplar models capture some subtle characteristics of the data. For example, in the normal prior condition (the middle row of the figure), a full Bayesian model would predict that bias is a linear function of fish width. This prediction is quite clearly reflected in the results for $n = \infty$, which most closely approximates exact Bayesian inference. In contrast, both the human data and the models with smaller values of n show a non-linear function, with bias reduced for more extreme stimuli. To understand this effect, we should note that the current observation x is always included as an exemplar in producing the reconstruction of x^*. Thus, when x takes an extreme value lying at the tails of the prior, it is often over-weighted since recent observations are unlikely to lie in proximity to this extreme value. In this case, the reconstruction of x^* relies more on x itself, resulting in smaller bias.

3.9 Discussion

The formal correspondence that we have shown to exist between exemplar models and importance sampling suggests a way to solve the computationally challenging problem of probabilistic inference using a common computational model of psychological processes. Our five simulations illustrate how this approach can be applied in a range of settings where probabilistic models have previously been proposed. Simulation 1 showed that exemplar models can be used to perform Bayesian inference for a simple speech perception problem, providing an account of the perceptual magnet effect that does not require parametric assumptions about the distribution of speech sounds associated with phonetic categories, or any form of learning of these

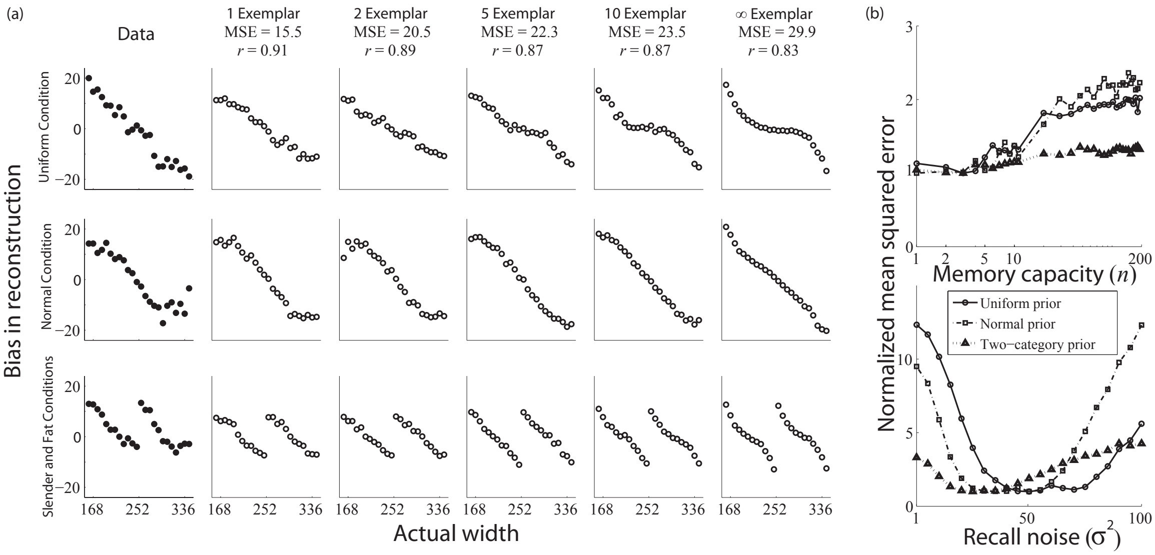


Figure 3.6: Reconstruction from memory with online recruitment of exemplars. (a) The left column shows the average bias in the reconstructed stimuli produced by participants (measured as the difference between the actual and reconstructed width of fish, in pixels) as a function of actual width. The rows show reconstructions produced for three prior distributions: a single category following a uniform and a normal distribution, and two categories following uniform distributions. Data are from Huttenlocher et al. [2000, Experiment 1]. The remaining columns show simulations using exemplar models with a memory capacity of 1, 2, 5, 10 and ∞ exemplars. Data were generated in a way that was consistent with the original experiment, and the results show an average across 10 simulated participants with 192 trials per participant. The only free parameter, the assumed noise level σ^2, is specified by minimizing mean squared error (MSE) in each case. (b) Sensitivity of the results to memory capacity and recall noise. In the upper panel, memory capacity (in number of exemplars) is fixed and σ^2 is chosen to minimize MSE. Interestingly, MSE grows with increasing memory capacity, suggesting that a limited memory model (< 10 exemplars) is consistent with human behavior. In the lower panel, the effect of different noise levels σ^2 is examined, optimizing memory capacity. For all three priors, the error curves have concave bell shape and share a region of minimum error, suggesting that a single assumed noise level can account for results in all three conditions.

distributions. Simulation 2 demonstrated that a similar approach could approximate the predictions of Shepard's [1987] classic analysis of generalization. Simulation 3 examined how exemplar models could be used in predicting the future. Simulation 4 extended our analysis to a case where hypotheses represent discrete, qualitatively different accounts of observed data. Finally, Simulation 5 considered how exemplars might be recruited in the course of an experiment, and showed that this approach could account for the results of a study of reconstruction from memory.

In the remainder of the chapter, we discuss three issues raised by these results. First, while our simulations show that exemplar models can be used to approximate Bayesian inference in a range of settings, this approach will not provide good approximations in all cases. The relationship with importance sampling makes it possible to clearly state in which cases we expect this to be an effective approximation scheme. Second, none of the cases we consider involve any kind of dynamics, with the hypothesis space remaining static over time. Since some cognitive problems require dealing with hypothesis spaces that change in size and content over time, we outline how our approach can be extended to accommodate this situation. Finally, we consider some of the broader implications of the correspondence between exemplar models and importance sampling that we have identified in this chapter, viewing this result as just one instance of a more general approach towards connecting rational models of cognition with psychological processes.

3.9.1 The limits of importance sampling

While importance sampling is widely used to approximate probabilistic inference, it is not appropriate for all problems. As discussed above, the quality of the approximation provided by importance sampling depends on the relationship between the target distribution $p(y)$, the function $g(y)$ for which we want to find an expected value, and the proposal distribution $q(y)$. In particular, we want the proposal distribution to assign high probability to values of y for which $p(y)$ and $g(y)$ are both large, and low probability to other values of y. Otherwise, samples from the proposal distribution may not correspond to values of y that make a large contribution to the expectation of $g(y)$.

The relationship between importance sampling and exemplar models that we have identified relies on the assumption that the exemplars are drawn from the prior (ie. that the prior is used as a proposal distribution). This makes it easy to identify the limitations of this approach: Bayesian inference can only be approximated effectively using the kind of exemplar models we have considered in this chapter when there is a reasonably close match between the posterior and the prior. This will be the case when the data are relatively uninformative, meaning that the posterior does not deviate significantly from the prior. Data can be uninformative because of small sample size, or because of a high level of uncertainty (as reflected in the likelihood). All of the settings we explored in our simulations met this criterion, requiring an inference to be made on the basis of only one or at most a handful of stimuli.

One way to extend the range of problems for which exemplar models yield approximations to Bayesian inference might be to remove the assumption that the exemplars are drawn from the prior. While we have focused on the equivalence between Eq. 3.3 and 2.10, the exemplar-based computations represented by Eq. 3.3 are also equivalent to those used in the more general formulation of the importance sampler in Eq. 2.7. Thus, exemplar models can be used to approximate expectations over a distribution $p(x^*|x)$ when the exemplars are generated from any distribution $q(x^*)$, provided the similarity function used to activate each exemplar is proportional to $p(x^*|x)/q(x^*)$. When $q(x^*) = p(x^*)$, we obtain the class of models analyzed in this chapter. However, relaxing this assumption broadens the range of proposal distributions that can be used, and may make it possible for exemplar models to produce efficient approximations to Bayesian inference across a wider range of problems.

3.9.2 Approximating dynamic inferences

A second limitation of the approach that we have presented in this chapter is that it is only appropriate in cases where the hypothesis space is static, with the same hypotheses being used in multiple inferences. The simple strategy of using a stored set of hypotheses does not work in cases where the hypothesis space itself changes over time, and results in a particularly poor approximation when that hypothesis space grows with the number of observations. One example where such a problem arises is dividing a set of observations into clusters, as in Anderson's [1990; 1991] rational model of categorization. In this model, the hypothesis space consists of all possible clusterings of a set of observations. This hypothesis space has to be revised with each new observation, reflecting all of the ways in which that observation could be added to the existing clusters. Not only does the hypothesis space change over time, but it grows super-exponentially in the number of observations.

While exemplar models are not appropriate for this situation, they are closely related to another Monte Carlo method that can be extremely effective for approximating dynamic inferences. This method, known as *particle filtering*, translates importance sampling into a dynamic setting. The basic idea is that the posterior distribution over hypotheses after n observations should be closely related to the posterior distribution after $n + 1$ observations, in the same way that the prior and posterior were closely related in the examples we considered above. The posterior after $n + 1$ observations can thus be approximated by importance sampling, using a proposal distribution based on the posterior after n observations. This idea can be applied recursively: while we may not know the posterior after n observations, we can approximate this by importance sampling too, using a proposal distribution based on the posterior after $n-1$ observations, and so on. A particle filter thus consists of a set of samples that evolves through time, with samples from the posterior distribution after n observations being used to generate samples from the posterior distribution after $n+1$ observations.

Particle filters share with the models that we have discussed in this chapter the idea of approximating a probability distribution with a small number of samples. However, the models we have considered all assume that these samples are fixed exemplars stored in memory, while a particle filter dynamically constructs a set of samples in response to the information provided by a sequence of observations. Despite this difference, the basic components of a particle filter are very similar to the components of an exemplar model, requiring activation of hypotheses in proportion to their likelihood, normalization, and random selection. As a consequence, particle filters may provide a psychologically plausible scheme for approximating Bayesian inference in dynamic settings. This idea has been explored in the context of the rational model of categorization by Sanborn *et al.*[2006], and similar models have been proposed as explanations of change point detection [Brown and Steyvers, 2009], associative learning [Daw and Courville, 2008], sentence processing [Levy *et al.*, 2009], and reinforcement learning [Yi *et al.*, in press]. In Chapter 5, we study sequential estimation problems in detail and find that importance sampling plays an important role in the neural implementation of dynamic inferences.

3.9.3 Rational process models

Probabilistic models of cognition are typically expressed at Marr's [1982] computational level, analyzing learning, reasoning, and perception in terms of ideal solutions to abstract problems posed by the environment. This is at odds with much of the history of cognitive psychology, in which theories are typically expressed at the level of representation and algorithm. As Marr noted, these two levels should not be considered independent of one another: findings at one level provide constraints on theories at the other. However, despite a few notable exceptions e.g., [Kruschke, 2006], there has been little exploration of the relationship between probabilistic models of cognition and psychological process models.

The connection between importance sampling and exemplar models that we have established in this chapter hints at a strategy that might help to establish a more general link between probabilistic models formulated at the computational level and psychological process models expressed at the algorithmic level. The computational challenges posed by probabilistic inference do not arise just as an obstacle for rational models of cognition: they also appear whenever a computer scientist or statistician wants to work with a probabilistic model. As a consequence, researchers in computer science and statistics have developed a variety of schemes for efficiently approximating probabilistic inference. Importance sampling is just one of these schemes, and the fact that it can be implemented in a psychologically plausible way suggests that there may be other approximate algorithms for probabilistic inference that are candidate explanations for how people might address the computational challenges posed by rational models of cognition.

In embodying an effective solution to the problem of approximating probabilistic inference, and making use of psychological notions common in mechanistic process models, exemplar models are an instance of a "rational" process model. Such rational process models push the principle of rationality embodied in existing rational models of cognition a level deeper. Rational models of cognition apply the principle of rationality – the assumption that optimal solutions are informative about human behavior – at the computational level. Rational process models apply a similar principle at the level of representation and algorithm, assuming that the psychological processes that are used to approximate probabilistic inference represent efficient solutions to this problem. As noted above, particle filters are another instance of a rational process model, but the great diversity of efficient approximation algorithms for probabilistic inference suggests that there may be many other psychologically plausible mechanisms for solving this problem that are still to be discovered.

In providing a connection between abstract probabilistic models of cognition and psychological processes, rational process models also have the potential to help us understand the neural mechanisms that underlie probabilistic computation. For example, our analysis of the perceptual magnet effect revealed that approximating Bayesian inference by importance sampling resulted in a model that was extremely similar to a neural network model proposed by Guenther and Gjaja. This connection is valuable in two ways: It shows how such a neural network could be used to approximate Bayesian inference, and it provides a high-level explanation of why this neural mechanism produces the perceptual magnet effect. We anticipate that similar connections will exist in other domains, particularly given the close correspondence between exemplar models and neural network architectures such as radial basis function networks [Kruschke, 1992; Shi and Griffiths, in press].

Chapter 4

Neural implementation of hierarchical Bayes by importance sampling

In this chapter, we show that importance sampling [Hastings, 1970] can be implemented in a simple neural circuit. The properties of the population of neurons comprising this circuit capture prior knowledge about the structure of the environment, and their firing rates encode information about an observed stimulus. Integrating across this population acts like averaging over the possible states of the environment that could have produced the observed stimulus, providing a way to deal with noisy inputs, incomplete information, and unobserved variables. We show how this basic idea can be extended to make it possible to combine sources of information and to propagate uncertainty through multiple layers of random variables, using a recursive form of importance sampling to approximate hierarchical Bayesian inference.

4.1 Neural implementation of importance sampling

The central idea behind the results presented in the remainder of the chapter is that importance sampling provides a way to approximate Bayesian inference that is easy to implement in a neural circuit. Specifically, we show that the key components of an importance sampler can be realized in the brain if: 1) there are feature detection neurons with preferred stimulus tuning curves proportional to the likelihood $p(s|x_i)$; 2) the frequency of these feature detection neurons is determined by the prior $p(x)$; and 3) divisive normalization can be realized by some biological mechanism. s represents the input stimuli and x_i represents exemplars. In this section, we first describe a radial basis function network implementing importance sampling, then discuss the feasibility of these three assumptions. The model is then extended to hierarchical Bayesian inference and networks of spiking neurons. Following the introduction of each new technical idea, we provide an example applying the model to a behavioral experiment.

Radial basis function (RBF) networks are a kind of multi-layer neural network [Poggio and Girosi, 1990]. A network consists of a set of input units, a set of output units, and a set of "hidden" units that connect inputs to outputs. The hidden units are parameterized by locations in a latent space x_i. On presentation of a stimulus s, these hidden units are activated according to a function that depends only on the distance $||s - x_i||$. Implementing importance sampling with RBF networks is straightforward. Each hidden unit represents a stimulus value x_i drawn from the prior, playing the role of a neuron sensitive to particular stimulus values s (Fig. 4.1). They also receive input from an inhibitory unit that sums the activities of all the hidden units. The activation function of this hidden unit is taken to be proportional to the likelihood $p(s|x_i)$. The ith hidden unit makes a synaptic connection to output unit j with strength $f_j(x_i)$, where f_j is a function of interest. Such a RBF network produces output exactly in the form of Eq. 2.10.

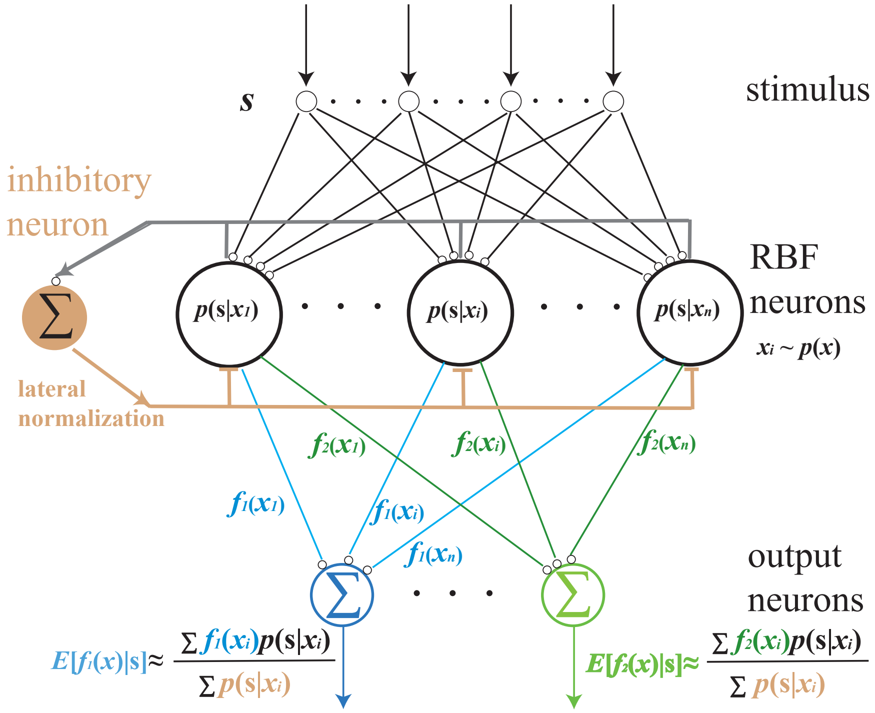


Figure 4.1: Importance sampler realized by a RBF network. Stimuli s are fed into hidden units whose tuning curves are proportional to the likelihood function $p(s|x)$. Their responses are normalized by lateral inhibition and then fed into output neurons. Multiple output neurons allow the same set of x_i to be used to calculate the posterior expectations of multiple functions.

4.1.1 Tuning curves, priors and divisive normalization

We now examine the neural correlates of the three components that went into the RBF model outlined above. First, we have a set of hidden units that have activation functions proportional to the likelihood $p(s|x_i)$. These hidden units are intended to be abstract versions of neurons, with the activation function corresponding to the tuning curve of the neuron. The responses of cortical neurons to stimuli are often characterized by receptive fields and tuning curves, where receptive fields specify the domain within a stimulus feature space that modify the neuron's response and tuning curves detail how the neuron's responses change with different feature values. A typical tuning curve (like orientation tuning in V1 simple cells) has a bell shape that peaks at the neuron's preferred stimulus parameter and diminishes as that parameter diverges [Swindale, 1998]. The neural plausibility of this component of the model thus rests on whether there is a population of neurons with tuning curves that can be interpreted as reflecting the likelihood $p(s|x_i)$ for a particular problem of Bayesian inference.

Second, importance sampling requires neurons with preferred stimuli x_i to appear with frequency proportional to the prior distribution $p(x)$. This can be realized if the number of neurons representing x is roughly proportional to $p(x)$. While systematic study of the distribution of neurons over their preferred stimuli is technically challenging, there are cases where this assumption seems to hold. For example, research on the "oblique effect" – the finding that the human visual system is more sensitive to changes in orientation near cardinal orientations (ie. horizontal and vertical) – supports the idea that the distribution of orientation tuning curves in V1 is proportional to the distribution of orientations in the visual environment. Electrophysiology [De Valois *et al.*, 1982], optical imaging [Coppola *et al.*, 1998] and fMRI studies [Furmanski and Engel, 2000] have found that there are more V1 neurons tuned to cardinal orientations than to oblique orientations, consistent with the prevalence of horizontal and vertical lines in the visual environment. Other evidence comes from motor areas. Repetitive stimulation of a finger expands its corresponding cortical representation in somatosensory area [Hodzic *et al.*, 2004], suggesting more neurons are recruited to represent this stimulus after its prior density is increased.

Third, divisive normalization is a critical component in many neural models, notably in the study of attention modulation [Lee and Mumford, 2003; Reynolds and Heeger, 2009]. It has been suggested that biophysical mechanisms such as shunting inhibition and synaptic depression might account for normalization and gain control [Kouh and Poggio, 2008; Mitchell and Silver, 2003; Rothman *et al.*, 2009]. Moreover, local interneurons [Markram *et al.*, 2004] act as modulators for pooled inhibitory inputs and are good candidates for performing normalization. Our approach makes no specific claims about the underlying biophysical processes, but gains support from the literature suggesting that there are plausible neural mechanisms for performing divisive normalization.

4.1.2 Example 1: Sensorimotor integration

Our first example uses the basic importance sampling scheme (Eq. 2.10) to solve a problem of Bayesian inference that the brain solves every day: sensorimotor integration. In [Körding and Wolpert, 2004], subjects were trained to reach a visual target with a cursor in an environment that allows the cursor to be displaced from their finger position and provides noisy feedback on the extent of displacement. During the experiment, displacements were randomly chosen from a fixed prior distribution and visual feedback on the cursor position was provided briefly midway through the movement. This feedback was either presented clearly (σ_0 condition, in which the uncertainty comes solely from intrinsic noise), blurred by medium (σ_M condition) or large (σ_L condition) amounts of noise, or simply blocked (equivalent to infinite noise, σ_∞ condition). The finger's position at the end of movement was revealed on clear feedback trials (σ_0). The Bayesian analysis predicts that the prior should play a more important role in estimation as sensory noise increases.

This experiment fits the schema for Bayesian inference outlined above, with s being the observed displacement and x the true displacement. In the first experiment described in [Körding and Wolpert, 2004], the prior distribution was Gaussian. Subjects were first trained on the distribution of displacements and then performed the task for 1000 trials under different uncertainty conditions. Letting μ_x and σ_x denote the mean and standard deviation of the prior distribution $p(x)$, and assuming that the likelihood function $p(s|x)$ given true lateral shift x is also a Gaussian with mean x and variance σ_s^2, the posterior distribution $p(x|s)$ is again Gaussian, with mean μ'_x and variance σ'_x, where

$$E[x|s] = \mu'_x = \frac{\sigma_x^2 s + \sigma_s^2 \mu_x}{\sigma_x^2 + \sigma_s^2}, \qquad \sigma'_x = \frac{\sigma_x \sigma_s}{\sqrt{\sigma_x^2 + \sigma_s^2}}. \tag{4.1}$$

Therefore, the mean deviation from target, which is the difference between μ'_x and s, is a linear function of s with a slope determined by the ratio σ_x^2/σ_s^2.

The results of [Körding and Wolpert, 2004] showed exactly this pattern (Fig. 4.2a). However, it is not necessary to assume that subjects were integrating over a continuous space as in exact Bayesian inference in order to achieve this result. Fig. 4.2b show corresponding simulation results produced using an RBF network. Assuming there are 50 feature detection neurons whose preferred stimuli are positioned at typical displacements x_i learned from the training session (i.e. the prior distribution $p(x)$). Tuning curves of these neurons, parameterized by x_i and noise levels σ_0, σ_M, and σ_L, have Gaussian shape $N(x_i, \sigma^2)$ and measure the likelihood (or similarity) for s and x_i. An output neuron, receiving inputs from feature detection neurons weighted by

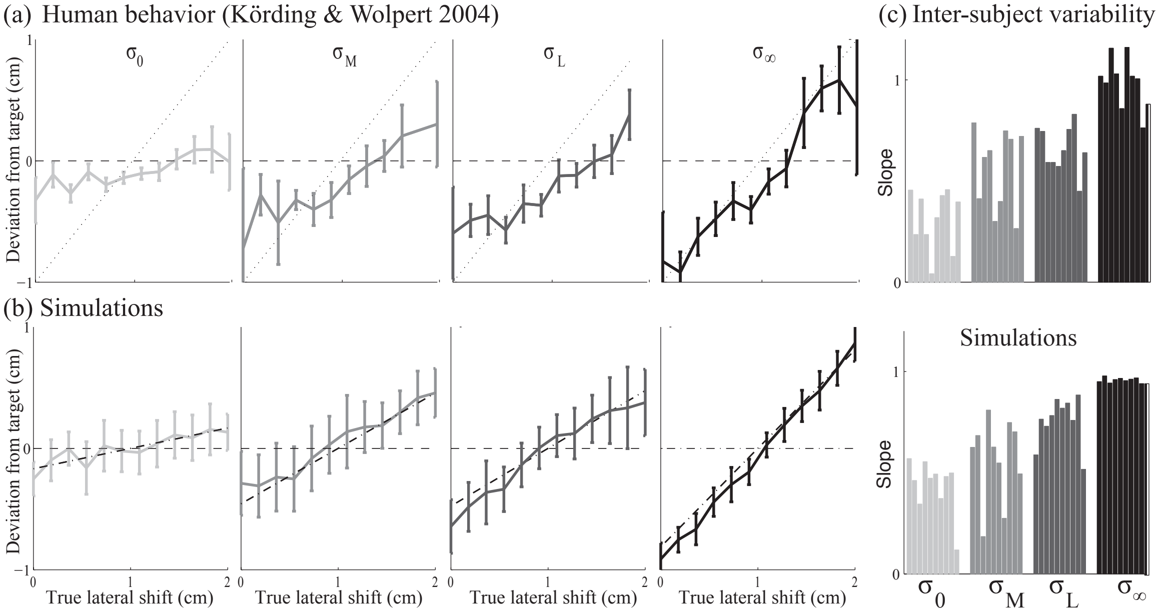


Figure 4.2: Sensorimotor integration using a Gaussian prior with $\mu_x = 1$cm and $\sigma_x = 0.5$cm (data from [Körding and Wolpert, 2004]). (a) Lateral deviation from the target at the end of the trial as a function of the imposed lateral shift, for a typical subject. Curve is the result of averaging over 1000 trials and error bars denote standard error. (b) Simulation using a RBF network with 50 hidden units drawn from the prior. Dash-dot line denotes the theoretical values predicted by Eq. 4.1. (c) The slope for the linear fits are shown for 10 human subjects and 10 simulated subjects whose intrinsic noise levels were drawn from distributions identified in [Körding and Wolpert, 2004].

$f(x_i) = x_i$, approximates the exact Bayesian inference (Eq. 4.1). σ_0, σ_M, and σ_L for each subject were sampled from the distribution over values estimated by [Körding and Wolpert, 2004]. Fig. 4.2b suggests that 50 hidden units are sufficient to account for human performance. Fig. 4.2c shows that the human data and the simulation share the same characteristics that mean slope increases with increasing visual noise.

4.2 Hierarchical Bayes and importance sampling

Inference problems solved by the brain often involve more than one random variable, with complex dependency structures between those variables. For example, visual information processing in primates involves dozens of subcortical areas that interconnect in a hierarchical structure containing two major pathways [Van Essen *et al.*, 1992]. Hierarchical Bayesian inference has been proposed as a solution to this problem [Lee and Mumford, 2003]. However, few studies have proposed neural models that are capable of performing hierarchical Bayesian inference (although see [Friston, 2008]). We show how a multi-layer neural network can perform such computations using importance samplers (Fig. 4.1) as building blocks.

Generative models are widely used in statistics and computer science to describe the causal process by which observed data are generated, assigning a probability distribution to each step in that process [Hinton and Ghahramani, 1997]. To understand brain function, it is often helpful to identify the generative model that determines how stimuli to the brain s are generated (see Fig. 4.3a). The shaded node s represents an observable variable and the open circles (X, Y and Z) represent latent variables. The brain then has to reverse the generative model to recover the latent variables expressed in the data. The direction of inference is thus the opposite of the direction in which the data are generated.

Hierarchical Bayesian inference can be solved by decomposing the hierarchy into

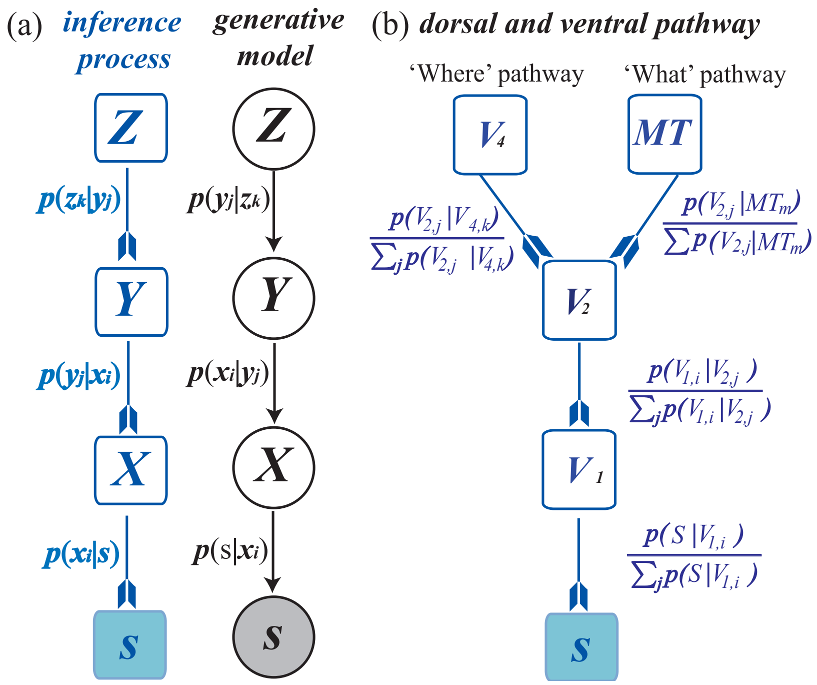

(a) inference process
generative model
Z
Z
p(z_k|y_j)
p(y_j|z_k)
Y
Y
p(y_j|x_i)
p(x_i|y_j)
X
X
p(x_i|s)
p(s|x_i)
s
s
(b) dorsal and ventral pathway
'Where' pathway
'What' pathway
V4
MT
p(V2,j |V4,k) / ∑j p(V2,j |V4,k)
p(V2,j |MTm) / ∑p(V2,j|MTm)
V2
p(V1,i |V2,j) / ∑j p(V1,i |V2,j)
V1
p(S |V1,i) / ∑j p(S |V1,i)
s

Figure 4.3: A hierarchical Bayesian model. (a) The generative model specifies how each variable is generated (in circles), while inference reverses this process (in boxes). s is the stimulus presented to the nervous system, while X, Y, and Z are latent variables at increasing levels of abstraction. (b) Possible implementation in dorsal-ventral visual inference pathways, with multiple higher levels receiving input from one lower level. Note that arrows direct the flow of inference, opposite to that of its generative model.

a sequence of steps that can each be approximated by importance sampling (see Appendix after the chapter). This gives rise to a multi-layer neural network implementation of hierarchial Bayesian inference (Fig. 4.4). The input layer X is similar to that in Fig. 4.1, composed of feature detection neurons with output proportional to the likelihood $p(s|x_i)$. Their output, after presynaptic normalization, is fed into a layer corresponding to the Y variables, with synaptic weights $\frac{p(x_i|y_j)}{\sum_j p(x_i|y_j)}$. The response of neuron y_j, summing over synaptic inputs, approximates $p(y_j|s)$. Similarly, the response of neuron z_k approximates $p(z_k|s)$, and the activities of these neurons are pooled to compute $E[f(z)|s]$. Note that, at each level, x_i,y_j and z_k are sampled from prior distributions. Posterior expectations involving any random variable can be computed because the neuron activities at each level approximate the posterior density. A single pool of neurons can also feed activation to multiple higher levels. Using the visual system as an example (Fig. 4.3b), such a multi-layer importance sampling scheme could be used to account for hierarchical inference in divergent pathways by projecting a set of V2 cells to both MT and V4 areas with corresponding synaptic weights. We evaluate this scheme in two generative models in the following examples.

4.2.1 Example 2: The oblique effect

The oblique effect is a well-established perceptual phenomenon in which people show greater sensitivity to bars with horizontal or vertical ($0^o/90^o$) orientations than "oblique" orientations [Orban *et al.*, 1984]. In this example, we illustrate how to translate an orientation detection task (Fig. 4.5a) into a hierarchical generative model and approximate probabilistic inference in this model by multi-layer importance sampling. We show that the oblique effect can be produced as the direct result of a preferential distribution of orientation selective neurons. In this task [Orban *et al.*, 1984], subjects exhibited higher sensitivity in detecting the direction of rotation of a bar when

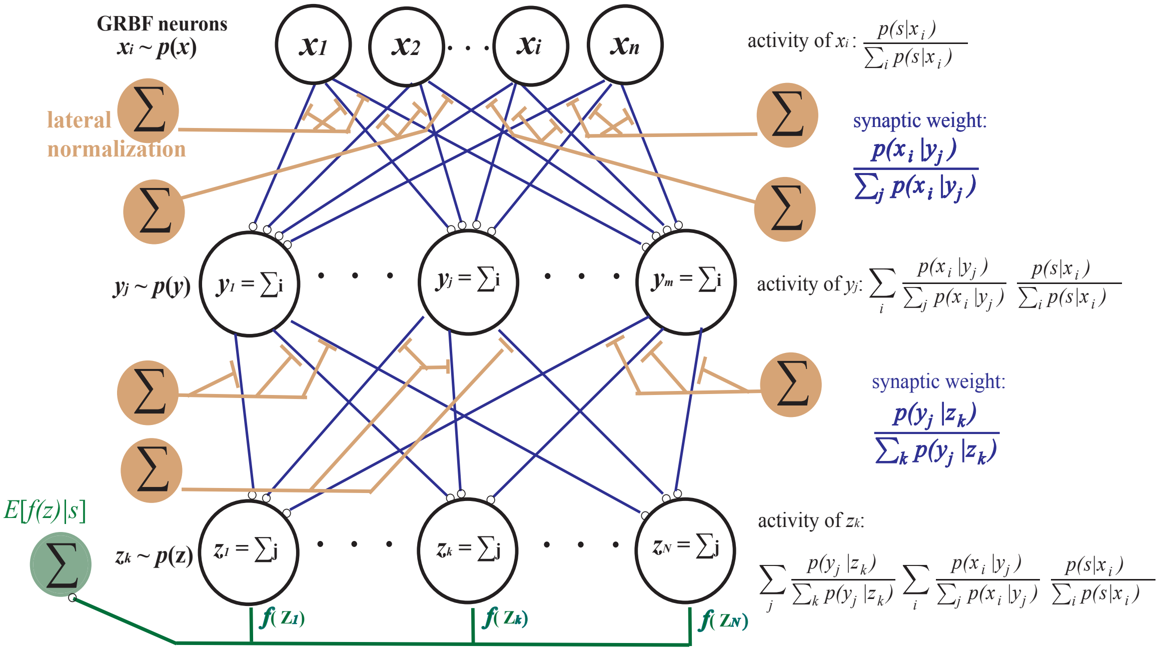

GRBF neurons
$x_i \sim p(x)$
x_1
x_2
x_i
x_n
activity of x_i: $\frac{p(s|x_i)}{\sum_i p(s|x_i)}$
lateral normalization
synaptic weight:
$\frac{p(x_i|y_j)}{\sum_j p(x_i|y_j)}$
$y_j \sim p(y)$
$y_1 = \sum i$
$y_j = \sum i$
$y_m = \sum i$
activity of y_j: $\sum_i \frac{p(x_i|y_j)}{\sum_j p(x_i|y_j)} \frac{p(s|x_i)}{\sum_i p(s|x_i)}$
synaptic weight:
$\frac{p(y_j|z_k)}{\sum_k p(y_j|z_k)}$
$E[f(z)|s]$
$z_k \sim p(z)$
$z_1 = \sum j$
$z_k = \sum j$
$z_N = \sum j$
activity of z_k:
$\sum_j \frac{p(y_j|z_k)}{\sum_k p(y_j|z_k)} \sum_i \frac{p(x_i|y_j)}{\sum_j p(x_i|y_j)} \frac{p(s|x_i)}{\sum_i p(s|x_i)}$
$f(z_1)$
$f(z_k)$
$f(z_N)$

Figure 4.4: Hierarchical Bayesian inference by importance sampling. Multi-layer importance sampler for hierarchical Bayesian inference.

the reference bar to which it was compared was in one of these cardinal orientations (Fig. 4.5a). Fig. 4.5b shows the generative model for this detection problem. The top-level binary variable D randomly chooses a direction of rotation. Conditioning on D, the amplitude of rotation $\Delta\theta$ is generated from a truncated normal distribution ($N_{T(D)}$, being restricted to $\Delta\theta > 0$ if $D = 1$ and $\Delta\theta < 0$ otherwise). When combined with the angle of the reference bar ρ (shaded nodes in the graphical model for observed variables), $\Delta\theta$ generates the orientation of a test bar θ, and θ further generates the observation s, that are both normal distributions with variance σ_θ and σ_s respectively.

The oblique effect has been shown to be closely related to the number of V1 neurons that tuned to different orientations [Orban *et al.*, 1984]. Many studies have found more V1 neurons tuned to cardinal orientations than other orientations [De Valois *et al.*, 1982; Coppola *et al.*, 1998; Furmanski and Engel, 2000]. Moreover, the uneven distribution of feature detection neurons is consistent with the idea that these neurons might be sampled proportional to the prior: more horizontal and vertical segments exist in the natural visual environment of humans.

Importance sampling provides a direct test of the hypothesis that preferential distribution of V1 neurons around $0^o/90^o$ can cause the oblique effect, which becomes a question of whether the oblique effect depends on the use of a prior $p(\theta)$ with this distribution. The quantity of interest for inference is:

$$p(D = 1|s, \rho) \approx \sum_{j'} \sum_{i} \frac{p(\theta_i|\Delta\theta_{j'}, \rho)}{\sum_j p(\theta_i|\Delta\theta_j, \rho)} \frac{p(s|\theta_i)}{\sum_i p(s|\theta_i)}, \tag{4.2}$$

where j' indexes all $\Delta\theta > 0$. If $p(D = 1|s, \rho) > 0.5$, then we should assign $D = 1$. Fig. 4.5c shows that detection sensitivity is uncorrelated with orientations if we take a uniform prior $p(\theta)$, but exhibits the oblique effect under a prior that prefers cardinal directions. In both cases, 40 neurons are used to represent each of $\Delta\theta_i$ and θ_i, and

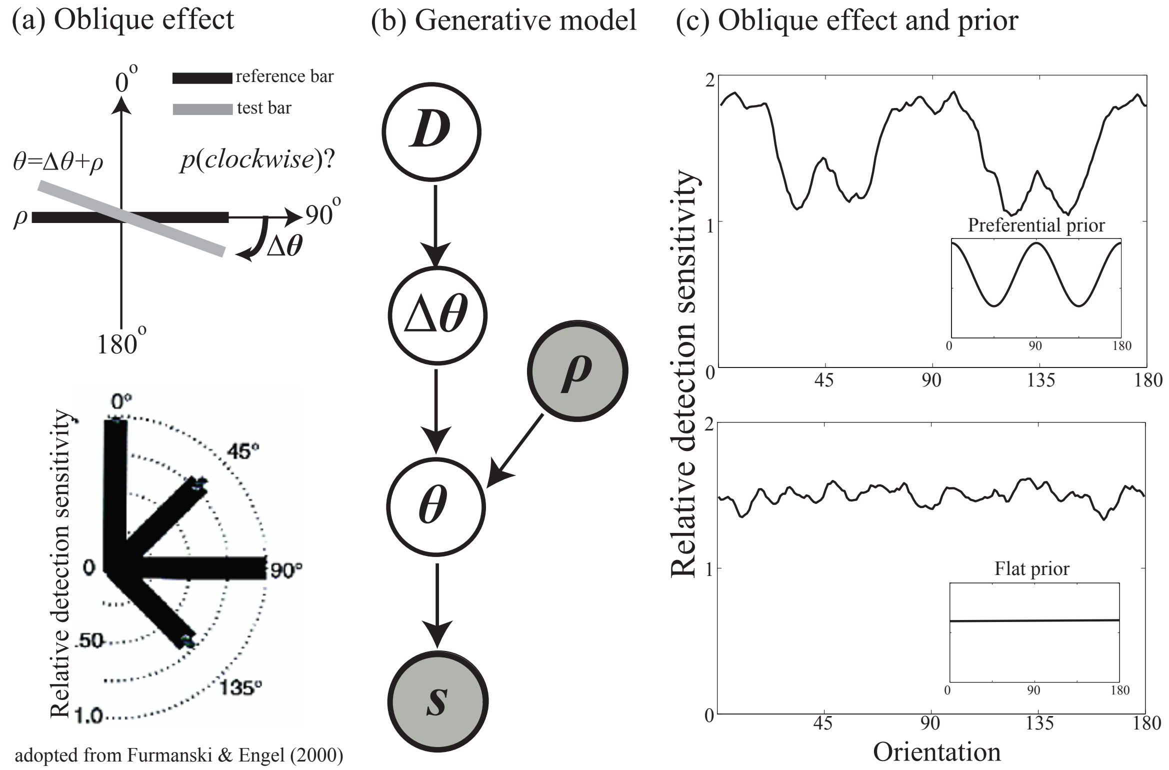


Figure 4.5: The oblique effect. (a) Orientation detection experiment. The oblique effect is shown in lower panel, being greater sensitivity to orientation near the cardinal directions. (b) Generative model. (c) The oblique effect emerges from our model (upper panel), but depends on having the correct prior $p(\theta)$ (inset). A flat prior results in uniform sensitivity (lower panel).

results are averaged over 400 trials. Sensitivity is measured by percentage correct in inference. Due to the qualitative nature of this simulation, model parameters were not tuned to fit experiment data.

4.3 Importance sampling by Poisson spiking neurons

In the neural network models described above, each neuron's output is a continuous signal. However, biological neurons communicate mostly by discrete spikes rather than continuous signals. Although instantaneous firing rate is often used to approximate this continuous signal, this could introduce biases in estimation, or greater variance due to noise in firing rate. Notably, Poisson spiking neurons play an important role in probabilistic models of neural population coding [Ma *et al.*, 2006]. We show that Poisson spiking neurons can perform importance sampling in an unbiased fashion if an ensemble of neurons x_i, drawn from the prior $p(x)$, fire at rates r_i proportional to $p(s|x_i)$ (see Appendix for details). The variance of the estimate is well controlled with a reasonable population firing rate, as we will see in our third example below.

4.3.1 Example 3: Haptic-visual cue combination

Combining information from multiple sensory modalities can be formulated as a problem of Bayesian inference, and approximated by importance sampling. When sensory cues come from multiple modalities, the nervous system is able to combine those cues optimally in the way dictated by Bayesian statistics [Ernst and Banks, 2002]. We use this example to explore the properties of our neural circuit for recursive importance sampling with Poisson spiking neurons. In the experiment, a subject measured the height of a bar through haptic and visual inputs. The object's visual input was ma-

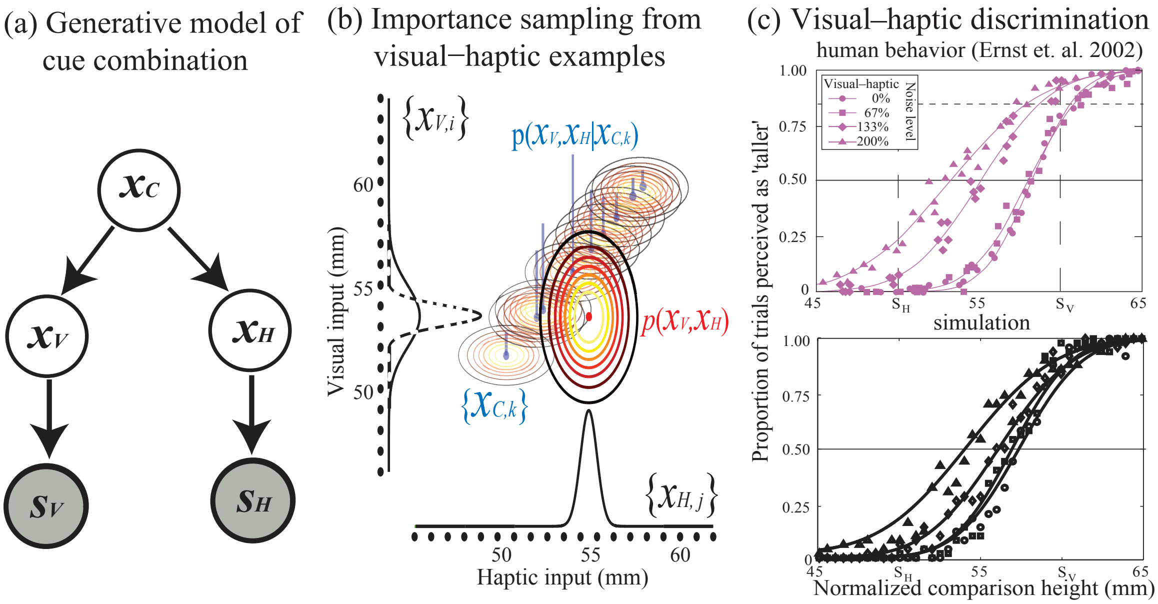


Figure 4.6: Modeling combination of information from multisensory cues. (a) Generative model. s_V and s_H are the sensory stimuli, X_V and X_H the values along the visual and haptic dimensions, and X_C the combined estimate of object height. (b) Illustration of importance sampling using two sensory arrays $\{x_{V,i}\}, \{x_{H,j}\}$. The transparent ellipses indicate the tuning curves of high level neurons centered on values $x_{C,k}$ over x_V and x_H. The big ellipse represents the manipulated input with inconsistent sensory input and different variance structure. Bars at the center of opaque ellipses indicate the relative firing rates of x_C neurons, proportional to $p(x_{C,k}|s_V, s_H)$. (c) Human data and simulation results.

nipulated so that the visual cues can be inconsistent with haptic cues and visual noise can be adjusted to different levels, i.e. the visual cue follows $x_V \sim \mathcal{N}(s_V, \sigma_V^2)$ and the haptic cue follows $x_H \sim \mathcal{N}(s_H, \sigma_H^2)$, where s_V, s_H, σ_V^2 are controlled parameters. The upper panel of Fig. 4.6c shows the percentage of trials that participants report the comparison stimulus (consistent visual/haptic cues from 45-65mm) is larger than the standard stimulus (inconsistent visual/haptic cues, $s_V = 60mm$ and $s_H = 50mm$). With the increase in visual noise, haptic input has greater weight in decision making and therefore responses are shifted towards s_H, consistent with Bayesian statistics.

The importance sampling solution approximates the posterior expectation of the bar's height x_C given s_V and s_H. Sensory inputs are channeled in through x_V and x_H (Fig. 4.6a). Because sensory input varies in a small range (45-65mm in [Ernst and Banks, 2002]), we assume priors $p(x_C)$, $p(x_V)$ and $p(x_H)$ are uniform over this range. To compute the posterior $p(x_V|s_V)$, we use the fact $p(x_V = x_{V,j}|s_V) = E[\mathbf{1}(x_V = x_{V,j})|s_V]$, where $\mathbf{1}(.)$ is an indicator function taking the value 1 when its argument is true (see Appendix for details). A similar method is used to compute $p(x_C|s_V, s_H)$ at the next level of the hierarchy (Fig. 4.6b). Simulation results (for an average of 500 trials) are shown in the lower panel of Fig. 4.6c, compared with human data in the upper panel. There are two free parameters – noise levels σ_V and σ_H – which are optimized to fit within-modality discrimination data (see [Ernst and Banks, 2002] Fig. 3a). The sets of $x_{V,i}$,$x_{H,j}$ and $x_{C,k}$ consist of 20 independently drawn samples each, and the total firing rate of each set of neurons is limited to 30 spikes. The simulations produce a close match to human behavior.

4.4 Discussion

Human perception is often in accord with the optimal solutions produced by Bayesian inference, but the computations required to implement these solutions are challenging. We have shown how Bayesian inference can be approximated using a simple neural circuit that implements a Monte Carlo method known as importance sampling. These circuits can be combined in a modular fashion to permit inferences to be made at multiple levels of abstraction, implementing hierarchical Bayesian inference. They can also be assembled from spiking neurons, and produce predictions that are in accordance with human behavior on several perceptual tasks. In the remainder of the chapter we explore the relationships that this approach holds to other methods for approximating Bayesian inference and proposed neural mechanisms supporting probabilistic computations.

4.4.1 Alternative importance sampling schemes

The importance sampling scheme that we have focused on in this chapter uses the prior as a surrogate for the posterior. The neural circuit implementing this scheme requires the population of neurons to reflect the statistics of the environment in their distribution of preferred stimuli. There are cases where this seems to hold for populations of cortical neurons, providing support for this kind of account, but it is not a necessary requirement in order for some form of importance sampling to be used by the brain. Other distributions of preferred stimuli will correspond to using other surrogate distributions, and can still implement Bayesian inference provided the tuning curves are modified appropriately.

One alternative to requiring strict correspondence to the prior distribution is to use the prior probability of the preferred stimulus to modulate the firing rate of the neuron. The simplest case would use the uniform distribution as a surrogate, with a population of neurons that have preferred stimuli uniformly spanning the space of possible stimuli. In this case, the activation function (or rate function for Poisson spiking neurons) should be proportional to the product of the prior and likelihood, $p(s|x_i)p(x_i)$. The prior probability of a stimulus thus directly influences the firing rate of neurons sensitive to that stimulus. This strategy also seems to be used by the brain: Studies in parietal cortex [Platt and Glimcher, 1999] and superior colliculus [Basso and Wurtz, 1997] show that increased prior probability at a particular location results in stronger firing for neurons with receptive fields at that location.

4.4.2 Connections to particle filtering

The brain's hierarchical structure is instrumental to its ability to carry out complex functions, such as visual perception. This structural attribute has inspired many theoretical proposals about the nature of neural computation [Lee and Mumford, 2003; Friston, 2008]. Notably, [Lee and Mumford, 2003] suggested that the computations performed by the different areas of visual cortex could be modeled as hierarchical Bayesian inference. While they did not propose a detailed neural model, they suggested that this kind of inference could be performed by repeated instances of a module that combined two schemes for probabilistic inference: particle filtering and belief propagation. Particle filtering is a sequential Monte Carlo method based on repeatedly performing importance sampling to update a set of samples ("particles") to correspond to a sequence of distributions [Doucet *et al.*, 2001], and belief propagation is an inference algorithm based on passing messages between units representing the values of random variables [Pearl, 1988].

The framework introduced by Lee and Mumford shares many key elements with our multilayer importance sampler. In our model, feature detection neurons at each level play a similar role to particles in a particle filter, and their outputs are similar to the messages passed from one variable to another in Lee and Mumford's formulation. However, there are three significant differences between our approaches. First, we have identified a specific neural circuit that carries out hierarchical Bayesian inference, while Lee and Mumford focused on the abstract idea that this is the kind of computation that the brain must perform. The results we present in this chpater can thus be viewed as an implementation of Lee and Mumford's proposal. Second, while our approach and particle filters are both based on repeatedly using importance sampling, there is are technical differences in the way that we recursively apply this method. The recursive form of importance sampling that we use was explicitly intended to lend itself to our form of neural implementation, being based on concatenating neural circuits. Finally, in Lee and Mumford's approach, particles generate beliefs using a winner-take-all procedure rather than a weighted averaging strategy. Thus, the distribution of particles does not necessarily approximate the posterior distribution, especially when the density function is multimodal.

Appendix 4.A Recursive importance sampling

In the case of a hierarchical Bayesian model, as shown in Fig. 4.3, the quantity of interest is the posterior expectation of some function $f(z)$ of a high-level latent variable Z given stimulus s, $E[f(z)|s] = \int f(z)p(z|s)\,dz$. By repeatedly using importance sampling (see Eq. 4.3), this hierarchical Bayesian inference problem can decomposed into three importance samplers with values x_i,y_j and z_k drawn from the prior.

$$\begin{aligned}
\boldsymbol{E[f(z)|S_x]} &= \int f(z)\, p(z|y)\boxed{[\int p(y|x)p(x|s)\ dx]}\,dy\ dz \\
& \qquad\qquad\qquad\qquad \text{importance sampling} \qquad x_i \sim p(x) \\
&\approx \int f(z)\, p(z|y)\ \frac{\sum_i p(y|x_i)p(s|x_i)}{\sum_i p(s|x_i)}\ dy\,dz \\
&= \int f(z)\ \frac{\sum_i \boxed{[\int p(z|y)p(y|x_i)\ dy]}\ p(s|x_i)}{\sum_i p(s|x_i)}\ dz \\
& \qquad\qquad\qquad\qquad \text{importance sampling} \qquad y_j \sim p(y) \\
&\approx \int f(z) \sum_i \frac{\sum_j p(z|y_j)p(x_i|y_j)}{\sum_j p(x_i|y_j)}\ \frac{p(s|x_i)}{\sum_i p(s|x_i)}\ dz \\
&= \sum_j \boxed{[\int f(z)p(z|y_j)\ dz]} \sum_i \frac{p(x_i|y_j)}{\sum_j p(x_i|y_j)}\ \frac{p(s|x_i)}{\sum_i p(s|x_i)} \\
& \qquad\qquad \text{importance sampling} \qquad\qquad\qquad z_k \sim p(z) \\
&\approx \sum_j\ \frac{\sum_k f(z_k)p(y_j|z_k)}{\sum_k p(y_j|z_k)}\ \sum_i \frac{p(x_i|y_j)}{\sum_j p(x_i|y_j)}\ \frac{p(s|x_i)}{\sum_i p(s|x_i)} \\
&= \sum_k f(z_k) \underbrace{\sum_j\ \frac{p(y_j|z_k)}{\sum_k p(y_j|z_k)}\ \underbrace{\sum_i \frac{p(x_i|y_j)}{\sum_j p(x_i|y_j)}\ \underbrace{\frac{p(s|x_i)}{\sum_i p(s|x_i)}}_{\boldsymbol{x_i}}}_{\boldsymbol{y_j}}}_{\boldsymbol{z_k}}
\end{aligned} \tag{4.3}$$

This result relies on recursively applying importance sampling to the integral, with each recursion resulting in an approximation to the posterior distribution of another random variable. This recursive importance sampling scheme can be used in a variety of generative models where there is a dependency between a sequence of random variables. For example, tracking a stimulus over time is a natural extension where an additional observation is added at each level of the generative model.

Appendix 4.B Poisson spiking neurons

To show how Poisson spiking neurons can perform importance sampling, we use a property of Poisson distributions: If $n_i \sim \text{Poisson}(r_i)$ is the number of spikes produced by neuron x_i in a given time period and $N = \sum_i n_i$, then $N \sim \text{Poisson}(\sum_i r_i)$ and $(n_1, n_2, \ldots, n_m|N) \sim \text{Multinomial}(N, r_i/(\sum_i r_i))$. This implies that $E(n_i/N|N) = r_i/(\sum_i r_i)$. Assuming a Poisson neuron in the hidden layer that is tuned to stimulus x_i emits n_i spikes with an underlying rate proportional to $p(s|x_i)$, the output neuron in the network shown in Fig. 4.1 of the main text computes $\sum_i f(x_i)n_i/\sum_i n_i$, which is the weighted average of a function $f(x_i)$ using weights n_i. Taking the expectation of this quantity we get

$$\begin{aligned} E\left[\sum_i f(x_i)\frac{r_i}{\sum_j r_j}\right] &= \sum_i f(x_i) E\left[\frac{r_i}{\sum_j r_j}\right] = \sum_i f(x_i)\frac{c\ r_i}{\sum_j c\ r_j} \\ &= \frac{\sum_i f(x_i)p(s|x_i)}{\sum_i p(s|x_i)} \approx E[f(x)|s]. \end{aligned} \tag{4.4}$$

Therefore, the response of the output neuron is an unbiased estimate of the importance sampling approximation to the posterior expectation. The variance of this estimator decreases as population activity $N = \sum_i n_i$ increases because $\text{var}[n_i/N] \sim 1/N$. Thus, Poisson spiking neurons, if plugged into an RBF network, can perform importance sampling and give similar results to “neurons” with continuous output.

Appendix 4.C Cue combination

The importance sampling solution approximates the posterior expectation of the bar’s height x_C given s_V and s_H through visual and haptic pathways x_V and x_H, respectively. Thus the posterior density become a posterior expectation, using importance sampling:

$$p(x_V = x_{V,j}|s_V) = E[\mathbf{1}(x_V = x_{V,j})|s_V] \approx \frac{p(s_V|x_{V,j})}{\sum_i p(s_V|x_{V,i})} \qquad x_{V,i} \sim p(x_V). \tag{4.5}$$

Assuming that $x_{V,i}$ are Poisson neurons and emit $n_{V,i}$ spikes ($n_{V,i} \sim$ Poisson$[c \cdot p(s_V|x_{V,i})]$), then $p(x_V = x_{V,j}|s_V) \approx \frac{n_{V,j}}{\sum_i n_{V,i}}$. A similar strategy applies to $p(x_H|s_H)$. The posterior $p(x_C|s_V, s_H)$, however, is not trivial since multiplication of spike trains is needed:

$$\begin{aligned} p(x_C = x_{C,k}|s_V, s_H) &= \int \mathbf{1}(x_C = x_{C,k}) p(x_C|x_V, x_H) p(x_V|s_V) p(x_H|s_H)\ dx_V\ dx_H \\ &\approx \sum_i \sum_j p(x_{C,k}|x_{V,i}, x_{H,j}) \frac{n_{V,i}}{\sum_i n_{V,i}} \frac{n_{H,j}}{\sum_j n_{H,j}}. \end{aligned} \tag{4.6}$$

Fortunately, the experiment gives an important constraint, namely subjects were not aware of the manipulation of visual input. Thus, the values $x_{C,k}$ employed in the computation are sampled from normal perceptual conditions, namely consistent visual and haptic inputs ($x_V = x_H$) and normal variance structure (transparent ellipses in Fig. 4.6c of the main text, on the diagonal). Therefore, the random variables $\{x_V, x_H\}$ effectively become one variable $x_{V,H}$ and values of $x_{V,H,i}$ are composed of samples drawn from x_V and x_H independently. Applying importance sampling,

$$p(x_C = x_{C,k}|s_V, s_H) \approx \frac{\sum_i p(x_{V,i}|x_{C,k})n_{V,i} + \sum_j p(x_{H,j}|x_{C,k})n_{H,j}}{\sum_i n_{V,i} + \sum_j n_{H,j}}. \tag{4.7}$$

Neuron $x_{C,k}$ combines cues from $\{x_{V,i}\}$ and $\{x_{H,j}\}$ neurons, with synaptic strengths $p(x_{V,i}|x_{C,k})$ and $p(x_{H,j}|x_{C,k})$ respectively, and is normalized by the activities of $\{x_{V,i}\}$ and $\{x_{H,j}\}$. Thus, the statistically optimal estimate based on visual and haptic cues is the posterior expectation

$$E[x_C|s_V, s_H] \approx \frac{\sum_k x_{C,k} \cdot n_{C,k}}{\sum_k n_{C,k}}, \tag{4.8}$$

where $n_{C,k} \sim \text{Poisson}(c \cdot p(x_{C,k}|s_V, s_H))$ (c is a positive constant) and $x_{C,k} \sim p(x_C)$.

Chapter 5

Neural implementation of sequential Bayesian inference

The nervous system excels in dynamical tasks such as visual tracking and motor control. In these tasks, sensory input is constantly fed in at each time step, requiring repeated updating of beliefs. Understanding how the brain processes dynamical data of this kind is a challenge for both experimental and theoretical neuroscientists [Friston, 2008; Barbieri *et al.*, 2004]. Particle filtering has been proposed as a way of modeling neural dynamics [Friston, 2008; Lee and Mumford, 2003], and has been successfully applied to high dimensional problems such as computer vision [Isard, 2003]. Most applications of particle filtering focus on modeling probability distributions that change through time, rather than the neural implementation of these algorithms given the constraints and features of the biological system.

In this chapter, we first formulate the problem of sequential estimation. Particle filtering is then introduced as a general algorithm to solve this problem. The connection between particle filters and recursive importance sampling 4 is established and we compare the performance of these algorithms. This connection suggests that

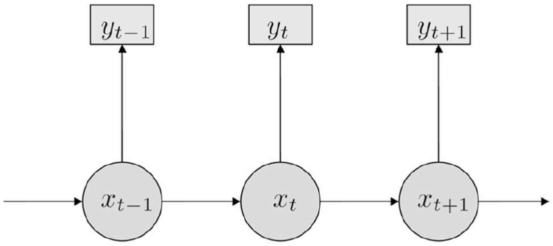


Figure 5.1: A diagram of sequential estimation, $\mathbf{y_{0:t}}$ are observable and $\mathbf{x_{0:t}}$ are hidden.

the recursive importance sampling approach is likely to be applicable to problems that require updating distributions over time, such as planning and executing motor movements. We then introduce the neural network model and detailed neural implementation of recursive importance sampling. Its similarity to cerebellar circuits indicates that cerebellum may be on neural substrate appropriate for implementing these kinds of circuits.

5.1 The sequential estimation problem

Sequential estimation, which requires estimating the state of the world while incorporating feedback signals over time, plays a significant role in human behavior. For example, in reaching a target, an agent constantly estimates the 'best movement' based on the current position of the hand, feedback signals provided by visual input. The sequential estimation can be described by a probabilistic model as following.

Formally, the problem is estimating the value of a hidden variable x_t at each time step t based on the noisy sensory information y_t available (see Fig. 5.1). The transition probability from x_{t-1} to x_t is $p(x_t|x_{t-1})$, and the likelihood function of observing sensory input y_t given hidden state x_t is $p(y_t|x_t)$. The collection of x_t and y_t over time are denoted as $\mathbf{x_{0:t}} = [x_1, x_2, \cdots x_t]$ and $\mathbf{y_{0:t}} = [y_1, y_2, \cdots y_t]$. The joint distribution of hidden variables and sensory inputs is

$$p(\mathbf{x_{0:t}}, \mathbf{y_{0:t}}) = p(x_0)p(y_0|x_0)\Pi_{i=1}^{t} p(x_i|x_{i-1})p(y_i|x_i) \tag{5.1}$$

and the posterior distribution of x_t given sensory inputs $y_{0:t}$ has the iterative form

$$p(x_t|\mathbf{x_{0:t-1}}, \mathbf{y_{0:t}}) \propto p(x_{t-1}|\mathbf{x_{0:t-2}}, \mathbf{y_{0:t-1}}) \cdot p(x_t|x_{t-1})p(y_t|x_t) \tag{5.2}$$

In a simple special case, this estimation problem has an analytical solution minimizing the squared error, known as Kalman Filter [Welch and Bishop, 1995]. This requires that the transition of hidden states $x_{t-1} \rightarrow x_t$ as well as the generation of y_t follows a linear transformation with Gaussian noise, i.e.

$$x_t = F_k x_{t-1} + w_k \tag{5.3}$$

$$y_t = G_k x_t + v_k \tag{5.4}$$

where w_k and v_k are random variables following a Gaussian distribution, F_k and G_k are matrices.

5.2 Particle filtering as a general solution to sequential estimation

Kalman filtering has only limited success when applied to systems with highly nonlinear dynamics and non-Gaussian noise [Arulampalam *et al.*, 2001]. In general, computing $p(x_t|x_{0:t-1}, y_{0:t})$ given arbitrary transition and noise functions is analytically intractable. Particle filtering, a sequential Monte Carlo method, approximates $p(x_t|\mathbf{x_{0:t-1}}, \mathbf{y_{0:t}})$ by updating a set of samples (or particles) to correspond to a sequence of distributions [Doucet *et al.*, 2001]. Particle filtering uses a group of particles $x_{1:t}^{(i)}$ and associated weights $w_t^{(i)}$ to approximate the posterior distribution, i.e.

$$p(x_{t-1}|\mathbf{x_{0:t-2}}, \mathbf{y_{0:t-1}}) \approx \sum_i w_{t-1}^{(i)} \delta(x_{t-1} - x_{t-1}^{(i)}). \tag{5.5}$$

Using sequential-importance-sampling (SIS), new particles are generated by a proposal distribution $x_t^{(i)} \sim q(x_t|x_{t-1}^{(i)})$. According to the updating rule (Eq. 5.2), the new posterior $p(x_t|\mathbf{x_{0:t-1}}, \mathbf{y_{0:t}})$ can be approximated by the new set of particles $x_t^{(i)}$ with weights $w_t^{(i)}$:

$$\begin{aligned} p(x_t|\mathbf{x_{0:t-1}}, \mathbf{y_{0:t}}) &\approx \sum_i w_t^{(i)} \delta(x_t - x_t^{(i)}) \\ w_t^{(i)} &\propto \frac{p(y_t|x_t^{(i)}) p(x_t^{(i)}|x_{t-1}^{(i)})}{q(x_t^{(i)}|x_{t-1}^{(i)})} w_{t-1}^{(i)} \end{aligned} \tag{5.6}$$

5.3 Connecting recursive importance sampling and particle filtering

The results in the previous chapter suggest that a recursive structure using importance sampling as building blocks provides an efficient algorithm for hierarchical Bayesian inference (Fig. 5.2). This recursive importance sampling (RIS) circuit uses Poisson spiking neurons to form flexible information flow mimicking the brain. And the model predicts human behavior in many perception tasks. Applying the chain rule,

$$p(x_2|x_1, x_0, s) \propto p(x_1|x_0, s) p(x_2|x_1) \tag{5.7}$$

Let $\{x_0^{(i)}\}$, $\{x_1^{(j)}\}$, $\{x_2^{(k)}\}$ be sets of particles sampled from proposal distribution $q(x_0|s)$,$q(x_1|x_0)$ and $q(x_2|x_1)$ respectively, where particles $\{x_t^{(i)}\} = [x_t^{(1)}, x_t^{(2)}, \cdots x_t^{(i)}, \cdots, x_t^{(N)}]$.

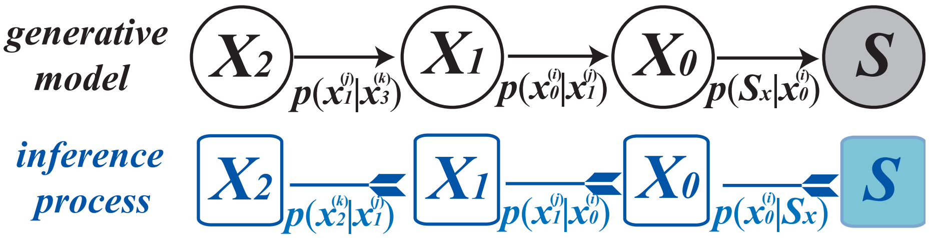


Figure 5.2: A Hierarchical Bayesian inference structure by recursive importance sampling. S are sensory inputs that are known. The inference process is comparable to the temporal dynamics of hidden variables shown in Fig.5.1.

The posterior distribution of x_0 given sensory input s, $p(x_0|s)$ can be approximated by a group of $\{x_0^{(i)}\}$ with importance weights $v_0^{(i)}$:

$$\begin{aligned} p(x_0|s) &= \int \delta(x_0 - x')p(x'|s)dx' \approx \sum_i v_0^{(i)} \delta(x_0 - x_0^{(i)}) \qquad\qquad x_0^{(i)} \sim p(x_0|s) \\ v_0^{(i)} &\propto \frac{p(x_0^{(i)}|s)}{q(x_0^{(i)}|s)}, \end{aligned} \tag{5.8}$$

where $\delta(\cdot)$ is a delta function. Similarly, $p(x_1|s)$ and $p(x_2|s)$ can be approximated by a group of $\{x_1^{(j)}\}, \{x_2^{(k)}\}$ with importance weights $v_1^{(j)}, v_2^{(k)}$:

$$\begin{aligned} p(x_1|s) &\approx \sum_j v_1^{(j)} \delta(x_1 - x_1^{(j)}), \\ v_1^{(j)} &\propto \sum_i \frac{p(x_1^{(j)}|x_0^{(i)})}{q(x_1^{(j)}|x_0^{(i)})} v_0^{(i)}; \\ p(x_2|s) &\approx \sum_k v_2^{(k)} \delta(x_2 - x_2^{(k)}), \\ v_2^{(k)} &\propto \sum_j \frac{p(x_2^{(k)}|x_1^{(j)})}{q(x_2^{(k)}|x_1^{(j)})} v_1^{(j)}. \end{aligned} \tag{5.9}$$

In fact, recursive importance sampling provides an alternative connection pattern for sequential Monte Carlo and constitute a neurally plausible mechanism for implementing sequential estimation. This opens up the question of relative performance of

RIS and existing particle filtering algorithms, which is tested in later section. Comparing Eq. 5.7 and Eq. 5.2, we find that, for sequential estimation as in Fig.5.1, the effect of sensory inputs $\mathbf{y_{0:t}}$ should be added to the weight updating rule:

$$v_t^{(j)} \propto \sum_i \frac{p(y_t|x_t^{(j)})p(x_t^{(j)}|x_{t-1}^{(i)})}{q(x_t^{(j)}|x_{t-1}^{(i)})} v_{t-1}^{(i)}. \tag{5.10}$$

The Appendix shows the detailed derivation of the updating rule.

The connection between Eq. 5.6 and Eq. 5.10 suggests that recursive importance sampling and particle filtering both represent posterior density by particles and follow similar updating rules except the cross-connections in recursive importance sampling model. In particle filters, one particle $x_t^{(j)}$ only receives contributions from one previous particle $x_{t-1}^{(j)}$; while in recursive importance sampling, one particle $x_t^{(j)}$ receives contributions from multiple particles $x_{t-1}^{(i)}, \{i = 1, 2, \cdots\}$ in the previous time step. Therefore, we call recursive importance sampling for the **cross-connected sequential importance sampling** (or CC-SIS).

5.4 Performance comparison

We compared the performance of various versions of particle filtering and cross-connected algorithms. The sequential estimation problem and particle filtering algorithms used here are widely applied in many studies [Arulampalam *et al.*, 2001; Gordon *et al.*, 1993; Kitagawa, 1996].

Problem setup. We consider the following generative model:

$$\begin{aligned} x_t &= f(x_{t-1}, t) + \epsilon_t & (5.11) \\ y_t &= g(x_t) + \lambda_t & (5.12) \end{aligned}$$

where

$$
\begin{aligned}
f(x_{t-1}, t) &= \frac{x_{t-1}}{2} + \frac{25x_{t-1}}{1 + x_{t-1}^2} + 8cos(1.2t) && (5.13)\\
g(x_t) &= \frac{x_t^3}{1000} && (5.14)\\
p(\epsilon_t) &= \frac{2}{\mu} exp(-\frac{|\epsilon_t|}{\mu}) && (5.15)\\
p(\lambda_t) &= 0.995\mathbf{Unif}[-1, 1] + 0.005\mathbf{Unif}[-20, 20] && (5.16)
\end{aligned}
$$

This is a nonlinear, time-dependent estimation problem with non-Gaussian noise. The problem cannot be solved exactly using a Kalman filter but can be approximated. The particle filtering algorithms [Arulampalam *et al.*, 2001] are applied in four versions. They differ in 1) the resampling procedure; 2) the proposal distribution $q(x_t|x_{t-1}, y_t)$ and the corresponding weight-updating rules. A practical issue in implementing particle filters is degeneration of the particle population. That is, if we simply update particles according to Eq. 5.6, diversity among the population is lost and weights are highly concentrated in one or few particles over time. Resampling is a remedy to rejuvenate the particle population by sampling the current pool of particles and assigning them equal weights (for details, see [Arulampalam *et al.*, 2001]). The efficiency of the particle filter also depends on the choice of proposal distributions. A good proposal distribution generates new particles in regions of high posterior density as well as keep a balance on particle diversity. Proposal distributions range from some fixed density function (e.g., the prior) to some statistically optimal proposal (e.g., [Khan *et al.*, 2004]). The optimal proposal distribution is difficult to compute since itself involves some posterior distribution that is often intractable. In the following simulations, we consider two common choices of proposals: the prior distribution and transition probability $p(x_t) \sim p(x_t^{(i)}|x_{t-1}^{(i)})$.

- **Sequential importance sampling (SIS)**

Proposal distribution: $p(x_t) \sim q(x)$, where $q(x)$ is the Gaussian approximation of prior $p(x)$. $p(x)$ is the marginal distribution of $\mathbf{x_{0:t}}$ over time.

Weight updating:

$$w_t^{(i)} \propto \frac{p(y_t|x_t^{(i)})p(x_t^{(i)}|x_{t-1}^{(i)})}{q(x_t^{(i)})} w_{t-1}^{(i)} \tag{5.17}$$

Theoretically, normalization is not necessary at each step until the end. However, to avoid numerical instability, weights should be normalized to a constant at each step, i.e. $w_t^{(i)} = \frac{w_t^{(i)}}{\sum_i w_t^{(i)}}$. This principle is also applied in other algorithms.

No resampling.

- **Sampling importance resampling (SIR)**

  Proposal distribution: $p(x_t) \sim q(x)$, where $q(x)$ is the Gaussian approximation of prior $p(x)$.

  Weights are updated as in Eq. 5.17.

  Resampling: once effective number of particles $N_{eff} < N_{thres} = \frac{N}{10}$, resampling according to algorithm 2 in [Arulampalam *et al.*, 2001], where N is the total number of particles and

$$N_{eff} = \frac{1}{\sum_{i=1}(w_t^{(i)})^2}. \tag{5.18}$$

- **Sampling importance resampling from transition probability (SIR-TP)**

  Proposal distribution: $p(x_t) \sim p(x_t^{(i)}|x_{t-1}^{(i)})$, i.e. the transition probability.

  Weight updating:

$$w_t^{(i)} \propto p(y_t|x_t^{(i)})w_{t-1}^{(i)} \tag{5.19}$$

  Resampling: once effective number of particles $N_{eff} < \frac{N}{10}$, resampling according to algorithm 2 in [Arulampalam *et al.*, 2001].

- **Cross-connected sequential importance sampling (CC)**

  Proposal distribution: $p(x_t) \sim q(x)$, where $q(x)$ is the Gaussian approximation of prior $p(x)$.

  Weight updating:

  $$w_t^{(j)} \propto \sum_i \frac{p(y_t|x_t^{(j)})p(x_t^{(j)}|x_{t-1}^{(i)})}{q(x_t^{(j)})} w_{t-1}^{(i)}. \tag{5.20}$$

  No resampling.

- **Cross-connected sequential importance sampling using fixed set of particles (CC-fixed)**

  Proposal: a fixed set of $\{x^{(i)}\}$ sampled from $q(x)$ are used repeatedly at every time t. This is motivated by the fact that the biological properties of neural circuits is stable in short term and therefore might requires a set of pre-fixed samples used throughout the computation.

  Weight updating: as in Eq. 5.20.

  No resampling.

- **Sampling importance cross-connected resampling (SIR-CC)**

  Proposal distribution: $p(x_t) \sim q(x)$, where $q(x)$ is the Gaussian approximation of prior $p(x)$.

  Weights are updated as in Eq. 5.17.

  Resampling: once effective number of particles $N_{eff} < \frac{N}{10}$, resample $x_t^{(j)} \sim q(x)$ and assign weights according to updating rule Eq. 5.20.

The performance of each algorithm is measured by:

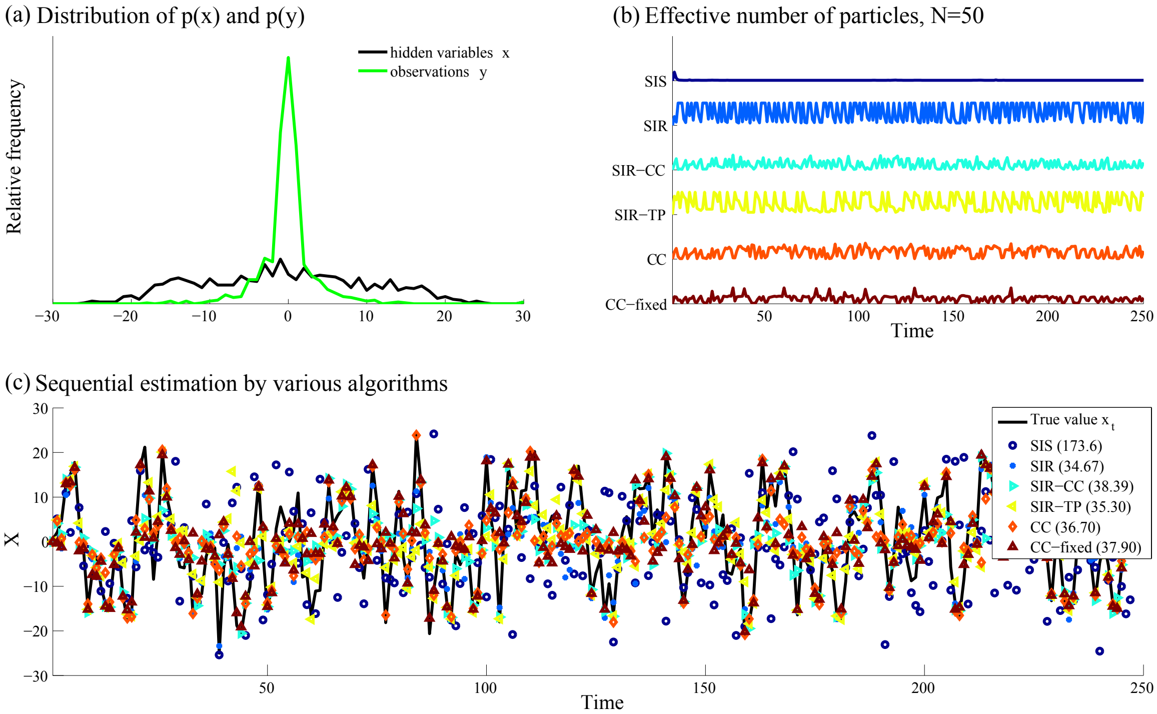
- **Mean square error** measures deviation of estimation from true value, defined as

$$err_w = \frac{1}{T}\sum_{t=1}^{T}(x_t - \sum_i w_t^{(i)} x_t^{(i)})^2 \tag{5.21}$$

- **Relative effective number of particles** indicates the effectiveness of using large number of particles, defined as $\widetilde{N}_{eff} = \frac{1}{T}\sum_{t=1}^{T}\frac{1}{N}N_{eff}(t)$. Note that comparing $\widetilde{N}_{eff}$ of algorithms with resampling and without resampling might not be meaningful since resampling generates many identical particles.

- **Survival rate** measures the robustness of the algorithm. In case of rare events with extreme value of y_t, the probability $p(y_t|x_t^{(i)})$ becomes diminishingly small for all particles and the algorithm aborted because of numerical instability. If estimation is repeated independently for multiple trials, survival rate $R_{surv}(t)$ is the percentage of trials survived at time t.

The marginal distribution of the hidden variable $\mathbf{x_{0:t}}$, fits closely to a Gaussian distribution (Fig. 5.3 (a)). The Gaussian fit $q(x)$ is used as a proposal distribution for SIS, SIR, SIR-CC, CC and CC-fixed algorithms. This choice is due to some considerations of neural plausibility discussed in detail in next section. All algorithms, except SIS, can track $\mathbf{x_{0:t}}$ well (Fig. 5.3 (c)). Resampling is frequently performed in SIR, SIR-CC and SIR-TP (Fig. 5.3 (c)). N_{eff} jumps to its maximum after resampling in SIR and SIR-TP, but it is worthwhile to note that the resampled set contains many identical particles. Meanwhile, cross-connected algorithms (CC and CC-fixed) see constant rejuvenation of particle sets without resampling. Although their N_{eff} does not recover to 100% level, there is little probability of having two identical particles in the set. Therefore, the cross-connected algorithm outperforms SIR algorithms (Fig. 5.4 (a)) for small number of particles ($N < 100$) despite inferior N_{eff} (Fig. 5.4 (c)). The same trends exist in survival rate (Fig. 5.4 (d)). For N=10, SIR algorithms are



(a) Distribution of p(x) and p(y)

(b) Effective number of particles, N=50

(c) Sequential estimation by various algorithms

Figure 5.3: Sequential estimation by particle filtering and cross-connected algorithms. (a) Distribution of hidden variables $\mathbf{x_{0:t}}$ and noisy sensory inputs $\mathbf{y_{0:t}}$ over time. (b) Temporal dynamics of effective number of particles N_{eff} defined as in Eq. 5.18. N_{eff} in SIS depletes quickly. For SIR, SIR-IC and SIR-TP, every jump in N_{eff} suggests a resampling operation. For CC and CC-fixed, N_{eff} rejuvenate without the help of resampling. (c) x_t and the estimation by various algorithms. In legend, numbers in parentheses are mean square error.

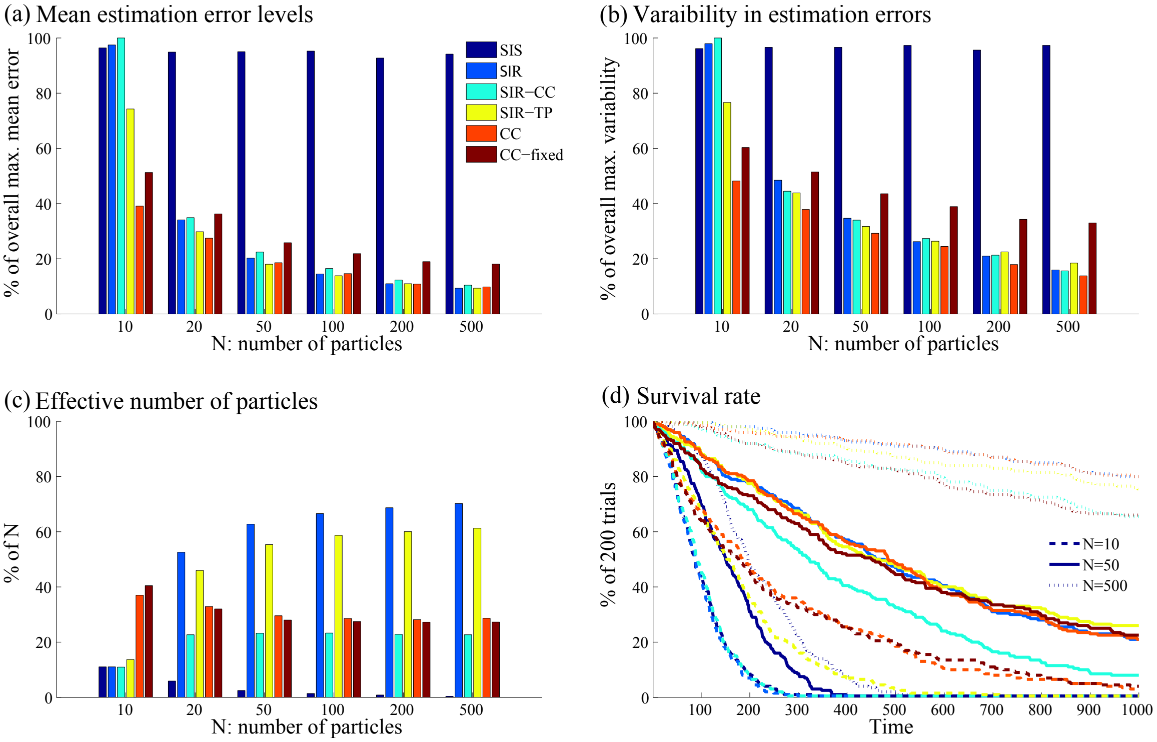


Figure 5.4: Statistics of algorithms for $N = \{10, 20, 50, 100, 200, 500\}$ of 200 trials. In general, performance improves with more particles. (a) Mean estimation error as defined in Eq. 5.21. (b) Variability of mean error over trials. (c) Mean N_{eff} as percentage of N. (c) Survival rate $R_{surv}(t)$ for $N = \{10, 50, 500\}$.

much more vulnerable than CC algorithms. For $N > 50$, survival rate are comparable and improve significantly with the increase of N.

5.5 Neural implementation of sequential estimation

The biological properties of the brain put constraints on the implementation of sequential estimation. These constraints favor cross-connected structures over traditional particle filters.

Cortical neurons are activated by certain stimulus patterns and their firing rates

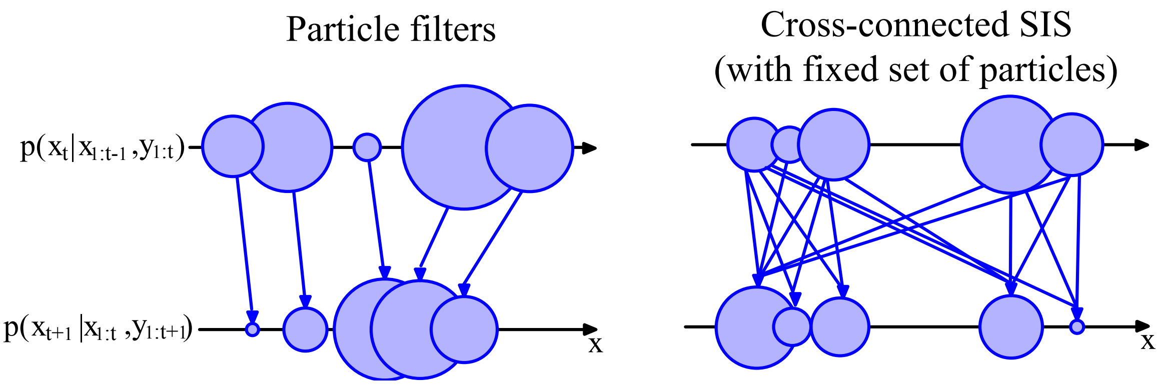


Figure 5.5: Diagram of particle filter (SIS, SIR) and cross-connected sequential importance sampling (CC). Circles are positioned at $x_{t-1}^{(i)}$ or $x_t^{(i)}$ and the sizes represent the weights $w_{t-1}^{(i)}$ or $w_t^{(i)}$.

are highly correlated to the strength of these stimuli. Therefore, it is natural to assume that particles are represented by a group of neurons, whose preferred stimulus patterns are located at $\{x_t^{(i)}\}$, and the particles' weights $\{w_t^{(i)}\}$ are proportional to the neurons' firing rate.

Cortical neurons are highly interconnected with each other. However, in traditional particle filtering algorithms such as SIS and SIR, every particle is connected only to its own (often sole) offspring, does not exhibits such a cross-connected structure. In contrast, in cross-connected sequential importance sampling, neurons at the previous stage $\{x_{t-1}^{(i)}\}$ are fully connected to neurons in the later stage $\{x_t^{(i)}\}$ (see Fig.5.5).

The biological properties of cortical neurons and their connections are relatively stable in the period of seconds to minutes. Since $\{x_t^{(i)}\}$ are related to the internal properties of neurons, traditional particle filtering algorithms is impractical due to its requirement of realtime conditional sampling and resampling $\{x_t^{(i)}\}$. Rather, it should be assumed that particles are time-invariant in the period of an estimation task. Again, this constraint fits the diagram of cross-connected sequential importance sampling. Moreover, it is desirable to use the same group of particles over time (corresponding to CC-fixed case) since this lifts the burden of recruiting ever more neurons as T increases. Note that in the previous recursive importance sampling scheme for hierarchical Bayesian inference, feature detection neurons are also a fixed population, sampled from prior distribution.

In the following sections, we will first build a neural network model of cross-connected sequential importance sampling. Then, we will discuss issues in its neural implementation including normalization, multiplication and particle recycling (i.e. using a same set of particles repeatedly). This leads us to a hypothesis that the cerebellum provides an ideal circuit layout for CC-SIS.

5.5.1 Neural network model

Fig. 5.6 shows a neural network structure implementing the CC-SIS algorithm. Assume a group of feature detection neurons with preferred stimuli at $\{x_t^{(i)}\}$ (sampled from $q(x)$). At time t, their activities are proportional to $\{w_t^{(i)}\}$ up to a constant (due to lateral normalization). At time $t+1$, y_{t+1} and $\{x_t^{(i)}\}$ neurons make multiplicative synapses with synaptic weights $p(y_{t+1}|x_{t+1}^{(j)})$ and $\frac{p(x_{t+1}^{(j)}|x_t^{(i)})}{q(x_{t+1}^{(j)})}$, respectively. The outputs of multipliers are pooled to next layer feature detection neurons indexed by j. These neurons have preferred stimuli at $\{x_{t+1}^{(j)}\}$ and are activated proportional to $\{w_{t+1}^{(j)}\}$ according to weight updating rule Eq. 5.30.

5.5.2 Detailed considerations in neural implementation

Fig. 5.6 provides a schema for neural implementation of sequential estimation. However, this requires more consideration to identify plausible neural mechanisms for each operation of the model.

First, sensory input by a single neuron is oversimplified. Sensory input, often

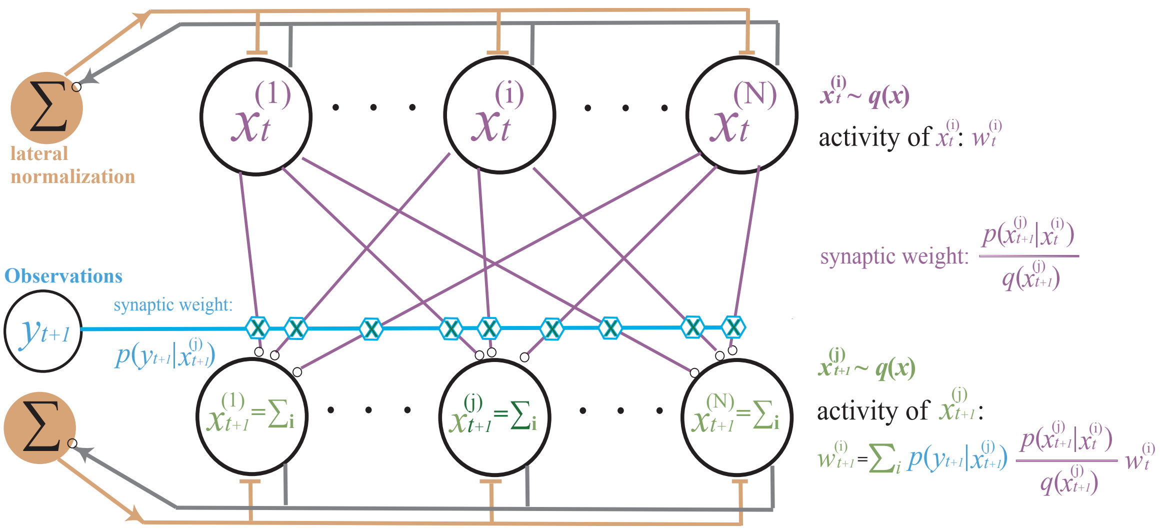

Σ
lateral
normalization
$x_t^{(1)}$
$x_t^{(i)}$
$x_t^{(N)}$
$x_t^{(i)} \sim q(x)$
activity of $x_t^{(i)}$: $w_t^{(i)}$
synaptic weight: $\frac{p(x_{t+1}^{(j)}|x_t^{(i)})}{q(x_{t+1}^{(j)})}$
Observations
y_{t+1}
synaptic weight:
$p(y_{t+1}|x_{t+1}^{(j)})$
Σ
$x_{t+1}^{(1)}=\sum_i$
$x_{t+1}^{(j)}=\sum_i$
$x_{t+1}^{(N)}=\sum_i$
$x_{t+1}^{(j)} \sim q(x)$
activity of $x_{t+1}^{(j)}$:
$w_{t+1}^{(i)}=\sum_i p(y_{t+1}|x_{t+1}^{(j)}) \frac{p(x_{t+1}^{(j)}|x_t^{(i)})}{q(x_{t+1}^{(j)})} w_t^{(i)}$

Figure 5.6: Neural network model for cross-connected sequential importance sampling.

composed of signals from multiple sensory modalities, is embodied in the activity of a population of neurons $\{y_t^{(k)}\}$ [Georgopoulos *et al.*, 1986; Liu *et al.*, 2003]. Moreover, $\{y_t^{(k)}\}$ are just a noisy representation of true observation Sy_t that the sequential inference is based on. Therefore, the real estimation problem is to compute the posterior $p(x_t|Sy_{1:t})$. And $\{y_t^{(k)}\}$ is the intermediate step between Sy_t and $\{x_t^{(i)}\}$: They are a group of feature detection neurons with preferred stimuli at $\{y_t^{(k)}\}$ and activities proportional to $p(Sy_t|y_t^{(k)})$. Thus,

$$p(Sy_t|x_t) = \int p(Sy_t|y_t)p(y_t|x_t)dy_t = E_{p(y_t|x_t)}[p(Sy_t|y_t)] \qquad (5.22)$$

Applying importance sampling to approximate the posterior expectation,

$$E_{p(y_t|x_t)}[p(Sy_t|y_t)] \approx \sum_k p(Sy_t|y_t^{(k)})\frac{p(y_t|x_t)}{q_y(y_t^{(k)})} \qquad y_t^{(k)} \sim q_y(y) \qquad (5.23)$$

Replacing $p(y_{t+1}|x^{(j)}_{t+1})$ by Eq. 5.23 in weight updating rule Eq. 5.30, we get:

$$w^{(j)}_{t+1} \propto [\sum_k p(Sy_{t+1}|y^{(k)}_{t+1}) \frac{p(y^{(k)}_{t+1}|x^{(j)}_{t+1})}{q_y(y^{(k)}_{t+1})}] \cdot [\sum_i \frac{p(x^{(j)}_{t+1}|x^{(i)}_t)}{q(x^{(j)}_{t+1})} w^{(i)}_t]. \tag{5.24}$$

Thus, a single input y_{t+1} becomes a bundle of input fibers $\{y^{(k)}_{t+1}\}$ with activities $p(Sy_{t+1}|y^{(k)}_{t+1})$ (blue neurons in Fig. 5.7).

Multiplication is an important operation in neural computation [Gabbiani *et al.*, 2002; Barlow and Levick, 1965; Sun and Frost, 1998]. It can be realized by means of nonlinear dendritic integration [London and Hausser, 2005] which requires spatial co-localization of two synaptic inputs on a single dendritic branch. If the synaptic inputs are distributed over the dendritic tree, linear synaptic integration becomes dominant [London and Hausser, 2005]. Moreover, the super-linear interaction on a dendritic branch saturates quickly for large synaptic inputs [London and Hausser, 2005]. The limited dynamical range suggests that, rather than multiplying two summation terms $\sum_k(\cdot)$ and $\sum_i(\cdot)$ in Eq. 5.24, they can be broken into smaller terms for multiplication first (on a single branch) and then sum these terms over the dendritic tree, i.e.

$$w^{(j)}_{t+1} \propto \sum_k \sum_i \frac{p(y^{(k)}_{t+1}|x^{(j)}_{t+1})}{q_y(y^{(k)}_{t+1})}) p(Sy_{t+1}|y^{(k)}_{t+1}) \cdot \frac{p(x^{(j)}_{t+1}|x^{(i)}_t)}{q(x^{(j)}_{t+1})}) w^{(i)}_t. \tag{5.25}$$

Note that $p(Sy_{t+1}|y^{(k)}_{t+1})$ and $w^{(i)}_t$ are activities of neurons $y^{(k)}_{t+1}$ and $x^{(i)}_t$ respectively. The 'local-multiplication' of neuron pair happens on a single branch of the extensive dendritic tree of neuron $x^{(j)}_{t+1}$ (Fig. 5.7, right panel).

A sequential estimation problem may last for a long and indefinite period of time. If it requires recruiting a different group of particles at every time step, a task could demand a daunting number of particles and makes the logistics difficult, such as feeding common sensory inputs to all particles. An alternative is to recycle particles repeatedly through a delayed line, e.g. a delay neuron (see Fig.5.7). A possible realization of the time delay function is to use the biophysical property of the passive dendritic membrane: A brief and sharp excitatory postsynaptic potential (EPSP) from distal passive dendrites will be transformed into a smaller but broader signal, which results in the delay of output spikes. Through time delay neurons, current particles (green neuron in Fig.5.7) become the particles from previous time step (purple neuron in Fig.5.7) and participate in the inference process with new sensory inputs.

Divisive normalization is a common function in nervous systems [Reynolds and Heeger, 2009; Wainwright *et al.*, 2001], often realized by ubiquitous inhibitory interneurons [Markram *et al.*, 2004]. In the CC-SIS algorithm, normalization of certain terms, although not mandatory, does help to achieve numerical stability. For example, every sensory input neuron's activity $p(Sy_{t+1}|y_{t+1}^{(k)})$ can be normalized by the summation of themselves, i.e. $\frac{p(Sy_{t+1}|y_{t+1}^{(k)})}{\sum_k p(Sy_{t+1}|y_{t+1}^{(k)})}$. Similarly, the activities of particles $w_{t+1}^{(j)}$ can be normalized by the summation of themselves, i.e. $\frac{w_{t+1}^{(j)}}{\sum_j w_{t+1}^{(j)}}$.

Theoretically, $x_t^{(i)}$ and $x_{t+1}^{(j)}$ should be fully cross-connected. However, the transition probabilities $p(x_{t+1}^{(j)}|x_t^{(i)})$ can be diminishingly small compared to the others. Connections between these neurons can be dropped without effect the performance of the algorithm.

5.5.3 The cerebellum as a neural substrate of cross-connected sequential importance sampling

The cerebellum is a brain region that plays an important role in planing sequential motor control [Paulin, 1993]. This requires the integration of sensory inputs and estimation of internal states sequentially [Liu *et al.*, 2003; Fine *et al.*, 2002]. The essential operation of cerebellum is to produce predictive signals to control muscle

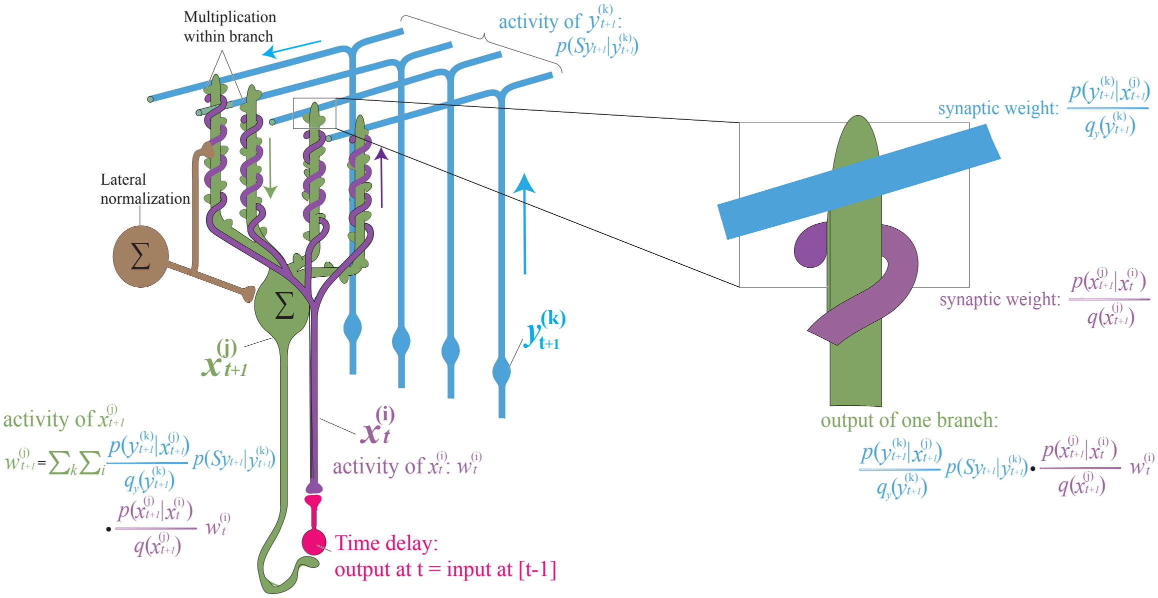

Multiplication within branch
activity of $y^{(k)}_{t+1}$:
$p(Sy_{t+1}|y^{(k)}_{t+1})$
Lateral normalization
Σ
Σ
$x^{(j)}_{t+1}$
$y^{(k)}_{t+1}$
activity of $x^{(j)}_{t+1}$
$w^{(j)}_{t+1}=\sum_k\sum_i\frac{p(y^{(k)}_{t+1}|x^{(j)}_{t+1})}{q_y(y^{(k)}_{t+1})}p(Sy_{t+1}|y^{(k)}_{t+1})\bullet\frac{p(x^{(j)}_{t+1}|x^{(i)}_t)}{q(x^{(j)}_{t+1})}w^{(i)}_t$
$x^{(i)}_t$
activity of $x^{(i)}_t$: $w^{(i)}_t$
Time delay:
output at t = input at [t-1]
synaptic weight: $\frac{p(y^{(k)}_{t+1}|x^{(j)}_{t+1})}{q_y(y^{(k)}_{t+1})}$
synaptic weight: $\frac{p(x^{(j)}_{t+1}|x^{(i)}_t)}{q(x^{(j)}_{t+1})}$
output of one branch:
$\frac{p(y^{(k)}_{t+1}|x^{(j)}_{t+1})}{q_y(y^{(k)}_{t+1})}p(Sy_{t+1}|y^{(k)}_{t+1})\bullet\frac{p(x^{(j)}_{t+1}|x^{(i)}_t)}{q(x^{(j)}_{t+1})}w^{(i)}_t$

Figure 5.7: Neural implementation of cross-connected sequential importance sampling. A bundle of input fibers $y^{(k)}_{t+1}$ (blue) is multiplied with the previous hidden state $x^{(i)}_t$ (purple) in the branch-specific way in the dendritic tree of neuron $x^{(j)}_{t+1}$. The activity of new hidden state, through a time delay neuron, is served as the previous state for the next time step. Zoom-in of the multiplication on a dendritic branch is shown in the right panel. Only one example neuron $x^{(i)}_t$ and $x^{(j)}_{t+1}$ are included. Neural implementation should include multiple $x^{(i)}_t$ and $x^{(j)}_{t+1}$ with cross-connections between them.

movement based on sensory inputs. This sequential computation is probabilistic in nature, because of the internal noise in nervous system and uncertainty in perception inputs. Therefore, cross-connected sequential importance sampling may serve as the underlying algorithm implemented by the cerebellum, with $\mathbf{x_{0:t}}$ denoting information associated with predictive motor commands and $\mathbf{y_{0:t}}$ sensory inputs.

Cerebellar neural circuits exhibit highly regulated structure [Bell *et al.*, 2008] (Fig. 5.8). This structure has two afferent pathways. One pathway is called the mossy fiber-parallel fiber system, channeling in sensory inputs from brain stem nuclei and the spinal cord. The second pathway is the climbing fiber system from the contralateral inferior oliver. Parallel fibers and climbing fibers interact with Purkinje cells at their extensive dendritic tree. Purkinje cells modulate deep cerebellar nuclei cells, which also receive inputs from inferior oliver and mossy fibers. Then deep cerebellar nuclei cells send efferent pathways that leave the cerebellum to regulate cerebral motor areas.

Comparing Fig.5.8 and Fig.5.7, we see that the neural implementation of cross-connected sequential importance sampling is similar to cerebellar circuits: Granule cells function as sensory input neurons $\{y_{t+1}^{(k)}\}$; Purkinje cells function as hidden variable neurons $\{x_{t+1}^{(j)}\}$ and climbing fibers function as $\{x_t^{(i)}\}$. Most notably, the synaptic interaction at the dendritic branches of Purkinje cells agree with the neuronal multiplication-sum structure in CC-SIS (Fig.5.7). Due to the functional and structural resemblance, we would like to hypothesize that the cerebellum implements cross-connected sequential importance sampling.

The activities of the same set of Purkinje cells and deep cerebellar nuclei cells are highly correlated to the onset of repeated movement during alternating movement. This fact suggests that one estimation task can use the same group of hidden variable neurons (or particles) repeatedly over time, an important assumption in the model. Since a coherent body motion involves a large amount of muscle, the hidden state

$\{x_{t+1}^{(j)}\}$ is of very high dimension and should be divided into multiple compartments (e.g. limbs). Thus, it raises the challenge of sensory inputs needing to reach many groups of particles. This challenge is met by the layout of the cerebellum, with groups of Purkinje cells' dendritic branches lying in a plane at right angles to the trajectory of the parallel fibers.

Various types of interneurons in cerebellar cortex modulate major pathways in these circuits[Apps and Garwicz, 2005]. Many of them are ideal candidates to perform divisive normalization and provide stability in computation. For example, basket cells form one of the most powerful inhibitory complex of synapses made around the Purkinje cell bodies, which can normalize the activity of $\{x_{t+1}^{(j)}\}$ neurons, i.e. normalizing the weights $\frac{w_{t+1}^{(j)}}{\sum_j w_{t+1}^{(j)}}$. Golgi cells, which receive input from parallel fibers and project its inhibitory outputs to the origin of these fibers (Granule cells), can provide normalization among sensory inputs, i.e. $\frac{p(Sy_{t+1}|y_{t+1}^{(k)})}{\sum_k p(Sy_{t+1}|y_{t+1}^{(k)})}$.

5.6 Discussion

Here, we provide a computational framework on neural implementation of dynamical inferences. Cross-connected sequential importance sampling is capable of sequential estimation and performs as good as sophisticated particle filtering algorithms. We suggest that cerebellar cortex provides an ideal structure to perform sequential estimation by CC-SIS. However, the cerebellum, although having a regular anatomic structure, is still incredibly complex for a model to capture all aspects. Especially, the model's time delay loop ($\{x_{t+1}^{(j)}\}$ neurons $\rightarrow$ time delay neuron $\rightarrow$ $\{x_t^{(i)}\}$) might be an oversimplified description to the cerebellar loop (Purkinje → Deep cerebellar nuclei cell → Inferior olive → climbing fiber). Moreover, it is worthwhile to keep in mind that the hidden variable $\{x_{t+1}^{(j)}\}$ is an abstract formulation of motor control problem and may take various forms. For example, Masao Ito [Ito, 1984] suggests that

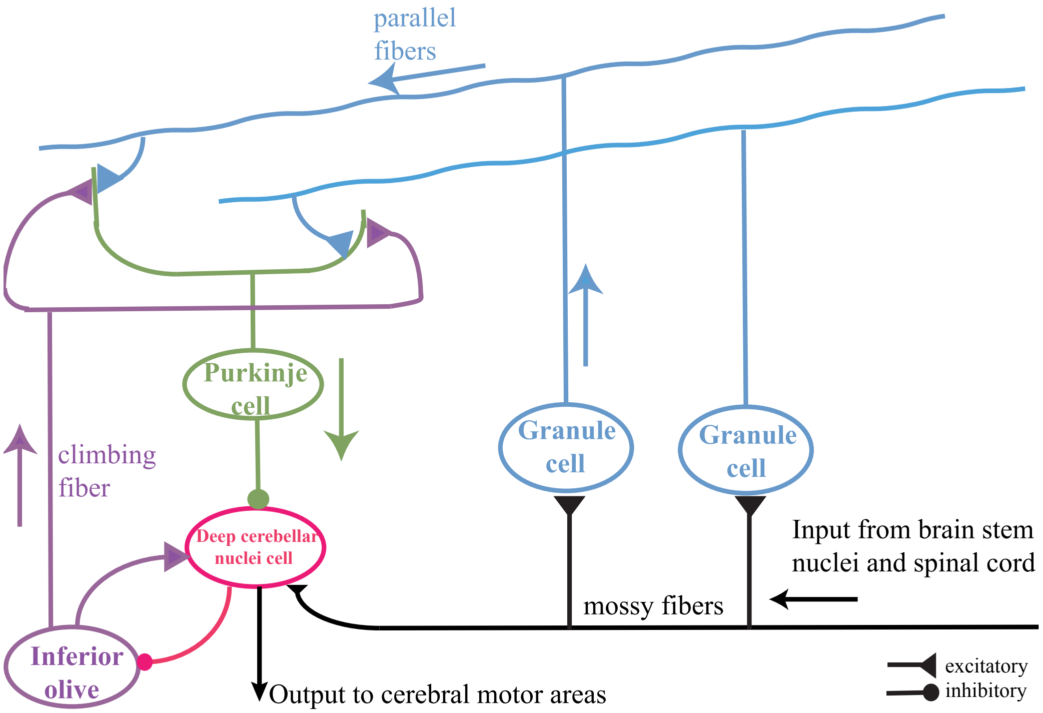


Figure 5.8: A cerebellar neural circuit diagram.

$\{x_{t+1}^{(j)}\}$ can take a form of error signal in climbing fibers. Nevertheless, our work provides a framework to understand cerebellum in the context of sequential estimation and, hopefully, give rise to more interests in cross-connected sequential importance sampling as a general neural mechanism for sequential and hierarchical probabilistic inference.

Appendix 5.A Derivation of cross-connected sequential importance sampling

Assume the posterior probability $p(x_t|y_{1:T})$ can be approximated by a set of particles $\{x_t^{(i)}\}$ and associated weights $\{w_t^{(i)}\}$, i.e.

$$p(x_t|y_{1:T}) \approx \sum_i w_t^{(i)} \delta(x_t - x_t^{(i)}) \tag{5.26}$$

Then,

$$\begin{aligned} p(x_{t+1}|y_{1:t+1}) &= \int p(x_{t+1}|x_t)p(y_{t+1}|x_{t+1})p(x_t|y_{1:t})\ dx_t \\ &\approx \sum_i p(x_{t+1}|x_t^{(i)})p(y_{t+1}|x_{t+1})w_t^{(i)} \end{aligned} \tag{5.27}$$

Meanwhile, applying importance sampling,

$$\begin{aligned} p(x_{t+1}|y_{1:t+1}) &= \int \delta(x - x_{t+1})p(x|y_{1:t+1})\ dx_t \\ &\approx \sum_j \frac{p(x_{t+1}^{(j)}|y_{1:t+1})}{q(x_{t+1}^{(j)})}\delta(x_{t+1}^{(j)} - x_{t+1}) \qquad x_{t+1}^{(j)} \sim q(x). \end{aligned} \tag{5.28}$$

Using Eq. 5.27, we find

$$p(x_{t+1}|y_{1:t+1}) \approx \sum_j \frac{\sum_i p(x_{t+1}^{(j)}|x_t^{(i)})p(y_{t+1}|x_{t+1}^{(j)}w_t^{(i)})}{q(x_{t+1}^{(j)})}\delta(x_{t+1}^{(j)} - x_{t+1}). \tag{5.29}$$

Therefore, the weight updating rule is

$$\begin{aligned} w_{t+1}^{(j)} &\propto \frac{\sum_i p(x_{t+1}^{(j)}|x_t^{(i)})p(y_{t+1}|x_{t+1}^{(j)})}{q(x_{t+1}^{(j)})} w_t^{(i)} \\ &= p(y_{t+1}|x_{t+1}^{(j)}) \sum_i \frac{p(x_{t+1}^{(j)}|x_t^{(i)})}{q(x_{t+1}^{(j)})} w_t^{(i)}, \end{aligned} \tag{5.30}$$

subject to a normalization constant.

Chapter 6

Conclusion

In this thesis, we studied psychological and neural implementations of Bayesian inference. Previous work typically addresses Marr's [1982] computational level. Here, we focus on the algorithmic level and the implementation level. Specifically, we built our models around a Monte Carlo method known as importance sampling and showed how importance sampling can explain human behavior in cognitive and perceptual tasks. The basic idea behind importance sampling – storing examples and activating them based on similarity – is at the heart of a variety of psychological models, neural network models and machine learning algorithms. Moreover, importance sampling can be extended to model neural functions in hierarchical Bayesian inference and sequential Bayesian inference.

We have presented both theoretical results and simulations showing that exemplar models provide a simple, psychologically plausible mechanism for performing at least some kinds of Bayesian inference. Our theoretical results indicate that exemplar models can be interpreted as a form of importance sampling, and can thus implement an approximation to Bayesian inference. Our simulations demonstrate that this approach produces predictions that correspond reasonably well with human behavior, and that relatively few exemplars are needed to provide a good approximation to the true Bayesian solution in at least five settings.

Understanding how the brain solves the problem of hierarchical Bayesian inference is a significant challenge for computational neuroscience. In this thesis, we have shown how a potential solution is provided by a multilayer neural network implementing a recursive scheme of importance sampling. This model has a simple neural implementation, either with deterministic or spiking neurons, and can perform tasks such as sensorimotor learning and cue-combination with a small number of feature detection neurons. It can also explain some characteristic behaviors in perception, such as the oblique effect. Another challenge in modeling Bayesian inference in the brain is understanding the neural mechanisms for sequential inference. We showed that cross-connected sequential importance sampling, an algorithm based on recursive importance sampling can perform sequential inference as well as the state-of-art machine learning algorithms know as particle filters. Further study suggests that neural implementation of CC-SIS is closely related to cerebellar circuits. This led us to propose CC-SIS as the underlying mechanism for cerebellum's motor coordination function.

The approach that we have taken in this thesis represents one way of addressing questions about the algorithms and mechanisms that could support probabilistic inference. Our results suggest that 1) exemplar models are not simply process models, but rational process models – an effective and psychologically plausible scheme for approximating statistical inference; and 2) recursive importance sampling is a potential mechanism to implement the rational process models by the brain to perform hierarchical Bayesian inference and sequential Bayesian inference. This approach pushes the principle of optimality that underlies probabilistic models down to the level of algorithm and mechanism, and suggests a general strategy for explaining how people perform Bayesian inference: Look for connections between neural circuits, psychological process models and approximate inference algorithms developed in computer science and statistics.

References

[Anderson and Milson, 1989] J. R. Anderson and R. Milson. Human memory: An adaptive perspective. *Psychological Review*, 96:703–719, 1989.

[Anderson, 1990] J. R. Anderson. *The adaptive character of thought*. Erlbaum, Hillsdale, NJ, 1990.

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

[Apps and Garwicz, 2005] R. Apps and M. Garwicz. Anatomical and physiological foundations of cerebellar information processing. *Nat Rev Neurosci*, 6(4):297–311, 2005.

[Arulampalam *et al.*, 2001] S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for on-line non-linear/non-gaussian bayesian tracking. *IEEE Transactions on Signal Processing*, 50:174–188, 2001.

[Ashby and Alfonso-Reese, 1995] F. G. Ashby and L. A. Alfonso-Reese. Categorization as probability density estimation. *Journal of Mathematical Psychology*, 39:216–233, 1995.

[Barbieri *et al.*, 2004] R. Barbieri, L. M. Frank, D. P. Nguyen, M. C. Quirk, V. Solo, M. A. Wilson, and E. N. Brown. Dynamic analyses of information encoding in neural ensembles. *Neural Comput.*, 16(2):277–307, 2004.

[Barlow and Levick, 1965] H. B. Barlow and W. R. Levick. The mechanism of directionally selective units in rabbit's retina. *J Physiol*, 178(3):477–504, 1965.

[Basso and Wurtz, 1997] M. A. Basso and R. H. Wurtz. Modulation of neuronal activity by target uncertainty. *Nature*, 389(6646):66–69, 1997.

[Bell *et al.*, 2008] C. C. Bell, V. Han, and N. B. Sawtell. Cerebellum-like structures and their implications for cerebellar function. *Annu Rev Neurosci*, 31:1–24, 2008.

[Blaisdell *et al.*, 2006] A. P. Blaisdell, K. Sawa, K. J. Leising, and M. R. Waldmann. Causal reasoning in rats. *Science*, 311(5763):1020–1022, 2006.

[Brown and Steyvers, 2009] S. D. Brown and M. Steyvers. Detecting and predicting changes. *Cognitive Psychology*, 58:49–67, 2009.

[Coppola *et al.*, 1998] D. M. Coppola, L. E. White, D. Fitzpatrick, and D. Purves. Unequal representation of cardinal and oblique contours in ferret visual cortex. *Proc Natl Acad Sci U S A*, 95(5):2621–2623, 1998.

[Daw and Courville, 2008] N. Daw and A. C. Courville. The pigeon as particle filter. In J.C. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20.* MIT Press, Cambridge, MA, 2008.

[De Valois *et al.*, 1982] R. L. De Valois, E. W. Yund, and N. Hepler. The orientation and direction selectivity of cells in macaque visual cortex. *Vision Res*, 22(5):531–544, 1982.

[DeLosh *et al.*, 1997] E. L. DeLosh, J. R. Busemeyer, and M. A. McDaniel. Extrapolation: The sine qua non of abstraction in function learning. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 23:968–986, 1997.

[Doucet *et al.*, 2001] A. Doucet, N. de Freitas, and N. Gordon. *Sequential Monte Carlo Methods in Practice.* Springer, New York, 2001.

[Doya, 2007] K. Doya. *Bayesian brain: probabilistic approaches to neural coding.* Computational neuroscience. MIT Press, Cambridge, Mass., 2007.

[Ernst and Banks, 2002] M. O. Ernst and M. S. Banks. Humans integrate visual and haptic information in a statistically optimal fashion. *Nature*, 415(6870):429–433, 2002.

[Feldman *et al.*, 2009] N. H. Feldman, T. L. Griffiths, and J. L. Morgan. The influence of categories on perception: Explaining the perceptual magnet effect as optimal statistical inference. *Psychological Review*, 116:752–782, 2009.

[Fine *et al.*, 2002] E. J. Fine, C. C. Ionita, and L. Lohr. The history of the development of the cerebellar examination. *Semin Neurol*, 22(4):375–384, 2002.

[Friston, 2008] K. Friston. Hierarchical models in the brain. *PLoS Comput Biol*, 4(11):e1000211, 2008.

[Fry *et al.*, 1962] D. B. Fry, Arthur S. Abramson, P. D. Eimas, and A. M. Liberman. The identification and discrimination of synthetic vowels. *Language and Speech*, 5:171–189, 1962.

[Furmanski and Engel, 2000] C. S. Furmanski and S. A. Engel. An oblique effect in human primary visual cortex. *Nat Neurosci*, 3(6):535–536, 2000.

[Gabbiani *et al.*, 2002] F. Gabbiani, H. G. Krapp, C. Koch, and G. Laurent. Multiplicative computation in a visual neuron sensitive to looming + views. *Nature*, 420(6913):320–324, 2002.

[Georgopoulos *et al.*, 1986] A.P. Georgopoulos, A.B. Schwartz, and R.E. Kettner. Neuronal population coding of movement direction. *Science*, 233(4771):1416–1419, SEP 26 1986.

[Gigerenzer and Todd, 1999] G. Gigerenzer and P. Todd. *Simple heuristics that make us smart.* Oxford University Press, New York, 1999.

[Gordon *et al.*, 1993] N. J. Gordon, D. J. Salmond, and A. F. M. Smith. Novel Approach to Nonlinear/Non-Gaussian Bayesian State Estimation. *Radar and Signal Processing, IEEE Proceedings F*, 140(2):107–113, 1993.

[Griffiths and Tenenbaum, 2006] T. L. Griffiths and J. B. Tenenbaum. Optimal predictions in everyday cognition. *Psychological Science*, 17:767–773, 2006.

[Griffiths and Tenenbaum, 2007] T. L. Griffiths and J. B. Tenenbaum. Two proposals for causal grammars. In A. Gopnik and L. Schulz, editors, *Causal learning: Psychology, philosophy, and computation.* Oxford University Press, Oxford, 2007.

[Guenther and Gjaja, 1996] F. H. Guenther and M. N. Gjaja. The perceptual magnet effect as an emergent property of neural map formation. *Journal of the Acoustical Society of America*, 100(2):1111–1121, 1996.

[Hastings, 1970] W. K. Hastings. Monte-carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1):97–&, 1970.

[Hemmer and Steyvers, 2009] P. Hemmer and M. Steyvers. A Bayesian account of reconstructive memory. *Topics in Cognitive Science*, 1:189–202, 2009.

[Hinton and Ghahramani, 1997] G. E. Hinton and Z. Ghahramani. Generative models for discovering sparse distributed representations. *Philos Trans R Soc Lond B Biol Sci*, 352(1358):1177–1190, 1997.

[Hodzic *et al.*, 2004] A. Hodzic, R. Veit, A. A. Karim, M. Erb, and B. Godde. Improvement and decline in tactile discrimination behavior after cortical plasticity induced by passive tactile coactivation. *J Neurosci*, 24(2):442–446, 2004.

[Hoyer and Hyvärinen, 2002] P. O. Hoyer and A. Hyvärinen. Interpreting neural response variability as monte carlo sampling of the posterior. In Suzanna Becker, Sebastian Thrun, and Klaus Obermayer, editors, *NIPS*, pages 277–284. MIT Press, 2002.

[Huttenlocher *et al.*, 2000] J. Huttenlocher, L. V. Hedges, and J. L. Vevea. Why do categories affect stimulus judgment? *Journal of Experimental Psychology: General*, 129:220–241, 2000.

[Isard, 2003] M. Isard. Pampas: real-valued graphical models for computer vision. *Proc. Comput. Vision Pattern Recog*, 2003.

[Ito, 1984] M. Ito. *Cerebellum and Neural Control*. Raven Pr, 1984.

[Iverson and Kuhl, 1995] P. Iverson and P. K. Kuhl. Mapping the perceptual magnet effect for speech using signal detection theory and multidimensional scaling. *Journal of the Acoustical Society of America*, 97(1):553–562, 1995.

[Juslin and Persson, 2002] P. Juslin and M. Persson. PROBabilities from EXemplars (PROBEX): a lazy algorithm for probabilistic inference from generic knowledge. *Cognitive Science*, 26:563–607, 2002.

[Kahneman and Tversky, 1972] D. Kahneman and A. Tversky. Subjective probability: A judgment of representativeness. *Cognitive Psychology*, 3:430–454, 1972.

[Khan *et al.*, 2004] Z. Khan, T. Balch, and F. Dellaert. A rao-blackwellized particle filter for eigentracking. In *In IEEE Computer Vision and Pattern Recognition*, volume 2, pages 980–986, 2004.

[Kitagawa, 1996] G. Kitagawa. Monte carlo filter and smoother for non-gaussian nonlinear state space models. *Journal of Computational and Graphical Statistics*, 5(1):1–25, 1996.

[Körding and Wolpert, 2004] K. Körding and D. M. Wolpert. Bayesian integration in sensorimotor learning. *Nature*, 427:244–247, 2004.

[Kouh and Poggio, 2008] M. Kouh and T. Poggio. A canonical neural circuit for cortical nonlinear operations. *Neural Comput*, 20(6):1427–1451, 2008.

[Kruschke, 1992] J. K. Kruschke. Alcove: An exemplar-based connectionist model of category learning. *Psychological Review*, 99:22–44, 1992.

[Kruschke, 2006] J. K. Kruschke. Locally Bayesian learning with applications to retrospective revaluation and highlighting. *Psychological Review*, 113:677–699, 2006.

[Kuhl *et al.*, 1992] P. K. Kuhl, K. A. Williams, F. Lacerda, K. N. Stevens, and B. Lindblom. Linguistic experience alters phonetic perception in infants by 6 months of age. *Science*, 255(5044):606–608, 1992.

[Lee and Mumford, 2003] T. S. Lee and D. Mumford. Hierarchical bayesian inference in visual cortex. *Journal of the Optical Society of America A*, 2003.

[Levy *et al.*, 2009] R. Levy, F. Reali, and T. L. Griffiths. Modeling the effects of memory on human online sentence processing with particle filters. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *Advances in Neural Information Processing Systems 21*, pages 937–944, 2009.

[Lewandowsky *et al.*, in press] S. Lewandowsky, T. L. Griffiths, and M. L. Kalish. The wisdom of individuals: Exploring people's knowledge about everyday events using iterated learning. *Cognitive Science*, in press.

[Liberman *et al.*, 1957] A. M. Liberman, K. S. Harris, H. S. Hoffman, and B. C. Griffith. The discrimination of speech sounds within and across phoneme boundaries. *Journal of Experimental Psychology*, 54(5):358–368, 1957.

[Liu *et al.*, 2003] X. Liu, E. Robertson, and R. C. Miall. Neuronal activity related to the visual representation of arm movements in the lateral cerebellar cortex. *J Neurophysiol*, 89(3):1223–1237, 2003.

[London and Hausser, 2005] M. London and M. Hausser. Dendritic computation. *Annu Rev Neurosci*, 28:503–532, 2005.

[Luce, 1959] R. D. Luce. *Individual choice behavior.* Wiley, New York, 1959.

[Ma *et al.*, 2006] W. J. Ma, J. M. Beck, P. E. Latham, and A. Pouget. Bayesian inference with probabilistic population codes. *Nat.Neurosci.*, 9(11):1432–1438, 2006.

[Markram *et al.*, 2004] H. Markram, M. Toledo-Rodriguez, Y. Wang, A. Gupta, G. Silberberg, and C. Wu. Interneurons of the neocortical inhibitory system. *Nat Rev Neurosci*, 5(10):793–807, 2004.

[Marr, 1982] D. Marr. *Vision.* W. H. Freeman, San Francisco, CA, 1982.

[Medin and Schaffer, 1978] D. L. Medin and M. M. Schaffer. Context theory of classification learning. *Psychological Review*, 85:207–238, 1978.

[Mitchell and Silver, 2003] S. J. Mitchell and R. A. Silver. Shunting inhibition modulates neuronal gain during synaptic excitation. *Neuron*, 38(3):433–445, 2003.

[Mozer *et al.*, 2008] M. Mozer, H. Pashler, and H. Homaei. Optimal predictions in everyday cognition: The wisdom of individuals or crowds? *Cognitive Science*, 32:1133–1147, 2008.

[Myung and Shepard, 1996] I. J. Myung and R. N. Shepard. Maximum entropy inference and stimulus generalization. *Journal of Mathematical Psychology*, 40:342–347, 1996.

[Neal, 1993] R. M. Neal. Probabilistic inference using Markov chain Monte Carlo methods. Technical Report CRG-TR-93-1, University of Toronto, University of Toronto, 1993.

[Nosofsky, 1986] R. M. Nosofsky. Attention, similarity, and the identification-categorization relationship. *Journal of Experimental Psychology: General*, 115:39–57, 1986.

[Oaksford and Chater, 1994] M. Oaksford and N. Chater. A rational analysis of the selection task as optimal data selection. *Psychological Review*, 101:608–631, 1994.

[Orban *et al.*, 1984] G. A. Orban, E. Vandenbussche, and R. Vogels. Human orientation discrimination tested with long stimuli. *Vision Res*, 24(2):121–128, 1984.

[Paulin, 1993] M. G. Paulin. The role of the cerebellum in motor control and perception. *Brain Behav Evol*, 41(1):39–50, 1993.

[Pearl, 1988] J. Pearl. *Probabilistic reasoning in intelligent systems.* Morgan Kaufmann, San Francisco, CA, 1988.

[Platt and Glimcher, 1999] M. L. Platt and P. W. Glimcher. Neural correlates of decision variables in parietal cortex. *Nature*, 400:233–238, 1999.

[Poggio and Girosi, 1990] T. Poggio and F. Girosi. Regularization algorithms for learning that are equivalent to multilayer networks. *Science*, 247(4945):978–982, 1990.

[Reynolds and Heeger, 2009] J. H. Reynolds and D. J. Heeger. The normalization model of attention. *Neuron*, 61(2):168–185, 2009.

[Robert and Casella, 1999] C. P. Robert and G. Casella. *Monte Carlo statistical methods.* Springer, New York, 1999.

[Rothman *et al.*, 2009] J. S. Rothman, L. Cathala, V. Steuber, and R A. Silver. Synaptic depression enables neuronal gain control. *Nature*, 457(7232):1015–1018, 2009.

[Sanborn *et al.*, 2006] A. N. Sanborn, T. L. Griffiths, and D. J. Navarro. A more rational model of categorization. In *Proceedings of the 28th Annual Conference of the Cognitive Science Society*, Mahwah, NJ, 2006. Erlbaum.

[Shepard, 1962] R. N. Shepard. The analysis of proximities: Multidimensional scaling with an unknown distance function: I. *Psychometrika*, 27:124–140, 1962.

[Shepard, 1987] R. N. Shepard. Towards a universal law of generalization for psychological science. *Science*, 237:1317–1323, 1987.

[Shi and Griffiths, in press] L. Shi and T. L. Griffiths. Neural implementation of hierarchical Bayesian inference by importance sampling. In J. Lafferty and C. K. I. Williams, editors, *Advances in Neural Information Processing Systems 22*, in press.

[Shiffrin and Steyvers, 1997] R. M. Shiffrin and M. Steyvers. A model for recognition memory: REM: Retrieving Effectively from Memory. *Psychonomic Bulletin & Review*, 4:145–166, 1997.

[Simon, 1957] H. A. Simon. *Models of man.* Wiley, New York, 1957.

[Smith and Zarate, 1992] E. R. Smith and M. A. Zarate. Exemplar-based model of social judgment. *Psychological Review*, 99:3–21, 1992.

[Steyvers *et al.*, 2003] M. Steyvers, J. B. Tenenbaum, E. J. Wagenmakers, and B. Blum. Inferring causal networks from observations and interventions. *Cognitive Science*, 27:453–489, 2003.

[Stocker and Simoncelli, 2008] A. Stocker and E. Simoncelli. A bayesian model of conditioned perception. In J.C. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 1409–1416. MIT Press, Cambridge, MA, 2008.

[Sun and Frost, 1998] H. Sun and B. J. Frost. Computation of different optical variables of looming objects in pigeon nucleus rotundus neurons. *Nature neuroscience*, 1(4):296–303, 1998.

[Swindale, 1998] N. V. Swindale. Orientation tuning curves: empirical description and estimation of parameters. *Biol Cybern*, 78(1):45–56, 1998.

[Tenenbaum and Griffiths, 2001] J. B. Tenenbaum and T. L. Griffiths. Generalization, similarity, and Bayesian inference. *Behavioral and Brain Sciences*, 24:629–641, 2001.

[Tenenbaum, 1999] J. B. Tenenbaum. *A Bayesian framework for concept learning.* PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, 1999.

[Van Essen *et al.*, 1992] D. C. Van Essen, C. H. Anderson, and D. J. Felleman. Information processing in the primate visual system: an integrated systems perspective. *Science*, 255(5043):419–423, 1992.

[Wainwright *et al.*, 2001] M. Wainwright, O. Schwartz, and E. Simoncelli. Natural image statistics and divisive normalization: Modeling nonlinearity and adaptation in cortical neurons, 2001.

[Welch and Bishop, 1995] G. Welch and G. Bishop. An introduction to the kalman filter. Technical report, Chapel Hill, NC, USA, 1995.

[Yi *et al.*, in press] S. K. M. Yi, M. Steyvers, and M. D. Lee. Modeling human performance on restless bandit problems using particle filters. *Journal of Problem Solving*, in press.

[Yuille and Kersten, 2006] A. Yuille and D. Kersten. Vision as Bayesian inference: analysis by synthesis? *Trends in Cognitive Sciences*, 10:301–308, 2006.

[Zemel *et al.*, 1998] R. S. Zemel, P. Dayan, and A. Pouget. Probabilistic interpretation of population codes. *Neural Comput*, 10(2):403–430, 1998.